LOKSYS (Locking Systems)
Deployable Prototype

Created by TEAM 2

Phillip Davis,
CSU Sacramento
Computer Engineering
pjdavis1988@hotmail.com

Adam Batakji,
CSU Sacramento
Computer Engineering
adambatakji@yahoo.com

Joel Barrera,
CSU Sacramento
Computer Engineering
jbjoel91@gmail.com

Phuc Nguyen,
CSU Sacramento
Computer Engineering
kevin_nguyen_12@hotmail.com

Table of Contents

I. Societal Problem 4
- **A. Introduction 4**
- **B. Negative Effects on Society 5**
- **C. Current Solutions and Their Shortcomings 5**
- **D. Conclusion (our solution) 6**

II. Design Idea 7
- **A. Design Overview 7**
- **B. Design Details 7**
  - 1) Cabinet Locker 7
  - 2) Mobile Lock 7
  - 3) Door Lock 8
  - 4) All locks 8

III. Funding 8

IV. Project Work Breakdown Structure 8
- **A. Overview 8**
- **B. Conclusion 11**

V. Project Timeline 13

VI. Risk Assessment and Mitigation 14

VII. Weekly Reports (Fall 2014) 15
- **A. Week 2 ..... 15**
- **B. Week 3 ..... 21**
- **C. Week 4 ..... 27**
- **D. Week 5 ..... 33**
- **E. Week 6 ..... 39**
- **F. Week 7 ..... 45**
- **G. Week 8 ..... 51**
- **H. Week 9 ..... 57**
- **I. Week 10 ..... 63**
- **J. Week 11 ..... 69**
- **K. Week 12 ..... 75**
- **L. Week 13 ..... 81**
- **M. Week 14 ..... 87**

VIII. Weekly Reports (Spring 2015) 93
- **A. Week 1 ..... 93**
- **B. Week 2 ..... 99**
- **C. Week 3 ..... 105**
- **D. Week 4 ..... 111**
- **E. Week 5 ..... 117**
- **F. Week 6 ..... 123**
- **G. Week 7 ..... 129**
- **H. Week 8 ..... 135**
- **I. Week 9 ..... 141**
- **J. Week 10 ..... 147**

K. Week 11 153
L. Week 12 159
M. Week 13 165
N. Week 14 171

IX. User Manual 177
A. How to use web page: 177
B. Mobile Application 177
C. Door Lock 177
D. Cabinet Lock 178
E. Mobile Lock 178

X. Hardware Block Diagram 180

XI. Software Flowchart 181

XII. Photo Documentation 182

XIII. Mechanical 183

XIV. Hardware Test Plan Results 185
A. Arduino Microcontroller Testing 185
B. Solenoid and Electric Door Strike Testing 185
C. RFID Reader Testing 185
D. Power Adaptor Voltage Measurements 185
E. Other Components/Sensor Testing 185
F. Power Outage 186

XV. Software Test Plan Results 186
A. User Interface 186
B. Security 186
C. Stress Testing 187
D. Browser Capability 187
E. Error Handling 187

XVI. Appendixes i
A. Data Sheets i
1) LCD2004 (HD44780) i
2) RFID 20-LA lii
3) SainSmart Sensor Shield v5.0 lxix
4) Songle Relay lxx
5) W5100 lxxii
6) CC3000 cxliii
B. Resumes clix
1) Phillip Davis clix
2) Adam Batakji clx
3) Kevin Nguyen clxi
4) Joel Barrera clxii
C. References clxiii
D. Glossary clxv

End of Term Documentation
Deployable Prototype:
LOKSYS (Locking Systems)

CPE 191 – Senior Design Project II
DEPLOYABLE PROTOTYPE DOCUMENTATION
Due: 3 May 2015

Phillip Davis,
CSU Sacramento
Computer Engineering
pjdavis1988@hotmail.com

Adam Batakji,
CSU Sacramento
Computer Engineering
adambatakji@yahoo.com

Joel Barrera,
CSU Sacramento
Computer Engineering
jbjoel91@gmail.com

Phuc Nguyen,
CSU Sacramento
Computer Engineering
kevin_nguyen_12@hotmail.com

***Abstract*—**The goal of this document is to share all of the information regarding Team 2's senior design project, the locking system (Loksys). The reader will be provided with documentation and data of the project, including problem statement, design idea, work breakdown structure, a user guide, and more.

***Keywords*—** *Radio Frequency Identification (RFID; Societal Problem; Design Idea; Work Breakdown Structure; Mobile Lock; Web Server*

I. SOCIETAL PROBLEM

A. Introduction

One of the most prevalent and persistent problems in human society is the one that Team 2 attempted to tackle. From bandits to pirates to people at war, theft has been a part of human society as long as there have been records. All people have something of value, regardless of whether it is worth money or just something that has deep meaning to the individual. As much as the possession is deeply cared for, it is not always by their side or within their sight. They may hide it in a 'safe' place, but how safe is it?

Large objects locked in plain sight do not even seem to stand a chance. About half of all active cyclists have their bike stolen [2]. Be it that it bolt cutters did the job or the lock was picked. To prevent the former from occurring, the lock needs to be made of thick steel. As for the latter, any lock that is opened by a key means there is a way to pick it.

Figure 1 shows how serious the problem is even within a partial sample of just college students. With numbers like these, it can only be assumed that many cases go unreported as well. How can these numbers be reduced?

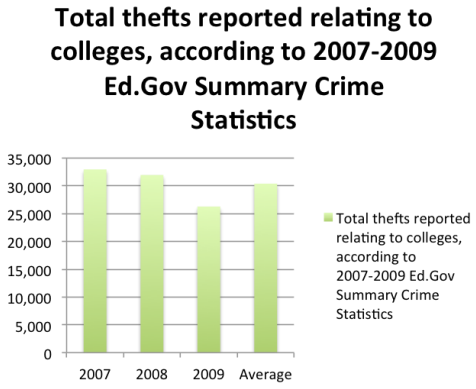


Fig. 1. Total thefts reported relating to colleges. [1]

Figure 2 showcases a crime survey of theft in the countries of England and Wales. As can be seen, it is more common to see theft of possession when the owner is away. As explained

in the beginning, you cannot always keep an eye on your possessions. The problem comes to be that items of value can still be stolen even after the fact that an effort has been made to prevent this.

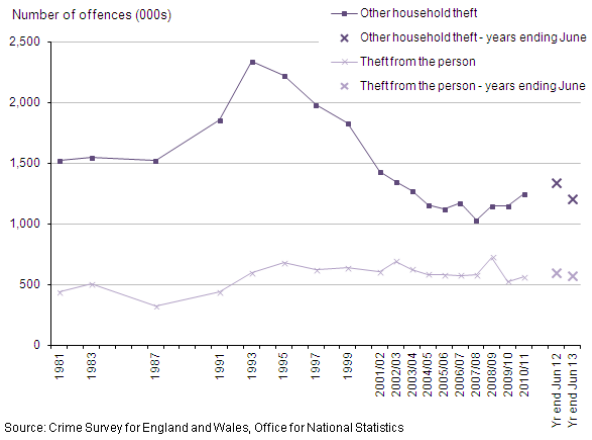


Fig. 2. Number of offences from 1981 through 2013. [2]

B. Negative Effects on Society

Theft, crime in general, has a detrimental effect on the health of society. P. Morrall from the University of Leeds conducted a study in which 886 students were surveyed from three universities in the UK. Results from this study indicated that there are serious psychological effects of crime. The data shown below in Figure [10] indicated that theft was the most frequent offense, with 25.5% of the total sample; and 30.0% of those who were victims of more than one offense [10].

Types of offence

Offence	Frequency	%
Mugging	7	3.6
Domestic	5	2.6
Racial	3	1.6
Burglary	19	9.9
Theft	49	25.5
Damage	28	14.6
Threat	20	10.4
Other	3	1.6
More than one offence	58	30.0
Total	192	100

Fig. 3. Frequency of each type of offense. [10]

As shown in Figure [3], not only does crime negatively affect the health of victims, but of non-victims as well. When asked about the fear of crime, both victims and non-victims reported high levels and similar stress-related symptoms: feeling anxious/stressed (62.1% victims; 65.1% non-victims), lacking confidence (35.3% victims; 38.3% non-victims), and experiencing sleeping difficulties (32.2% victims; 27.4% non-victims) [10].

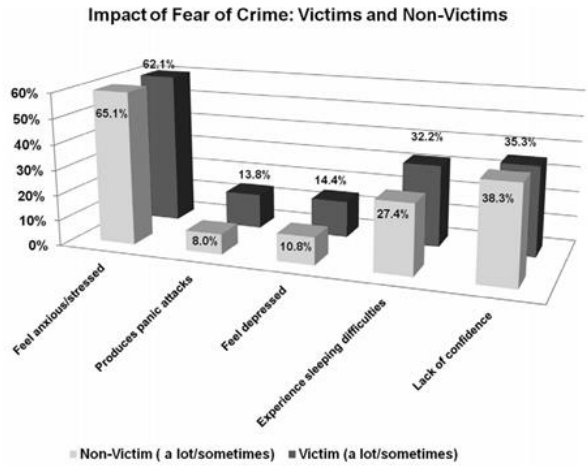


Fig. 4. Impact of fear and crime. [10]

Crime has induced a change in behavior. It causes us to not trust each other and become afraid. We do not give rides to strangers, and they do not give rides to us. 45.2% of women and 10.3% of men reported that they 'don't go out after dark' after being a victim. And 51.1% of women and 17.2% of men avoid going out alone. 65.4% of women and 32.4% of men reported that they 'avoid certain streets' [10].

C. Current Solutions and Their Shortcomings

When we look at current solutions to protecting our valuables, many options come to mind. Home security, lockers, backpacks, alarms, and video surveillance are only a few to name. With so many options available, one can't help but keep wondering; how is theft still reoccurring? A reasonable answer to this is the lack of security entirely or the shortcomings and inefficiency of the type of security. While we cannot completely dictate or enhance the behavior of people, which would lie as the best possible solution, understanding how security and lock systems work in general, whether they are small locks for your bike or a full scale home security, how we could potentially enhance them could greatly reduce the amount of offences, regardless of the criminal's behavior. Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

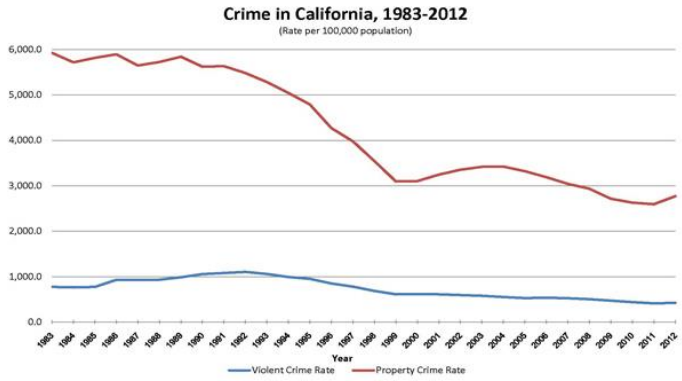


Fig. 5. Crime in California. [6]

The figure above, taken from the State of California Department of Justice, illustrates both the property crime rate

and the violent crime rate in California from the years of 1983 to a recent 2012. Something to note is the graphs steep decline from the years of 1991 to 1999, where it showed the amount of property crime rate dropping dramatically, and violent crime rate trending similarly. Although that is very great news to hear, it is not the main focus of the graph. If we continue to look at the years after 1999, we can see that these rates begin to flatten out, meaning that there has been little change in both crime rates. A reasonable explanation to this trend of burglary and crime rate, could be the issue of criminal adaptively. This includes the criminals have understood how security and lock systems work, that there are new methods to bypass them, and that these methods have potentially become more commercially inclined and deviated from the principal focus of safeguarding property.

Systems are undeniable becoming outdated and inefficient, as they are currently very susceptible to outside forces. An area of interest, aimed to prevent some of the shortcomings of current systems, is that of biometric technologies. Biometric technology enables automatic identification or verification of an individual based on the person's physiological or behavioral characteristics[7]. With criminal adaptively being an issue, systems that prevent any type of theft, need to adapt even faster. The need for reliable user authentication techniques has increased in the wake of heightened concerns about security and rapid advancements in networking, communication, and mobility [8].

Using a tool to break a bike lock, knowing how to pick a lock, how to break into a car, and even breaking into a garage in under six seconds are other examples and tutorials that can be looked up online. Investing into a system that cannot be affected by others would augment the privacy and security on many applications. Biometrics offers greater security and convenience than traditional identity authentication systems (based on passwords and cryptographic keys) since biometrics characteristics are inherently associated with a particular individual, making them insusceptible to being stolen, forgotten, lost or attached [9]. Security, without a doubt, needs to become more efficient and less susceptible to outside forces if we want to continue protecting our valuables.

D. Conclusion (our solution)

Team 2 has decided to use Radio Frequency Identification (RFID) to help solve some of the problems we have with security, including access to the “break in point” and the ability to update your security information. RFID is a generic term for technologies employing radio waves for detecting objects [3]. The two main elements in RFID are the reader and the tag. An RFID tag is a small device consisting of an integrated circuit and an antenna incorporated into any object or living being for tracking and identification by storing ID information or data [3]. A passive RFID tag draws power from field created by the reader and uses it to power the microchip’s circuit; and then the chip then modulates the waves that the tag sends back to the reader and the reader converts the new waves into digital data [4]. The reader converts the radio waves reflected back from the RFID tag into digital information that can then be passed on to computers that can make use of it [4]. We are going to use the reader and tag together to upgrade security on doors, cabinets, and even a portable lock that can help secure things when you are not at home.

The problems with combination locks are solved because the combination is much longer and can no longer be manually inserted. The problem with key locks are solved because there is no key hole to use metal on.

II. Design Idea

A. Design Overview

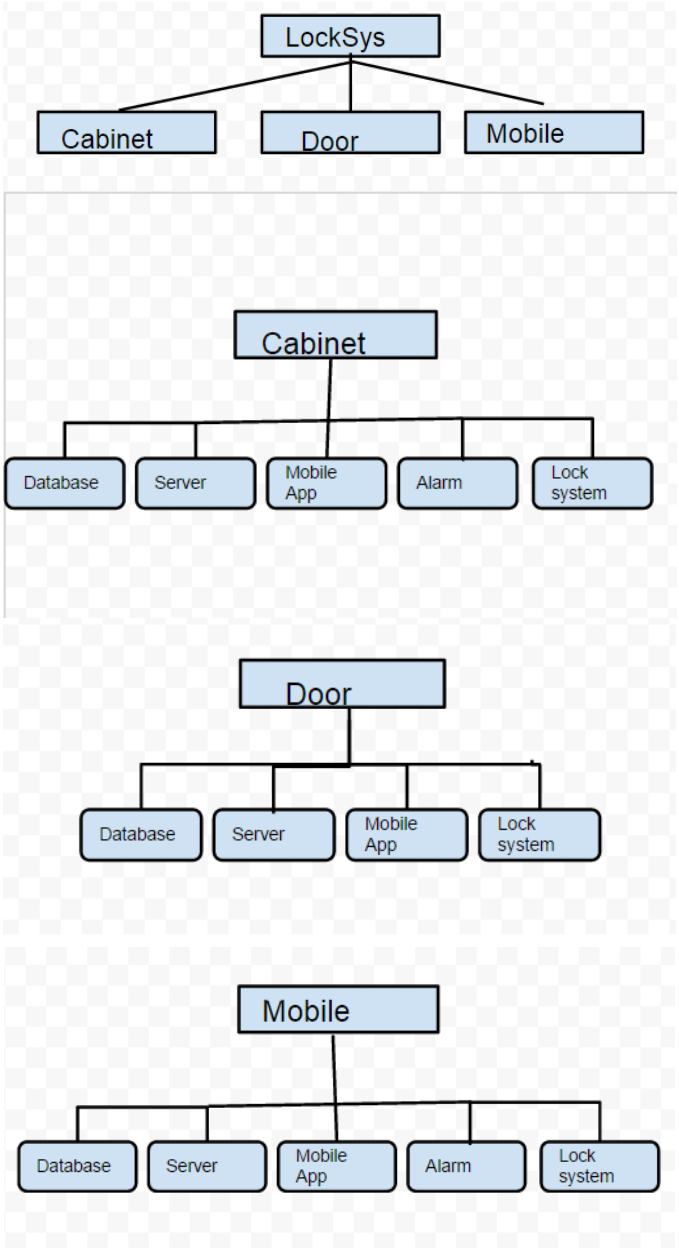


Fig. 6. Design overview.

In order to accomplish our goal, we have decided to build and write code for a locking system. We have three separate locks, a lock for cabinets, a mobile lock for lockers, or similar things that can be locked with a padlock, and a lock for a door. There will be feedback to the client about the locks, and the client will be able to give and remove permissions and users. We believe that our locks we can greatly reduce chance of theft, and add some convenience to the clients life at the same time.

Our cabinet lock is internal, so as to remove the ability to pick the lock. We keep a log of who is using the cabinet on a database, so that the client will be able to check the web server to see who used the cabinet. The cabinet has permission levels so that only certain people will have the ability to open certain cabinet doors. The cabinets also have an alarm system for unauthorized entry. There is a mobile application as well.

Our mobile lock has the ability to secure clients possessions similar in fashion to a padlock. It is equipped with alarm system for unauthorized entry.

Our final lock is a door lock. We remove key entry, once again, to deter lock picking. The door also has the ability to communicate with the database and server, and has a display that will output information.

Our devices are unique due to the decision to make it much harder to steal, or break into their protected areas. This, combined with using RFID which has ten to the thirty-second number of possible combinations, makes our system unique. There are a few RFID locks that you can buy, that range in price from about $200 to about $900[12]. Biometrics offers greater security and convenience than traditional identity authentication systems (based on passwords and cryptographic keys) since biometrics characteristics are inherently associated with a particular individual[18]. The problem with this, is that without RF technology, you would be able to used severed parts and dead bodies to open the lock, which could lead to violence.

B. Design Details

1) Cabinet Locker

a) Database

Software: MySQL
Outcome: Test of data feed from server.

b) Web Server

Software: A server
Outcome: Run website and all its functions.

c) Mobile Application

Hardware: iPhone
Software: Xcode
Outcome: Test of data feed from server.

d) Alarm

Hardware: Buzzer
Outcome: Arm alarm and attempt to set it off. When the alarm goes off, deactivate it using method such as sending message from mobile application or server, or timer.

2) Mobile Lock

a) Database

Software: MySQL
Outcome: Test of data feed from server.

b) Web Server

Software: A server
Outcome: Run website and all its functions.

c) Mobile Application

Hardware: iPhone
Software: Xcode
Outcome: Test of data feed from server.

d) Alarm

Hardware: Buzzer
Outcome: Arm alarm and attempt to set it off. When the alarm goes off, deactivate it using method such as sending message from mobile application or server, or timer.

3) Door Lock

a) Database

Software: MySQL
Outcome: Test of data feed from server.

b) Web Server

Software: A server
Outcome: Run website and all its functions.

c) Mobile Application

Hardware: iPhone
Software: Xcode
Outcome: Test of data feed from server.

4) All locks

a) Lock/Unlock Mechanism

Hardware: RFID Reader
RFID Reader Breakout
RFID Tag
Software: Integrated Development Environment (IDE)
Outcome: Reader and tag system that communicates with the lock and database.

III. FUNDING

Date	Item	Cost	Purchaser
10/9/2014	10 PCS 125kHz Keyfobs	4.54	Kevin
10/9/2014	The Complete iOS8 and Swift Course	49.00	Kevin
10/14/2014	Adafruit Wave Shield for Arduino Kit - v1.1	24.09	Kevin
10/21/2014	Adafruit Wave Shield for Arduino Kit - v1.1	22.99	Kevin
10/21/2014	100 PCS Stackable Headers	14.99	Kevin
11/1/2014	SainSmart Ethernet Shield W5100 for Arduino UNO	19.99	Kevin
2/2/2015	LilyPad Power Supply	79.81	Kevin
2/2/2015	XBee WiFi Module - PCB Antenna		Kevin
2/2/2015	LilyPad XBee		Kevin
2/2/2015	LilyPad Light Sensor		Kevin
2/7/2015	XBee Explorer Dongle	29.95	Kevin
2/9/2015	Zitrades Prototyping Prototype Shield	8.52	Kevin
2/11/2015	XBee WiFi Module - PCB Antenna	41.96	Kevin
2/27/2015	LilyPad Arduino 328 Main Board	15.72	Kevin
4/9/2015	Futaba S3114 Micro High Torque Servo	16.53	Kevin
4/9/2015	SainSmart XBee USB Adapter	18.34	Kevin
4/13/2015	KEEDOX® DC/DC Converter 12V Step Down to 5V	7.70	Kevin
4/16/2015	CC3000 WiFi Shield with Onboard Ceramic Antenna		Kevin
4/16/2015	3M Heavy Duty Mounting Tape, 1-Inch by 50-Inch	48.92	Kevin
4/26/2015	Futaba S3114 Micro High Torque Servo		Kevin
4/26/2015	CC3000 WiFi Shield with Onboard Ceramic Antenna	72.21	Kevin
10/9/2014	RFID Tags/Readers	77.75	Phillip
10/1/2014	Solderless Headers	6.93	Phillip
10/27/2014	Lock Hardware	30.99	Phillip
4/18/2015	12mm Dia 5 Pcs DC 5V 2 Terminals Electronic Continuous Sound Buzzer	5.07	Phillip
4/15/2015	CC3000 WiFi Shield with Onboard Ceramic Antenna	45.92	Phillip
4/11/2015	SainSmart Digital Analog Module V5 Sensor Shield	14.72	Phillip
4/11/2015	SunFounder 5V 1 Channel Solid State Relay Board	10.99	Phillip
4/1/2015	2x4 board, Door, Bottom Board for Door	60.00	Phillip
	Total Cost	736.11	

IV. PROJECT WORK BREAKDOWN STRUCTURE

A. Overview

Detail the tasks and detail responsibilities as well as estimated time of completion.

Level 0

Lock System

- The overall project. We are creating a locking system that will help keep the clients belongings safe and secure.
    - Who: Team2
    - Time Allotted: 15 weeks

Level 1

Cabinet Lock

- The built-in lock for cabinets.
- LEVEL 2 COMPONENTS

- Database
  - Store information given from the lock and alarm.
  - LEVEL 3 COMPONENTS
    - Design
    - Build structure and design layout.
      - Who: Adam Batakji
      - Time Allotted: 36 days
    - Data Communication
      - Send information to the app and the web server
        - Who: Adam Batakji
        - Time Allotted: 31 Days

- Web Server
  - Allow user to interact with database, and locking system through website.
  - LEVEL 3 COMPONENTS
    - Database Communication
      - Communicate with database, allow for client access and updating.
      - Who: Adam Batakji
      - Time Allotted: 31 days
    - Interface
      - Layout and design for client.
      - Who: Adam Batakji
      - Time Allotted: 66 days

- Mobile Application
  - Allow user to interact with database and locking system with phone application.
  - LEVEL 3 COMPONENTS
    - Database Communication
      - Communicate with database, allow for client access and updating.
      - Who: Kevin Nguyen
      - Time Allotted: 32 days
    - Interface
      - Layout and design for client.
      - Who: Kevin Nguyen
      - Time Allotted: 66 days

- Alarm
  - Warn the client when unauthorized access has taken place.
  - LEVEL 3 COMPONENTS
    - Database Communication
      - Communicate with database about the status of the alarm.
      - Who: Phillip Davis
      - Time Allotted: 31 days
    - Design
      - Procure hardware and code the parts together
      - Who: Phillip Davis
      - Time Allotted: 71 days

- Locking System
  - Actual locking system, including the code as well as the hardware.
  - LEVEL 3 COMPONENTS
    - Design
      - Hardware and parts as well as setup
      - Who: Joel Barrera
      - Time Allotted: 71 days
    - Permissions
      - Access allowances and communication code.
      - Who: Joel Barrera
      - Time Allotted: 49 days

Mobile Lock

- The portable lock for when the client is on the go and needs to be able to lock up their valuables. Useable where padlocks can be used.

- LEVEL 2 COMPONENTS
  - Database
    - Store information given from the lock and alarm.
    - LEVEL 3 COMPONENTS
      - Design
        - Build structure and design layout.
        - Who: Adam Batakji
        - Time Allotted: 36 days
      - Data Communication
        - Send information to the ap and the web server
        - Who: Adam Batakji
        - Time Allotted: 39 Days
  - Web Server
    - Allow user to interact with database, and locking system through website.
    - LEVEL 3 COMPONENTS
      - Database Communication
        - Communicate with database, allow for client access and updating.
        - Who: Adam Batakji
        - Time Allotted: 31 days
      - Interface
        - Layout and design for client.
        - Who: Adam Batakji
        - Time Allotted: 88 days
  - Mobile Application
    - Allow user to interact with database and locking system with phone application.
    - LEVEL 3 COMPONENTS
      - Database Communication
        - Communicate with database, allow for client access and updating.
        - Who: Kevin Nguyen
        - Time Allotted: 32 days
      - Interface
        - Layout and design for client.
        - Who: Kevin Nguyen
        - Time Allotted: 66 days
  - Alarm
    - Warn the client when unauthorized access has taken place.
    - LEVEL 3 COMPONENTS
      - Database Communication
        - Communicate with database about the status of the alarm.
        - Who: Phillip Davis
        - Time Allotted: 31 days
      - Design
        - Procure hardware and code the parts together
        - Who: Phillip Davis
        - Time Allotted: 71 days
  - Locking System
    - Actual locking system, including the code as well as the hardware.
    - LEVEL 3 COMPONENTS
      - Design
        - Hardware and parts as well as setup
        - Who: Joel Barrera
        - Time Allotted: 71 days
      - Permissions
        - Access allowances and communication code.
        - Who: Joel Barrera
        - Time Allotted: 44 days

Door Lock

- Lock for the clients door.
- LEVEL 2 COMPONENTS
  - Database
    - Store information given from the lock and alarm.
    - LEVEL 3 COMPONENTS
      - Design

- Build structure and design layout.
    - Who: Adam Batakji
    - Time Allotted: 36 days
- Data Communication
    - Send information to the app and the web server
    - Who: Adam Batakji
    - Time Allotted: 46 Days

- Web Server
    - Allow user to interact with database, and locking system through website.
    - LEVEL 3 COMPONENTS
        - Database Communication
            - Communicate with database, allow for client access and updating.
            - Who: Adam Batakji
            - Time Allotted: 31 days
        - Interface
            - Layout and design for client.
            - Who: Adam Batakji
            - Time Allotted: 88 days
- Mobile Application
    - Allow user to interact with database and locking system with phone application.
    - LEVEL 3 COMPONENTS
        - Database Communication
            - Communicate with database, allow for client access and updating.
            - Who: Kevin Nguyen
            - Time Allotted: 32 days
        - Interface
            - Layout and design for client.
            - Who: Kevin Nguyen
            - Time Allotted: 66 days
- Locking System
    - Actual locking system, including the code as well as the hardware.
    - LEVEL 3 COMPONENTS
        - Design
            - Hardware and parts as well as setup
            - Who: Joel Barrera
            - Time Allotted: 71 days
        - Permissions
            - Access allowances and communication code.
            - Who: Joel Barrera
            - Time Allotted: 31 days

<u>Documentation</u>

- Problem Statement: Document entailing the scope of our project and our societal problem.
    - Who: Team2
    - Time Allotted: 7 days

- Design Contract: Document detailing the hardware, software, as well as design of the project.
    - Who: Team2
    - Time Allotted: 14 days

- Work Breakdown Structure: A document detailing the task and breakdown of the project.
    - Who: Team2
    - Time Allotted: 7 days

- Weekly Reports: A weekly reports designed to document the teams progression.
    - Who: Team2
    - Time Allotted: Each Week 7 days

B. Conclusion

The work breakdown structure is meant to plan out the beginning steps of the project and to attempt to determine where more work, or help will be needed during the realization process. It is not necessarily the end to all work scheduling, but a step necessary in the planning process to attempt to stay on time. For the remaining eleven weeks Team 2 will be using this breakdown to deliver our project on time and in a completed fashion.

Some areas where we had to allow for modification in our structure are in our mobile application section and the web server section. The reason for the extra time for the mobile application was the simple fact that we needed to enroll in Apple's iOS Developer Program, which we would have to pay for if we wanted to test our application. We decided to wait until next semester to get enrolled in this program, as we will not be guaranteed that the license will still be valid by the time we need to present at the end of the second semester. Although we didn't get the opportunity to upload the application onto the mobile device, we managed to get the application to work as compensation. As for the web server, we needed to add extra time to this, as communication between the Arduino boards and the web server proved to be more complicated than we had anticipated. We managed to get the web server working correctly and even adding a few users to the database to make sure it was operational.

V. PROJECT TIMELINE

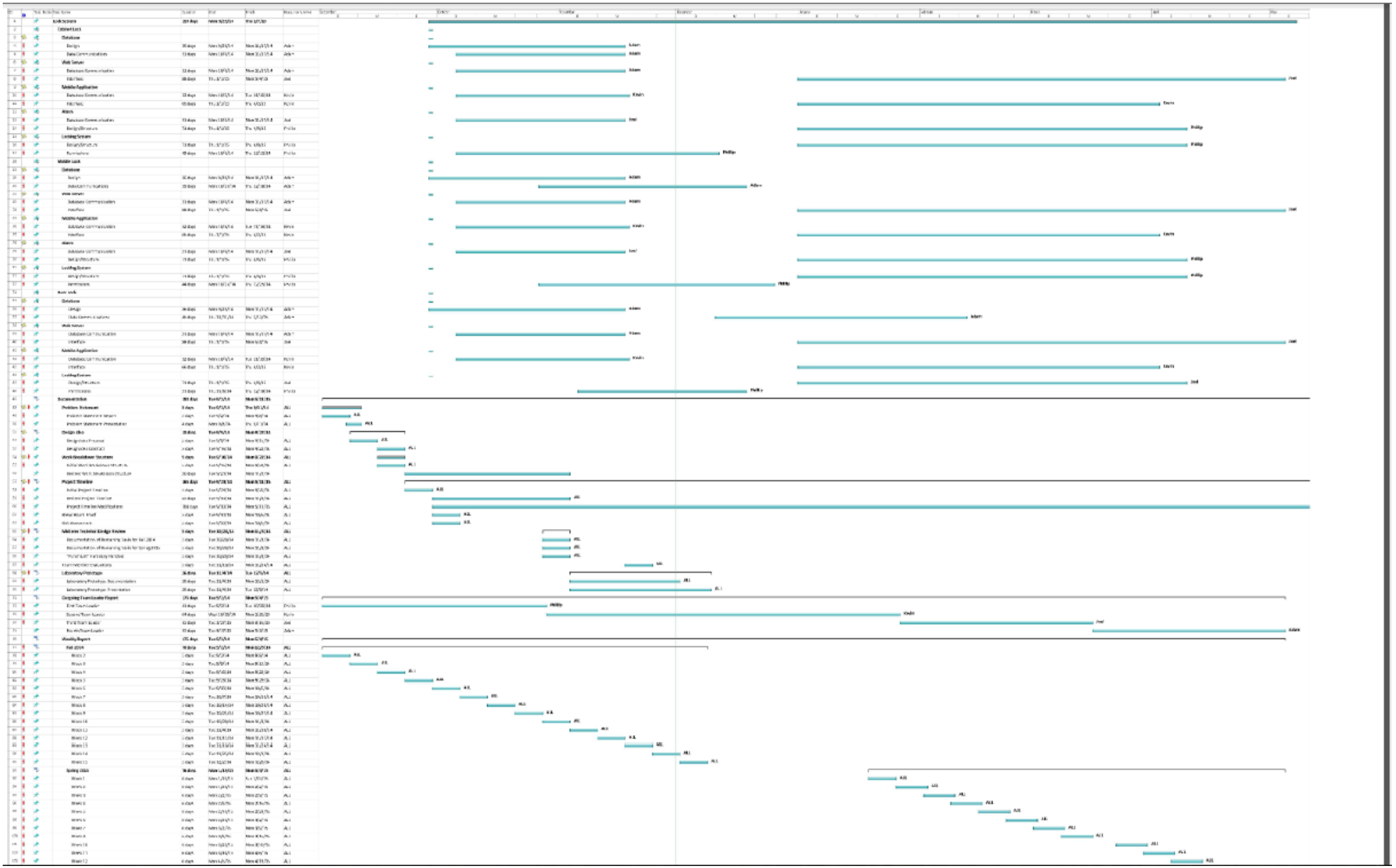


Fig. 7. Project timeline.

VI. Risk Assessment and Mitigation

		Impact				
		Minimum	Impact Can Be Tolerated	Limited Impact	May Jeopardize Project	Will Jeopardize Project
Probability	None	Clubs				
	Low Likelihood		HW From Other Classes		Device Malfunction	
	Likely		Work	Exams		Projects From Other Classes
	Highly Likely				Family Priorities	
	Near Certainty					

Fig. 8. Risk Assessment chart.

The Risk Assessment discusses the effects of an unwanted outcome, or a task not completed on time. Doing this allows the group to understand things that are in or out of our own control. This allows us to become more aware of possible hazards that can prove to be detrimental to the overall progress of the project and allows us to appropriately handle them with time. This also lets us plan out our work breakdown structure that can allow for the change in case of such occurrences.

Things that had the highest risk of impact on the project of course were things that were not in our control. These instances include situations like family emergencies, projects from other class, exams, and device malfunction. Other instances can be tolerated. These include getting sick, homework from other classes, and work. Luckily, our group of four is more than capable of carrying on the project in case any one of us goes missing or needs to attend a situation that could jeopardize the project.

In our work breakdown structure, areas that required modification included the mobile application section and web server section. The reason these areas needed modification were for reasons that were not in our control. The mobile application required to be enrolled in Apple's Developer Program, which we opted to wait until next semester to enroll in. As for the web server, it was due in some part due to hardware malfunction, as one of our routers were not working as intended and this wasn't apparent at first.

VII. WEEKLY REPORTS (FALL 2014)

A. Week 2

Team 2 Date: September 8, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	12
Adam Batakji	10
Joel Barrera	12
Kevin Nguyen	13
Total	47

Group Meeting Minutes and synopsis of past week:

September 2, 2013
Team 2 weekly lab period. Talked about what we wanted to do, as group members had been agreed upon the previous semester. Learned about what we were going to need to get started.

September 3, 2013
Team 2 members Kevin and Phillip meet to help decide what to do for senior project and ideas we can use to improve the project.

September 4, 2013
Team 2 members Joel, Adam, and Phillip meet for the changed lab time. Meet during the lab time to nail down everything we want to work on for the project, and also meet for a couple of hours after to get down who is going to write which part for the problem statement.

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Meeting(9/2/13)	3	Completed, 100%
Meeting (9/3/13)	2	Completed, 100%
Meeting (9/4/13)	3	Completed, 100%
Weekly Report (9/8)	1	Completed, 100%
Problem Statement	5	Completed, 100%
Total	14	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Elevator Pitch/Presentation	1.5	Completed, 100%
Feature/hardware/budget list -door	5	Completed, 100%
Feature/hardware/budget list -cabinet	5	Completed, 100%
Feature/hardware/budget list -locker	5	Completed, 100%
Resume/self eval	1.5	Completed, 100%
Weekly report (9/22)	1	Completed, 100%
Lab/Meetings	6	Completed, 100%
Total	25	Completed, 100%

Phillip Davis:

Synopsis of your past week's work:

Decided upon a societal problem to solve, and that I was going to be the first leader. Worked as a group to determine the first steps of our project and understanding the documentation process. My specific part of the Problem Statement was working on the conclusion as well as the cover page, abstract, and keywords.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab/Team meetings	8	Completed, 100%
Problem Statement-conclusion/abstract/cover	3	Completed, 100%
Weekly Report (9/8)	1	Completed, 100%
Total	12	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab/Team meetings	6	Ongoing
Elevator Pitch/Presentation	3	Completed, 100%
Weekly report (9/22)	1	Completed, 100%
Resume/self-evaluation	1.5	Completed, 100%
Total	10	Completed, 100%

Adam Batakji:
Synopsis of your past week's work:

Our team has been formed and a rough project idea was tackled by identifying the societal problem. Thousands of theft occurs every day regardless of the precaution people take. My part in this week's work was defining the problem in depth along with statistics and facts.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Introduction - defining the problem	3	Completed, 100%
Team Meeting	6	Completed, 100%
Weekly Report (9/8)	1	Completed, 100%
Total	10	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
feature/hardware/budget list -cabinet	5	Completed, 100%
Lab/Team meetings	6	Completed, 100%
Resume/self-evaluation	1.5	Completed, 100%
Weekly report (9/22)	1	Completed, 100%
Total	13.5	Completed, 100%

Special problems or other reporting information not included elsewhere:

This week has us working on the Design Idea Contract.

Joel Barrera:

Synopsis of your past week's work:

Grouped with the team and came together to have a better understanding of our project idea. Came to a decision and started to address the problem statement. For this assignment, we divided the problem statement into respective sections. My section for this week's work was describing the current solutions and their shortcomings as well as considering more efficient system that helps security and ultimately help protect our valuables.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Current Solutions and Their Shortcomings	3	Completed, 100%
Met With Team and Discussed Project Idea	6	Completed, 100%
Weekly Report (9/8)	1	Completed, 100%
Research and Write-Up	2	Completed, 100%
Total	12	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Feature/hardware/budget list -door	5	Completed, 100%
Lab/Team meetings	6	Completed, 100%
Resume/self-evaluation	1.5	Completed, 100%
Weekly report (9/22)	1	Completed, 100%
Total	13.5	Completed, 100%

Kevin Nguyen:

Synopsis of your past week's work:

This week, I met with my team and decided on our problem statement. Sections of the problem statement was divided and assigned to a team member to complete. I was assigned to research and write about theft and how it affects society.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meeting (Problem Statement) - 9/2	3	Completed, 100%
Team Meeting (Problem Statement, cont.) - 9/3	2	Completed, 100%
Research/write-up on how problem affects society	8	Completed, 100%
Total	13	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab/Team meetings	6	Completed, 100%
Feature/hardware/budget list -locker	5	Completed, 100%
Weekly report (9/22)	1	Completed, 100%
Resume/self-evaluation	1.5	Completed, 100%
Total	13.5	Completed, 100%

B. Week 3

Team 2 Date: September 15, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	12
Adam Batakji	14
Joel Barrera	12
Kevin Nguyen	11
Total	49

Group Meeting Minutes and synopsis of past week:

September 9, 2014:
Team 2 meeting to get the presentation details together, decided who would do which part of the design idea documentation.

September 11, 2014:
Team 2 weekly lab period. Talked about design ideas and met with the teacher to decide where to go from this point and what to be thinking about for next week. As well as deciding on what the work was going to be for the weekend.

Synopsis:
Team 2 worked on creating a Design Idea Contract for our Problem Statement that we had from last week. We decided upon the features we would like to implement as well as some of the hardware, and software that we would like to use to accomplish this. We also worked on our presentation for our Problem Statement and had our advisor Dennis Dhalquist tell us what we needed to work on for next week.

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Problem Statement Presentation	4	Completed, 100%
Design Idea Documentation	25	Completed, 100%
Weekly report	4	Completed, 100%
Total	49	Completed, 100%

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	16	Completed, 100%
Design Idea Contract	12	Completed, 100%
Work Breakdown Structure	14	Completed, 100%
Weekly Report	4	Completed, 100%
Total	46	Completed, 100%

Phillip Davis:

As a group we worked on the Design Idea Contract. I have called it documentation, because it is not a contract until it is approved. Worked on the presentation, as well as working on the design idea documentation and brainstorming ideas for our problems solution. Did part of the work breakdown structure and worked on answering the questions for the report.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Design Idea Documentation	6	Completed, 100%
Problem Statement Presentation	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	12	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Work Breakdown Structure	3	Completed, 100%
Design Idea Contract	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	11	Completed, 100%

Adam Batakji:

The scope of last week's work was creating a design proposal contract. This is practically our answer, or solution to the problem statement. In brief, the problem is theft and the solution is to reduce the occurrence of it. On my part, I contributed toward the design overview of what our project will demonstrate along with some features with their specifications, as well as the research on hardware and software necessary to implement these features. I also worked on the introduction.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Design Idea Documentation	6	Completed, 100%
Software/Hardware Specifications	3	Completed, 100%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	14	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Work Breakdown Structure	4	Completed, 100%
Design Idea Contract	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	13	Completed, 100%

Next week will be busy as we establish a work breakdown between the team and revise the Design Contract.

Joel Barrera:
Coming together as a team to create a design proposal agreement has been the emphasis of last week. In this contract, we state what we will be implementing into our design, and like Adam stated, is our answer to last weeks emphasis (problem statement). This week I contributed to the contract design ideas, what we could use when it comes to hardware and software.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Design Contract Ideas	6	Completed, 100%
Problem Statement Presentation	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	12	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Work Breakdown Structure	4	Completed, 100%
Start Research/Study for Apps/Software	2	15%
Design Idea Contract	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	13	Completed, 100%

Special problems or other reporting information not included elsewhere:

Kevin Nguyen:

The initial Design Idea Contract was created in the past week. Parts were researched for the door lock. The formatting for done for the Design Idea Contract. Also worked on references and resume, as well as hardware list.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Design Idea Documentation	5	Completed, 100%
Problem Statement Presentation	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	11	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Work Breakdown Structure	3	Completed, 100%
Design Idea Contract	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	11	Completed, 100%

C. Week 4

Team 2 Date: September 22, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	13
Adam Batakji	15
Joel Barrera	10
Kevin Nguyen	14
Total	52

Group Meeting Minutes and synopsis of past week:
Team 2 worked on the Design Idea Contract. We solidified our contract and changed it so that it was a bit more general in the hardware and software areas and a bit more professional. We then worked on the Work Breakdown Structure. The breakdown structure was broken into its features and each member worked on their own features. We thought that we had to finish the timeline as well, and it was a part of our week.

September 16, 2014
Team 2 had an online meeting to test the online meeting equipment and to go over strategy for future meetings and ways that members would be able to work together and give advice/opinions.

September 18, 2014:
Team 2 decided on who was doing which part for the breakdown, and in a little confusion decided to start work on the timeline for the board as well. We met with our advisor and went over the changes that were going to be necessary for the Design Idea Contract as well as information on the Work Breakdown Structure.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Design idea Contract revisions	9	Completed, 100%
Work Breakdown Structure	18	Completed, 100%
Weekly report	4	Completed, 100%
Timeline	4	60%
Research	1	5%
Total	52	Completed, 100%

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	24	Completed, 100%
Bulletin Board	7	30%
Project Timeline	7	Completed, 100%
Weekly Report	4	Completed, 100%
Total	42	

Phillip Davis:

Made the changes to the design idea contract that were suggested by our advisor, including changing the wording that was used, making the report more professional. Also created the work breakdown structure document, and did the weekly report.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Design Idea Contract Revision	2	Completed, 100%
Work Breakdown Structure Document	6	Completed, 100%
Weekly Report	1	Completed, 100%
Total	13	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Project Timeline	1	Completed, 100%
Bulletin Board	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total		Completed, 100%

Adam Batakji:

This was a strenuous week as our team came to an agreement for our terms in the design contract. Our project advisor gave us some ideas on what to fix as we were too specific in some of our proposals. Obviously, it is not a good idea to lock ourselves in this early with specifics since we have not started yet. Other than the design contract, the work breakdown structure was a useful exercise to help break apart the tasks it will take to design each feature.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Design Idea Contract Revisions	2	Completed, 100%
Database breakdown structure	4	Completed, 100%
Web server breakdown structure	4	Completed, 100%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	15	Completed, 100%

On a broad scale, next week's tasks consist of starting the bulletin board and creating a project timeline.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Timeline (Database, Web Server)	3	Completed, 100%
Bulletin Board (Database, Web Server)	3	Completed, 100%
Weekly Report	1	Completed, 100%
Total	11	Completed, 100%

Joel Barrera:

This week was definitely an eye opener. Our contract design had the right idea, at least in our minds. After meeting with our advisor, we noticed how critical the design idea contract is. As a team, we had a general idea of what our project was going to address, but we weren't completely on the same page. With the contract, we managed to divide our project into it's important components and get everyones ideas solidified cohesively. Also, we were being too specific when it came to details, this in turn narrowed our options, which is something we do not want.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Design Idea Contract (Revisions)	2	Completed, 100%
Research Servers	1	Completed, 100%
Weekly Report	1	Completed, 100%
Revise Breakdown Structures	2	Completed, 100%
Total	10	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Timelines (All)	2	Completed, 100%
Bulletin Board	1	30%
Weekly Report	1	Completed, 100%
Total	10	Completed, 100%

Special problems or other reporting information not included elsewhere:

Kevin Nguyen:

This week, our team sat down with our advisor and discussed our Design Contract Proposal. Issues with our contract was noted and corrected in our final Design Idea Contract. The Work Breakdown Structure was also completed this week. I completed the work breakdown for the Mobile Application. I also finalized the formatting for the Design Idea Contract and created the LockSys Project file in Microsoft Project.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Design Idea Contract	3	Completed, 100%
Mobile Application Work Breakdown Structure	2	Completed, 100%
Timeline	4	60%
Total	14	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Bulletin Board	2	30%
Timeline	1	Completed, 100%
Total	9	Completed, 100%

D. Week 5

Team 2 Date: September 29, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	7
Adam Batakji	11
Joel Barrera	10
Kevin Nguyen	9
Total	**37**

Group Meeting Minutes and synopsis of past week:
Team2 created a timeline for our project using our work breakdown contract. We also decided upon what we wanted to start the bulletin board with and started getting our equipment in the mail and testing it.

September 22 , 2014:

An RFID reader, tag, and breakout board were bought and arrived in the mail, so Team 2 worked on figuring the configuration out as well as beginning the testing of the equipment, as well as beginning breadboard proof and talking about what we want to show.

September 25 , 2014:

Met with the advisor on our project. Asked questions and split up the parts for the timeline. Also talked as a team about adding time for research on our project and its deliverables.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Bulletin Board	2	Completed, 100%
Weekly report	4	Completed, 100%
Project Timeline	8	60%
Research	7	10%
Total	**37**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	24	Completed, 100%
Breadboard Proof	4	Completed, 100%
Breadboard Proof Testing	25	Completed, 100%
Weekly Report	4	Completed, 100%
Total	**57**	

Phillip Davis:

Worked on filling in the timeline to turn it in and assigning people to their parts of the project. Also decided what to put on the bulletin board, and worked on research for the RFID board and tags that we are currently testing.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Timeline naming and filing	1	Completed, 100%
Research	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**7**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Soldering research/practice	3	Completed, 100%
Breadboard Proof Testing	5	Completed, 100%
Breadboard Proof	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**15**	

Adam Batakji:

With a design contract and work breakdown established, our work on the project is just about ready to begin. A timeline was established to put all the activities and their subtasks in order. The bulletin board has been loaded with our picture and basic idea of our project. The breadboard proof will be soon which is the queue to get to work.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Timeline (Database, Web Server)	3	Completed, 100%
Bulletin Board (Database, Web Server)	1	Completed, 100%
Research web servers	2	Completed, 100%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**11**	

Next week will mainly consist of structure models for the database to foresee the database in the right perspective.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Discover normalization for database	2	Completed, 100%
Entity-Relationship Model	2	Completed, 100%
Relational Data Model	2	50%
Weekly Report	1	Completed, 100%
Breadboard Proof	1	Completed, 100%
Total	**11**	

Joel Barrera:

After making our adjustments to our design contract, it was time to take a step forward and finish most of the paperwork for this semester. We created a timeline in order to keep better track of our progress and to better organize what needed to be addressed first. We also started on our bulletin board, which of course we will continue to update as the semester advances. This week we will focus on getting our hands dirty.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Timeline (Start and End Dates)	2	Completed, 100%
Bulletin Board	1	Completed, 100%
Weekly Report	1	Completed, 100%
Research (PHP and Objective-C)	2	Completed, 100%
Total	**10**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Breadboard Proof	1	Completed, 100%
Breadboard Proof Testing	8	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**13**	

Kevin Nguyen:
This week, I completed the documentation section of the Project Timeline. Dates and duration of each written assignment through the end of Fall 2014 have been added. Spring 2015 assignment dates will be added once the syllabus is released. The RFID Reader ID-20LA arrived and time was spent reading through the datasheet. Further research with its implementation with the Arduino was done in preparation for the Breadboard Proof.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Timeline	2	Completed, 100%
Research	2	Completed, 100%
Total	**9**	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Breadboard Proof Testing	5	Completed, 100%
Breadboard Proof	1	Completed, 100%
Total	**12**	Completed, 100%

E. Week 6

Team 2 Date: October 6, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	10
Adam Batakji	13
Joel Barrera	10
Kevin Nguyen	13
Total	**46**

Group Meeting Minutes and synopsis of past week:
Team 2 worked on printing things for the bulletin board as well as starting research for certain aspects of the project such as metal on RFID and database information. We also worked on the report and getting parts for the breadboard proof.

September 30, 2014:
Talked about what was going to be due next week and responsibilities for them for each team member, as well as talk about things needed to research.

October 2, 2014:
Team 2 met to talk about what we wanted to show for the breadboard proof. We also talked about what we should work on next for the project, and who was going to be the next leader for the project as well as what we needed to have done for Monday's report.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Bulletin Board	2	Completed, 100%
Weekly report	4	Completed, 100%
Breadboard Proof Work	15	Completed, 100%
Research	7	10%
Web Server Work	2	2.5%
Total	**46**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	16	Completed, 100%
Research	9	30%
Database work	10	30%
Server Work	1	5%
Mobile Ap Work	5	2.5%
Weekly Report	4	Completed, 100%
Total	**45**	

Phillip Davis:

This week was our first week after launch. We had some of our parts come in and we proceeded testing to see if we could read through metal with the low frequency RFID, and since we couldn't we had to decide upon a little bit of a different design that still fits in our design contract. We also worked on the breadboard proof and figuring out what the teacher wanted to see as well as making an overly huge poster for our board.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Breadboard Proof Testing	3	Completed, 100%
Soldering	1	Completed, 100%
Breadboard Proof	1	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**10**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	4	Completed, 100%
Mobile padlock research	1	10%
Part research and acquisition	2	30%
Door Lock Hardware Start	2	30%
Weekly Report	1	Completed, 100%
Total	**10**	

Adam Batakji:

Another week of progress has been put in the books. My contributions consisted of drawing design models for the database. It is much more efficient to create the tables based off of the models to stay consistent. The entity-relationship model distinguished between the end users and moderators along with the type of data that will be collected. The web server was also established as XAMPP now runs on my laptop. XAMPP is a full package that includes HTML, Apache Web Server, MySQL, PHP and Perl.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Discover normalization for database	2	Completed, 100%
Entity-Relationship Model	2	Completed, 100%
Relational Data Model	2	75%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Establishing web server	2	Completed, 75%
Total	**13**	

More models for the database will be realized for efficiency. If time permits, some HTML coding may begin to create a homepage for the server.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Design Schema Diagram	2	Completed, 100%
Three-Schema Architecture	2	Completed, 100%
Relational Data Model	1	Completed, 100%
Database Tables	2	50%
Weekly Report	1	Completed, 100%
Establishing web server	1	Completed, 100%
Breadboard Proof	1	Completed, 100%
Total	**14**	

Joel Barrera:

This week has been included a lot of research outside of class. Mainly to get more acquainted with PHP, Objective-C, JavaScript as well as HTML. This information will come in handy when we start coding more throughout the semester. This week also included taking pictures for our bulletin board, it does not look so plain anymore.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Bread Board Proof	1	Completed, 100%
Bread Board Proof Testing	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**10**	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Help with database and server	3	Completed, 100%
Research on RFID and metal	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**10**	Completed, 100%

Kevin Nguyen:
This week, the RFID reader component of the project was selected for demonstration for the breadboard proof. Phillip acquired the RFID Reader ID-20LA and the breakout board. We went to the soldering lab and soldered the reader onto the breakout board. The reader was then interfaced with the Arduino to test its functionality. I also designed our Team 2 poster that is on our bulletin board. Photos were also added on there as well.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Breadboard Proof Testing	5	Completed, 100%
Breadboard Proof	1	Completed, 100%
Bulletin Board	1	Completed, 100%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Research and resource gathering for mobile app	3	50%
Begin programming	1	1%
Total	**10**	

F. Week 7

Team 2 Date: October 13, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	10
Adam Batakji	15
Joel Barrera	10
Kevin Nguyen	10
Total	**45**

Group Meeting Minutes and synopsis of past week:

Team 2 worked on obtaining parts for the breadboard proof as well as the next steps in our building process. Since we have the RFID and it works with our LCD and a lock that we have tested, we will begin ordering all RFID parts. There is also more work going on for the server and the database as well as the beginning of programming a mobile application.

October 7, 2014:

Met with group about breadboard proof and getting all the information that might be asked about on the proof, including research of the difference between a FOB and RFID as well as some of the specs of our equipment.

October 9, 2014:

Met with our advisor and had a breadboard proof. After we had a meeting to decide where to go next, and talk about the parts that we have already done.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Research	9	20%
Database work	10	30%
Server Work	1	1%
Mobile App Work	5	2.5%
Weekly Report	4	Completed, 100%
Total	**45**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	16	Completed, 100%
Research	7	30%
Web Server Work	8	15%
Mobile Ap Work	6	7.5%
Hardware	4	5%
Weekly Report	4	Completed, 100%
Total	**45**	

Phillip Davis:

This week was important for getting our part list together, and beginning research for what I assume is the hardest part of our project, the mobile lock. I also worked on the breadboard proof presentation as well as buying the next round of items to continue hardware growth.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Mobile Padlock Research	1	Completed, 100%
Part research and acquisition	2	70%
Door Lock Hardware Start	2	30%
Weekly Report	1	Completed, 100%
Total	**10**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	4	Completed, 100%
Initial Door Lock building(hardware)	4	10%
Cabinet research locking systems	2	50%
3D printing research	1	5%
Weekly Report	1	Completed, 100%
Total	**12**	

Adam Batakji:

More diagrams and models were designed this week to further landscape the database. As the designs come together, it shows how helpful they will be in creating the database in a Server/Client environment. The web server is fully up now and ready to run PHP/HTML code in the localhost. The breadboard proof was a success as the RFID was demonstrated along with some other components.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Relational Data Model	2	Completed, 100%
Establishing web server	2	Completed, 100%
Design Schema Diagram	2	Completed, 100%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Three-Schema Architecture	2	75%
Database Tables	2	50%
Total	**15**	

Next week's major focus will be on the web server.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Three-Schema Architecture	2	Completed, 100%
Web pages	5	15%
Research serial communication	2	30%
Database Tables	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**16**	

Joel Barrera:

A lot of research was done in terms of coding. I wanted to primarily get acquainted with PHP, Objective-C, JavaScript and HTML as these will be some of the building blocks with the web server and database. Something I was also looking into was the possibility of getting past the metal, with stronger tags and readers at first, but RFID On Metal (ROM) came up that could be a better alternative.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Help with database and server	3	25%
Research on RFID and metal	2	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**10**	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
RFID On Metal (ROM) Research	2	Completed, 100%
HTML code for website	3	10%
Weekly Report	1	Completed, 100%
Total	**10**	Completed, 100%

Kevin Nguyen:

This week, I began acquiring resources necessary to develop an iOS app for our project. Attempts were made to enroll into the iOS Developer Program, but were unsuccessful as Tatro and I ran into technical difficulties. For the time being, I am able to use the Xcode software to program iOS apps and test it through its built-in simulator. The program enrollment will become necessary later on when I need to load an app onto an iOS device for testing. I have enrolled in Udemy's online iOS8 and Swift Course to retouch on iOS's programming language and the Xcode software.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Research & resource gathering for mobile app (iOS8 and Swift Course Lectures 13/148)	3	50%
Begin programming	1	1%
Total	**10**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 1-4)	5	Completed, 100%
iOS App Development	1	5%
Total	**12**	

G. Week 8

Team 2 Date: October 20, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	12
Adam Batakji	15
Joel Barrera	11
Kevin Nguyen	12
Total	**50**

Group Meeting Minutes and synopsis of past week:

Team 2 worked on advancing our knowledge that we will need to build everything properly, as well as continuing current portions of the project including the web server and database. We procured more RFID tags and readers as well as door parts to start on the next step after the breadboard.

October 14, 2014:

Talked about what we wanted to do for the week and the next part we are going to work on. Short meeting to check on other members and mindset.

October 16, 2014:

Conversed about electric door strikes vs solenoids and other options that are available to implement our design idea. Also talked about what we had worked on since the last meeting and where we wanted to go next.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Research	7	30%
Hardware	4	5%
Web Server Work	8	15%
Mobile App Work	6	7.5%
Database Work	5	50%
Weekly Report	4	Completed, 100%
Total	**50**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	17	Completed, 100%
Research	5	55%
Web Server Work	9	35%
Mobile App Work	8	10%
Hardware	11	8.5%
Weekly Report	4	Completed, 100%
Total	**54**	

Phillip Davis:

This week I worked did research on cabinet locks and our options for going with electrical strikes or using solenoids. I also ordered more parts for the door and the cabinet as well as the padlock that arrived yesterday (10/19). I also researched into the options and the steps necessary for using the 3d printer. Also started the hardware for the door lock. We have an electric strike for the door and I worked on getting information on how it works and how we are going to implement it in our system.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	4	Completed, 100%
Initial Door Lock building(hardware)	4	10%
Cabinet research locking systems	2	80%
3D printing research	1	5%
Weekly Report	1	Completed, 100%
Total	**12**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	4	Completed, 100%
Door Lock Building	3	25%
Wave Shield Soldering	2	Completed 100%
RFID soldering	2	Completed 100%
Weekly Report	1	Completed, 100%
Total	**12**	

Adam Batakji:
Lots of new things were learned this week as I coded PHP/HTML scripts. It has become apparent how much potential PHP scripts have. So far, a home page has been created to access the information that will come from the database. My goal for this week is to have a login page that allows privileged users stored in the database permission to log in.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Web pages	5	15%
Research serial communication	2	30%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Three-Schema Architecture	2	100%
Database Tables	2	100%
Total	**15**	

The web server will continue to be the main focus.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Web pages	7	35%
Research serial communication	2	50%
Weekly Report	1	Completed, 100%
Total	**14**	

Joel Barrera:

Last week, I found useful research that could bypass our dilemma of the RFID tag and readers communicating through metal. There are specific tags and readers we could option for, but of course they would come at a much higher cost, also replacing what we have already. As for this coming week, my emphasis will shift towards the locking mechanism we will use for the doors.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
RFID On Metal (ROM) Research	2	Completed, 100%
HTML code for website	3	10%
Weekly Report	1	Completed, 100%
Total	**10**	Completed, 100%

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Web Page	3	35%
Research Implementation of Electronic Lock	3	90%
Weekly Report	1	Completed, 100%
Total	**11**	Completed, 100%

Kevin Nguyen:
This week, I ordered an Adafruit Wave Shield Kit for the Arduino for the alarm component of the project. Assembly will be required (soldering) when it arrives in several days. I have also completed Sections 1-4 of the iOS8 and Swift Course. Basic introduction to Xcode and to Swift code have been covered. A basic example program was made to perform mathematical operations on a number provided by the user to display the results on screen.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 1-4)	5	Completed, 100%
iOS App Development	1	5%
Total	**12**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 5-6)	6	Completed, 100%
iOS App Development	1	10%
Wave Shield Soldering	2	Completed, 100%
Total	**15**	

H. Week 9

Team 2 Date: October 27, 2014

Current Team Leader: Phillip Davis

Team Hour Summary:

Team Member	Hours
Phillip Davis	12
Adam Batakji	14
Joel Barrera	11
Kevin Nguyen	15
Total	**57**

Group Meeting Minutes and synopsis of past week:
Team 2 continued with our development with the project. We worked more on the web server as well as the database and soldering for our hardware gadgets. We also

October 21, 2014:
Team 2 had a short meeting to discuss what was necessary for the upcoming thursday, as well as any questions we had on what was expected to be turned in for the coming week.
October 23, 2014:
Team 2 met to talk about how we want to format meetings in the future as well as updating each other on our progress on the project. We also conversed about technical parts of the project including how web server communication works and some of the problems we were facing including having problems obtaining developer permissions. Met with our teacher about this problem and continued work on studying for when we do get said permission.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	16	Completed, 100%
Research	11	55%
Web Server Work	10	35%
Mobile Ap Work	7	10%
Hardware	9	8.5%
Weekly Report	4	Completed, 100%
Total	**57**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	16	Completed, 100%
Research	5	60%
Web Server Work	10	35%
Mobile Ap Work	7	15%
Hardware	9	12.5%
Weekly Report	4	Completed, 100%
Total	**51**	

Phillip Davis:

This week Kevin and I worked on soldering the Adafruit Wave Shield Kit for our Arduino. I also worked on coding for the hardware on the door lock and ordered breakout boards that I realized I had forgotten when all of the other pieces of our project came in the mail.

Tasks Assigned – Last Week:

Task	Hours	Projected % Completion
Lab meetings/class	4	Completed, 100%
Door Lock Building	3	25%
Wave Shield Soldering	2	Completed 100%
RFID soldering	2	Completed 100%
Weekly Report	1	Completed, 100%
Total	**12**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	4	Completed, 100%
Door Lock Building	2	30%
2nd Wave Shield Soldering	2	Completed 100%
2nd and 3rd RFID soldering	2	Completed 100%
Weekly Report	1	Completed, 100%
Total	**11**	

Adam Batakji:

The web server has proven to be much work the deeper I delve into it. So far, a basic homepage has been created with links to other pages which have yet to be coded. The login page and register page is currently in progress. Once that is complete, the next step is to execute serial communication between the arduino and web server.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Web pages	7	35%
Research serial communication	2	50%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**14**	

This week's focus is serial communication.

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meeting	4	Completed, 100%
Web pages	7	45%
Research serial communication	2	65%
Weekly Report	1	Completed, 100%
Total	**14**	

Joel Barrera:

This week has been kind of hectic with midterms slowly taking over our lives. We managed to get together as a group and discuss important group issues that we believed would benefit us in the long run. We plan for more thorough meetings as well as begin incorporating all of our individual work into the project as a whole, to begin to see the big picture.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Team Meetings	4	Completed, 100%
Web Page	3	35%
Research Implementation of Electronic Lock	3	90%
Weekly Report	1	Completed, 100%
Total	**11**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Team Meetings	4	Completed, 100%
Web Page	2	45%
Research for 3D Printing for Team2	3	15%
Weekly Report	1	Completed, 100%
Padlock Building	3	10%
Total	**13**	

Kevin Nguyen:

This week, a second Adafruit Wave Shield Kit for the Arduino ordered. Soldering was completed for the first Shield Kit and soldering for the second kit will be done this week. Sections 5-6 of the iOS8 and Swift Course was completed. Navigation, storage, live content, as well as working with media in swift code have been covered.

Tasks Assigned – Last Week:

Task	Hours	Status and % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 5-6)	6	Completed, 100%
iOS App Development	1	10%
Wave Shield Soldering	2	Completed, 100%
Total	**15**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 7-8)	7	Completed, 100%
iOS App Development	1	12.5%
Second Wave Shield Soldering	2	Completed, 100%
Total	**16**	

I. Week 10

Team 2 Date: November 3, 2014

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	11
Adam Batakji	14
Joel Barrera	13
Kevin Nguyen	14
Total	**52**

Group Meeting Minutes and synopsis of past week:

October 30, 2014:

Team 2 met to talk about our progress on the project. The Wave Shield was not functioning as expected. SD Card I/O errors were received. After inspection, we noticed that five connections were skipped during our initial soldering. After that was fixed, the SD card was recognized, but could not be read. Our team advisor helped us correctly format the card, but there were still issues with audio playback. The serial monitor indicates that the audio files are in fact being played, but audio could not be heard through the headphones connected to the audio jack. The purpose of the Wave Shield was to play a loud alert audio in the events of a break-in or multiple failed access attempts. For now, we have agreed to set the Wave Shield aside and settle with our working Piezo Buzzer for the upcoming Midterm Technical Review.

The web pages, databases, and iOS app are coming along well. Serial communication between the server and Arduino is the goal for this week. Once that fundamental aspect of the project is completed, we will have our "project" for the Midterm Technical Review.

Synopsis:

Summary of Team Activities:

Task	**Task Hours This Week**	**Task Status**
Lab meetings/class	16	Completed, 100%
Research	5	60%
Web Server Work	9	45%
Mobile App Work	6	15%
Hardware	12	15%
Weekly Report	4	Completed, 100%
Total	**52**	

Forecast tasks for next week:

Task	**Task Hour Forecast**	**Task Status Forecast**
Lab meetings/class	16	Completed, 100%
Research	5	75%
Web Server Work	10	50%
Mobile App Work	3	25%
Hardware	16	25%
Weekly Report	4	Completed, 100%
Total	**54**	

Phillip Davis:

Ordered the breakout board for our other two locking systems. Also helped Kevin with attempting to re-solder the board for the Wave Shield since we were/still are having problems with receiving sound from the card. Need to finish soldering the other boards and work on getting the hardware together since the midterm is next week. Also need to help with understanding internet problems at Sac State, and our options for the midterm.

Tasks Assigned – Last Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Door Lock Building	4	30%
2nd Wave Shield Soldering	2	Completed 100%
Weekly Report	1	Completed, 100%
Total	**11**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Soldering 2nd and 3rd RFID	2	Completed, 100%
Cabinet Lock Building	3	50%
Padlock Building	3	30%
Total	**13**	

Adam Batakji:

As week 10 arrives, the realization of how much work has been put into this so far is strong. Feelings of exhaustion and frustration are starting to arrive as the other classes decided it would be a good idea to assign a project as well. Putting complaints aside, a user can now register and login to the homepage with success. For now, there is no security on the page as the goal is to have everything function first. Serial communication is running behind, but it should be on its way soon.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Web pages	7	45%
Research serial communication	2	65%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**14**	

The focus is serial communication again for this week.

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meeting	4	Completed, 100%
Web pages	7	60%
Research serial communication	2	85%
Implement serial communication	2	30%
Weekly Report	1	Completed, 100%
Total	**16**	

Joel Barrera:

Week 10 is here and the technical review is getting even closer. With that said, we still have a lot of work to do. We need to get everything assembled together and hopefully get it ready for presentation. We need to create the padlock we will be using. It won't be the prettiest, but it will hopefully get the job done.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Team Meetings	4	Completed, 100%
Padlock Building	3	10%
Web Page	2	60%
Weekly Report	1	Completed, 100%
Research for 3D Printing for Team2	3	40%
Total	**13**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	10%
Research for 3D Printing for Team2	3	45%
Weekly Report	1	Completed, 100%
Padlock Building	3	20%
Total	**15**	

Kevin Nguyen:

An Ethernet Shield for the Arduino has been ordered. Technical difficulties arose when trying to test the Wave Shield. The serial monitor shows that audio files are being played, but the actual audio is not heard through the headphone jack. Sections 7-8 of the iOS8 and Swift Course was completed. Working with core data was introduced and an exercise has been completed to create an Instagram clone.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS8 and Swift Course (Sections 7-8)	5	Completed, 100%
iOS App Development	1	12.5%
Second Wave Shield Soldering	2	Completed, 100%
Total	**14**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Research iOS Database Communications	2	Completed, 100%
iOS App Development	1	25%
Connect Arduino Onto Local Network	4	Completed, 100%
Total	**13**	

J. Week 11

Team 2 Date: November 10, 2014

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	15
Adam Batakji	16
Joel Barrera	13
Kevin Nguyen	11
Total	**55**

Group Meeting Minutes and synopsis of past week:

November 6, 2014:

Team 2 met to talk about our progress on the project. We discussed several possible check-out methods for our door lock system. It was decided that once the RFID reader reads a tag with appropriate permissions, the Arduino will send a signal to the relay that will allow the 5V power supply to power the electric door strike for a set amount of time. This will allow the door strike to unlock its latch, allowing a locked door to simply push past the latch to open the door. It will then lock itself after the set amount of time. From the inside, the user can always turn the doorknob and let themselves out without the need of triggering the door strike. This solves the issue of a possible power outage that would otherwise keep everyone locked inside. On the outside, the door will simply have a handlebar or lever to pull on to open the door once access is granted.

We have connected the Arduino onto the university's network and have a simple web server program loaded that displays raw analog data from the Arduino's pins. We are currently trying to have the web page display RDIF tag IDs as they are scanned, but are coming across issues as the Arduino cannot multitask (running web server and actively reading tags at the same time). Other possible methods to send the tag data is being explored.

Synopsis:

Summary of Team Activities:

Task	**Task Hours This Week**	**Task Status**
Lab meetings/class	16	Completed, 100%
Research	6	75%
Web Server Work	10	50%
Mobile App Work	3	25%
Hardware	16	25%
Weekly Report	4	Completed, 100%
Total	**55**	

Forecast tasks for next week:

Task	**Task Hour Forecast**	**Task Status Forecast**
Lab meetings/class	16	Completed, 100%
Research	5	90%
Web Server Work	10	75%
Mobile App Work	3	35%
Hardware	16	50%
Weekly Report	4	Completed, 100%
Total	**54**	

Phillip Davis:

Soldered the last two RFIDs to their boards. Tested them to make sure they work the same as the first RFID. Decided during meeting how to implement check-in/out system, and started code. Also worked on code for cabinet lock and assisted in the first stages of communication between Arduino and our web server.

Tasks Assigned – Last Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Soldering 2nd and 3rd RFID	2	Completed, 100%
Cabinet Lock Building	3	50%
Padlock Building	3	30%
Total	**13**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Cabinet Lock Building	5	75%
Padlock Building	5	50%
Total	**17**	

Adam Batakji:

Good progress was made with serial communication this past week. Turns out the data will be sending data to the web server through PHP. Originally, it was being considered to be java-based, but that became more complicated than need be. The key data will be the identification number the RFID sends when reading the card. That data will then be time stamped along with the status of the door becoming unlocked until a timer goes off.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Web pages	7	60%
Research serial communication	2	85%
Implement serial communication	2	30%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**16**	

The current focus is sending data in the correct form to the web server.

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meeting	4	Completed, 100%
Web pages	7	80%
Research serial communication	2	100%
Implement serial communication	2	60%
Weekly Report	1	Completed, 100%
Total	**16**	

Joel Barrera:

The Midterm Technical Review is around the corner (next week to be exact) and we are approaching the final stages of our design. Although we still need a lot of work to do, I think everything is coming together nicely. Last week I invested time into looking at AutoCAD and how we can use it to create a padlock design, in order to use 3D printing. Next week will be very similar, as having little to no prior skills in AutoCAD makes it difficult to progress, especially when things new-user friendly.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	10%
Research for 3D Printing for Team2	3	45%
Weekly Report	1	Completed, 100%
Padlock Building	3	20%
Total	**15**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	40%
Research for 3D Printing for Team2	1	65%
Weekly Report	1	Completed, 100%
Padlock Building	3	30%
Total	**13**	

Kevin Nguyen:

This week, the Ethernet Shield for the Arduino arrived pre-assembled. The shield was stacked onto the Arduino to test its functionality. A basic web server program was loaded onto the Arduino that displayed raw data from the analog pins. The Arduino was then connected to the university's network via Ethernet, and the data was displayed on the Arduino's web page as expected.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Research iOS Database Communications	2	Completed, 100%
iOS App Development	1	25%
Connect Arduino Onto Local Network	2	Completed, 100%
Total	**11**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
iOS App Development	1	35%
Transmit Data from Arduino to Internet	6	Completed, 100%
Total	**13**	

K. Week 12

Team 2 Date: November 17, 2014

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	15
Adam Batakji	16
Joel Barrera	13
Kevin Nguyen	14
Total	**58**

Group Meeting Minutes and synopsis of past week:

November 13, 2014:

Team 2 met to continue working on our project for the Midterm Technical Review. To compensate for the Arduino's inability to multitask, two separate Arduinos now used. One will serve as the locking system itself, controlling the LCD, LED, Piezo Buzzer, RFID Reader, and the electric door strike. The second Arduino will act as a server, transmitting received RFID data from the main Arduino to the database via Ethernet.

Currently, we have been able to display the RFID tags on the web page, which is accessible through a web browser. Progress is being made to successfully connect to the database on the same network. Mobile application will be placed on temporary hold until database is fully functional with the Arduino.

Synopsis:

Summary of Team Activities:

Task	**Task Hours This Week**	**Task Status**
Lab meetings/class	16	Completed, 100%
Research	5	90%
Web Server Work	10	75%
Mobile App Work	3	35%
Hardware	20	75%
Weekly Report	4	Completed, 100%
Total	**58**	

Forecast tasks for next week:

Task	**Task Hour Forecast**	**Task Status Forecast**
Lab meetings/class	16	Completed, 100%
Research	5	90%
Web Server Work	10	75%
Mobile App Work	0	35%
Hardware	16	80%
Weekly Report	4	Completed, 100%
Total	**54**	

Phillip Davis:

Worked on getting the 2nd Arduino up and operational, mostly part development and connection. Worked on the cabinet lock communicating with Arduino, and the pieces used for the locking device.

Tasks Assigned – Last Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Cabinet Lock Building	5	75%
Padlock Building	5	50%
Total	**15**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Arduino Communication(for midterm)	5	Completed, 100%
Hardware locking device work	7	60%
Total	**15**	

Adam Batakji:

Time is reaching its expiration date with midterm technical review this week. The plan is to demonstrate the mechanical aspect of the lock switching from its lock and unlock mode with the scan of a card through RFID. This will be paired with sending serial data to the web server and storing it in the database. As a result, a user will be able to log in the homepage and check the history of user interaction with the lock.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Web pages	7	80%
Research serial communication	2	100%
Implement serial communication	2	60%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**16**	

The current focus is showing the project in its current state for the technical review.

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meeting	4	Completed, 100%
Web pages	7	90%
Implement serial communication	4	100%
Weekly Report	1	Completed, 100%
Total	**16**	

Joel Barrera:

The time has come for the midterm technical review, which we have this week. We are approaching completion and in fact we continue to meet this week to get all of the kinks worked out. I worked on researching more AutoCAD stuff to the point where I hope to finish this week. Will complete the AutoCAD design for the Padlock and hopefully find a place where they can print at an affordable rate.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	40%
Research for 3D Printing for Team2	1	65%
Weekly Report	1	Completed, 100%
Padlock Building	3	30%
Total	**15**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	70%
Finish 3D Printing	1	100%
Weekly Report	1	Completed, 100%
Padlock Building	3	70%
Total	**13**	

Kevin Nguyen:

This week, the Arduino's multitasking limitations were overcome by introducing a second Arduino to the project. One Arduino will be locking system itself, while the other will transmit the data to other devices connected to the local network via Ethernet. The two Arduinos are communicating with each other through I2C communications. After many attempts, I have successfully sent RFID tag information from the main Arduino to the second.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	6	Completed, 100%
Weekly Report	1	Completed, 100%
iOS App Development	1	35%
Transmit Data from Arduino over Network	6	75%
Total	**14**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Tweak Arduino Code for Tech. Review	4	Completed, 100%
Transmit Data from Arduino over Network	5	Completed, 100%
Total	**15**	

L. Week 13

Team 2 Date: November 24, 2014

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	13
Adam Batakji	16
Joel Barrera	14
Kevin Nguyen	15
Total	**58**

Group Meeting Minutes and synopsis of past week:

November 18, 2014:

Team 2 met to continue working on our project for the Midterm Technical Review. The electric door strike and relay module was successfully connected to the Arduino and functions as expected. Progress is being made to establish communications from Arduino to the database.

November 19, 2014:

Met to resolve Arduino to database communication issues for the Midterm Technical Review on the following day. Created the "Punch List".

November 20, 2014:

Team 2 met with project advisor to present the Midterm Technical Review. The project in its current state cannot update the database from the Arduino. Feedback was received and pointers on what to include on our slides for the final presentation was given. The main issue is our mobile lock. What was planned was in fact a mobile lock that would be the size of a suitcase. We are in the process of looking at other solutions to compact the design into a reasonable size.

Synopsis:

Summary of Team Activities:

Task	**Task Hours This Week**	**Task Status**
Lab meetings/class	16	Completed, 100%
Research	6	90%
Web Server Work	10	75%
Mobile App Work	0	35%
Hardware	16	80%
Weekly Report	4	Completed, 100%
Total	**56**	

Forecast tasks for next week:

Task	**Task Hour Forecast**	**Task Status Forecast**
Lab meetings/class	16	Completed, 100%
Web Server Work	10	85%
Mobile App Work	2	50%
Hardware	5	90%
Weekly Report	4	Completed, 100%
End of Term Documentation	20	50%
Total	**57**	

Phillip Davis:

This week was about testing our project to get ready for the Midterm Technical Review, and then finishing up any of the problems we encountered. We got all of the hardware working together, but still had problems with communication. I also received the part to put the finishing touches on our second lock, and did research into the lilypads for wearable technology after our advisor said to do so.

Tasks Assigned – Last Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Arduino Communication(for midterm)	5	Completed, 100%
Hardware locking device work	7	80%
Wearable Technology Research	1	30%
Total	**18**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Videos for Report	2	85%
Documentation Start	5	12.5%
Cabinet Lock Hardware	1	Completed, 100%
Total	**13**	

Adam Batakji:

Serial communication has been delayed as some issue still persist. The client successfully connects to the server but the problem seems to be executing a php script from the Arduino side. Unfortunately, not much time can be spent towards that right now as the prototype documentation is the main priority.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Web pages	7	90%
Laboratory Prototype Documentation	5	12.5%
Implement serial communication	2	80%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**16**	

The current focus is working on the documentation.

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meeting	4	Completed, 100%
Web pages	3	92%
Implement serial communication	2	90%
Weekly Report	1	Completed, 100%
Laboratory Prototype Documentation	5	12.5%
Total	**15**	

Joel Barrera:

Now that the Midterm Technical Review is over with, it's on to the next part. We are approaching the end of the semester presentation, so it's important to get our project finished and somewhat polished. Starting on a presentation now wouldn't be a bad idea, but making sure we finish would be priority right now.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Team Meetings	4	Completed, 100%
AutoCAD Design for Padlock	4	70%
Finish 3D Printing	1	100%
Weekly Report	1	Completed, 100%
Padlock Building	3	70%
Total	**15**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Team Meetings	4	Completed, 100%
Finish AutoCAD Design for Padlock	6	Completed, 100%
Finish 3D Printing	2	Completed, 100%
Weekly Report	1	Completed, 100%
Begin/Continue App	1	12.5%
Total	**14**	

Kevin Nguyen:

This week, the electric door strike was interfaced with the Arduino. A relay module is used to control when to power the door strike, which would release the latch, allowing access to the room/cabinet. Progress is being made to update the database from data from the Arduino.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	2	Completed, 100%
Tweak Arduino Code for Tech. Review	4	Completed, 100%
Transmit Data from Arduino over Network	5	Completed, 100%
Total	**15**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Arduino Code	5	Completed, 100%
Laboratory Prototype Documentation	5	12.5%
Total	**14**	

M. Week 14

Team 2 Date: December 1, 2014

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	13
Adam Batakji	12
Joel Barrera	12
Kevin Nguyen	15
Total	**52**

Group Meeting Minutes and synopsis of past week:

November 25, 2014:

Team 2 met to discuss the tasks for the rest of the Fall semester. No more hardware changes will be done for the final two weeks. Minor software changes will be made, but nothing significant. Work on the Laboratory Prototype Documentation is the main priority for the week. Documentation work has been divided, with each team member working on their chosen section.

Synopsis:

Summary of Team Activities:

Task	**Task Hours This Week**	**Task Status**
Lab meetings/class	16	Completed, 100%
Web Server Work	6	85%
Mobile App Work	2	50%
Hardware	5	90%
Weekly Report	4	Completed, 100%
End of Term Documentation	20	Completed, 100%
Total	**52**	

Forecast tasks for Winter Break:

Task	**Task Hour Forecast**	**Task Status Forecast**
Meetings	64	Completed, 100%
Web Server Work	32	95%
Mobile App Work	24	75%
Hardware	32	95%
Weekly Report	4	Completed, 100%
Total	**156**	

Phillip Davis:

Worked on End of Term Document. Did the societal problem as well as design idea. Also worked on how things were going to work for the locks and put together last part of the cabinet lock. Also worked on videos of the parts working for the presentation.

Tasks Assigned – Last Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	4	Completed, 100%
Weekly Report	1	Completed, 100%
Videos for Report	2	85%
Documentation Start	5	Completed, 100%
Cabinet Lock Hardware	1	Completed, 100%
Total	**13**	

Tasks Assigned – Winter Break:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	16	Completed, 100%
Weekly Report	1	Completed, 100%
Presentation Preparation	3	Completed, 100%
Presentation Practice	1	Completed, 100%
Hardware	13	95%
Total	**34**	

Adam Batakji:

Lab prototype documentation has been completed. This was the focus for this week. Software plans and a story of what was achieved so far were the main parts of the documentation touched by me.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab Prototype Documentation	7	Completed, 100%
Team Meeting	4	Completed, 100%
Weekly Report	1	Completed, 100%
Total	**12**	

The focus now lies on preparation for the presentation.

Tasks Assigned – Winter Break:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	16	Completed, 100%
Weekly Report	1	Completed, 100%
Web Server Work	8	95%
Hardware	8	95%
Total	**33**	

Joel Barrera:

Most of the hardware is done for the semester, the rest of the focus will be on the end of term documentation as well as the presentation. With that being said, below are what I had focused on as well as the end of term documentation. The documentation is finished and it took a significant amount of time.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
End of Term Documentation	6	95%
Total	**12**	

Tasks Assigned – Winter Break:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	16	Completed, 100%
Weekly Report	1	Completed, 100%
End of Term Presentation	8	50%
Practice Presentation	8	50%
Total	**33**	

Kevin Nguyen:
This week, minor modifications were made to the Arduino code. The primary task was to complete the Laboratory Prototype Documentation.

Tasks Assigned – Last Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	5	Completed, 100%
Weekly Report	1	Completed, 100%
Arduino Code	5	Completed, 100%
Laboratory Prototype Documentation	5	Completed, 100%
Total	**15**	

Tasks Assigned – Winter Break:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	16	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App Work	24	75%
Web Server Work	8	95%
Hardware	8	95%
Total	**57**	

VIII. WEEKLY REPORTS (SPRING 2015)

A. Week 1

Team 2 Date: January 25th, 2015

Current Team Leader: Kevin Nguyen

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	5
Adam Batakji	6
Joel Barrera	6
Kevin Nguyen	7
Total	**24**

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Weekly Report	4	Completed, 100%
Outgoing Leader Report	2	Completed, 100%
Rest	400+	Completed, 100%
Miscellaneous Project Work	6	Completed, 100%
Total	**12 (+400 rest)**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Problem Statement Document Revision	20	Completed 100%
Revised Problem Statement Presentation	20	Completed 100%
Research	7	75%
Weekly Report	4	Completed, 100%
Total	**63**	

Phillip Davis:

We worked on resting our bodies this break. After much grueling hard work the previous semester we wanted to be able to attack this semester with an enthusiasm unknown to mankind. I did do a little bit of research so that we would have a direction to go when the new semester came upon us.

Tasks Assigned – Winter Break:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Hardware Upgrade Research	1	10%
Rest	100+	Completed, 100%
Total	**5**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	Completed 100%
Hardware Upgrade Research	2	50%
Total	**16**	

Adam Batakji:

The break was a time to restore oxygen to the brain. Now that this has been done, it is time to get some more work done on the project. Research was done on a new method of SPI communication using JDBC.

Tasks Assigned – Winter Break:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Rest	100+	Completed, 100%
SPI communication research	2	Completed, 100%
Total	**6**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	Completed 100%
Total	**14**	

Joel Barrera:
Much needed rest was done over the break.

Tasks Assigned – Winter Break:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Rest	100+	Completed, 100%
Software Research	2	Completed, 100%
Total	**6**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	Completed 100%
Software Update Research	3	25%
Brainstorm Project's Innovations	2	25%
Total	**19**	

Kevin Nguyen:

Majority of the time spent over the break was for resting to recover from a strenuous first semester of Senior Design. Any spare time was used to prepare for the Spring semester.

Tasks Assigned – Winter Break:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Outgoing Leader Report	2	Completed, 100%
Hardware Upgrade Research	1	10%
Rest	100+	Completed, 100%
Total	**7**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	Completed 100%
Total	**14**	

B. Week 2

Team 2 Date: February 1st, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	16
Adam Batakji	20
Joel Barrera	17
Kevin Nguyen	17
Total	**70**

Group Meeting Minutes and synopsis of past week:

Monday 26, 2015:

Team 2 met the first day of class. We met up to shortly discuss what we have done during Winter Break and also to get back into the grove. Also discussed the change in current Team Leader. Shortly discussed SPI communication using JDBC. Also discussed how we were going to shift our project to a deployable prototype as opposed to a rapid prototype which we created in the first semester of Senior Design.

Wednesday 28, 2015:

Team 2 met twice this day; once using skype and the other at school. Discussed the assignment for the week and started to assign task for each member. Went over the timeline and how we were going to update it, discussed our Problem Statement as well as our Design Idea. In addition we are also preparing for our presentation on Monday.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Problem Statement Document Revision	20	Completed 100%
Revised Problem Statement Presentation	20	Completed 100%
Research	8	75%
Weekly Report	4	Completed, 100%
Total	**64**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Test Timeline	20	Completed 100%
Testing Results Report	20	Completed, 100%
Device Test Plan	12	Completed, 100%
Research	8	75%
Weekly Report	4	Completed, 100%
Total	**76**	

Phillip Davis:

Worked on one of the articles for our revised problem statement, as well as helping to decide where we want to go for the report. E-mailed the teacher for clarification on some issues and practiced for the presentation. Also began brainstorming for next week and our testing plan.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Hardware Upgrade Research	2	50%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	75%
Total	**16**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Device Test Plan	3	Completed 100%
Revised Problem Statement Presentation	2	Completed 100%
Test Timeline	3	Completed, 100%
Testing Results Report	3	Completed, 100%
Total	**15**	

Adam Batakji:

Most of the server-side code has been imported to Eclipse IDE for simplicity of collecting data. For the most part, the code implemented so far is running well. The only thing that needs to be done is the client-side code of writing data to a socket and connecting to the server.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed, 100%
Revised Problem Statement Presentation	2	Completed, 100%
Java server Eclipse Implementation	5	Completed, 100%
Java Database Connectivity	4	In-progress, 60%
Total	**20**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Java Database Connectivity	5	Completed 100%
Plans for testing project	3	Completed 100%
Total	**12**	

Joel Barrera:

So this week starts the beginning of a new experience for me. It is the start of a leadership opportunity as leader of the team as it seems Kevin has now passed on the baton to me. This week my main focus was to get better acquainted with the ECS Hive, as it will be required when it comes time to turn in team assignments. Since my teammates will now be looking to me for guidance and direction, I spent a great deal of time trying to organize what will be required for this week as well as the upcoming one. So far this semester is wasting no time to get back in the grind, and if we want to be successful this semester, we need to stay up to date.

Tasks Assigned – This Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed, 100%
Revised Problem Statement Presentation	5	Completed, 100%
Software Update Research	3	50%
Total	**17**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Test Timeline	5	Completed, 100%
Testing Results Report	5	Completed, 100%
Device Test Plan	3	Completed, 100%
Total	**17**	

Kevin Nguyen:

This week, work was done on revising our problem statement for documentation and presentation. Other possible solutions to the mobile lock design were looked at and using XBee modules is the only viable solution at this time.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Problem Statement Document Revision	5	Completed 100%
Revised Problem Statement Presentation	5	Completed 100%
Mobile Lock possible solution	3	Completed 100%
Total	**17**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Test Timeline	3	Completed, 100%
Testing Results Report	3	Completed, 100%
Device Test Plan	3	Completed, 100%
Total	**13**	

C. Week 3

Team 2 Date: February 8th, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	15
Adam Batakji	14
Joel Barrera	17
Kevin Nguyen	13
Total	**59**

Group Meeting Minutes and synopsis of past week:

Monday February 2, 2015:

Team 2 met on Monday to go over what was to be expected for the upcoming week. We discussed how we were gonna go about our presentation (revised problem statement) and finalized our thoughts. After presentations were over, we shortly discussed what was to be expected for the week. We left off with having thoughts about how we were going to start testing our system extensively.

Wednesday February 4, 2015:

Team 2 met on Skype on Wednesday to finalize and assign projects for the team. We broke down the Testing Results Report into four different parts, where each one of us will address each accordingly. The meeting was short, as we already had a pretty clear understanding of what was expected of us for the week. Next week, we will have to start brainstorming and figuring out how we are going to start our market review. We will have to start early since we will have another presentation.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Test Timeline	20	Completed, 100%
Testing Results Report	20	Completed, 100%
Device Test Plan	12	Completed, 100%
Research	8	75%
Weekly Report	4	Completed, 100%
Total	**76**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	Completed, 100%
Market Review Presentation Preparation	20	Completed, 100%
Research	8	75%
Weekly Report	4	Completed, 100%
Total	**84**	

Phillip Davis:

Brainstormed testing situations as well as deciding how long each test would take. Did the testing plan section for tasks, timeline, and allocation in the written report.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Device Test Plan	3	Completed, 100%
Revised Problem Statement Presentation	2	Completed, 100%
Test Timeline	3	Completed, 100%
Testing Results Report	3	Completed, 100%
Total	**15**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Business Major Solicitation	3	Completed, 100%
Market Review Report	5	Completed, 100%
Market Review Presentation Preparation	5	Completed, 100%
Total	**17**	

Adam Batakji:

The MCU should now be able to hook up to the new system implemented for the server and send sample data to be assessed. Also, we are slowly moving files from the local host server to an Amazon web server so that the pages can be accessed from any network.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Amazon web server initialization	2	Completed, 100%
Plans for testing project	3	Completed, 100%
Java Database Connectivity	5	Completed, 100%
Total	**14**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software test cases	5	Completed, 100%
Server maintenance	3	Completed, 100%
Market Review Presentation Preparation	5	Completed, 100%
Total	**17**	

Joel Barrera:

We have now completed the second week of lecture at Sac State and have adjusted relatively well to the amount of work that will be expected of us this semester. There is a lot of paper work needed to be completed and reevaluated to see if it will match our more focused problem statement. One thing we are anticipating is more presentations, which we believe are valuable as we will be getting more used to speaking in front of our peers. This week I worked on our Testing Results Report. We need to continue figuring out how extensively we can test our system.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Test Timeline	5	Completed, 100%
Testing Results Report	5	Completed, 100%
Device Test Plan	3	Completed, 100%
Total	**17**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	5	Completed, 100%
Market Review Presentation Preparation	5	Completed, 100%
Mobile App	3	25%
Total	**17**	

Kevin Nguyen:

This week I worked on the Device Test Plan documentation. I was responsible for completing the Hardware Test Plan section of the document. I also began ordering parts for our revamped mobile lock portion of the project. Work will be done to assemble the mobile lock as the parts arrive the following week.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Test Timeline	3	Completed, 100%
Testing Results Report	3	Completed, 100%
Device Test Plan	3	Completed, 100%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	5	Completed, 100%
Market Review Presentation Preparation	5	Completed, 100%
Mobile Lock	5	33%
Total	**19**	

D. Week 4

Team 2 Date: February 14th, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	13
Adam Batakji	17
Joel Barrera	17
Kevin Nguyen	15
Total	**62**

Group Meeting Minutes and synopsis of past week:

Monday February 9, 2015:

Team 2 met on Monday to go over what was to be expected for the upcoming week. We discussed the main topic of the week which was the Market Review Report and brainstormed what it would be mainly about or what we think it will consist off. We also discussed how we were going to start approaching our testing phase and what days would be optimal for us to do so.

Wednesday February 11, 2015:

Team 2 met on Skype on Wednesday to finalize and assign projects for the team. We broke down the Market Review Report into four different parts, where each one of us will address each accordingly. The meeting was longer than expected, as we had to focus on ideas that would make our system more marketable, how our system compares to other products, and the history of smart locks. We also had to figure out exactly what was expected for the Market Review Report. Next week, we will have to start testing our system and getting to the stages of completion. We will have to start picking up the pace if we want to make comprehensive test cases.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	15%
Market Review Presentation Preparation	20	15%
Research	8	75%
Weekly Report	4	Completed, 100%
Total	**84**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	40%
Market Review Presentation Preparation	20	40%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	25%
Total	**104**	

Phillip Davis:

Brainstormed testing situations as well as deciding how long each test would take. Did the testing plan section for tasks, timeline, and allocation in the written report.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Business Major Solicitation	3	Completed, 100%
Market Review Report	3	20%
Market Review Presentation Preparation	3	20%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	50%
Market Review Report	3	60%
Market Review Presentation Preparation	3	60%
Testing on Hardware	5	20%
Total	**17**	

Adam Batakji:

The core frame of the server is complete now. All that remains is brushing up on details and making it more efficient in the manner it works. The next couple of weeks will be spent running tests on the software, which will mainly compose of white box testing.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software test cases	5	Completed, 100%
Server maintenance	3	Completed, 100%
Market Review preparation	5	Completed, 100%
Total	**17**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	4	50%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	50%
Total	**15**	

Joel Barrera:

Third week has now come and gone and we are directed towards a new path it seems. I am referring to the following weeks where we will be primarily focused on our Market Review Report. What I had originally thought the Market Review Report was going to consist of was completely different of what is actually expected and required to make a cohesive and comprehensive review. This week we came together as a team to come up with ways that would make our product more marketable as well as competitive. "What does our product do that will really stand out" was a common thought in my mind.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	5	15%
Market Review Presentation Preparation	5	15%
Mobile App	3	25%
Total	**17**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	25%
Market Review Presentation Preparation	4	25%
Mobile App	3	40%
Hardware/Software Test Cases	5	25%,
Total	**20**	

Kevin Nguyen:

This week, the majority of my time was spent on pushing the mobile lock to completion. All the internal hardware arrived (Lilypad Xbee, Xbee Wifi, Power Supply). The Xbee can be configured to connect to any Wifi signal when connected to the computer. The device powers on but currently does not have the ability to load Arduino code. I discovered that a second Xbee Wifi is required to program the device. I am waiting on this part to further proceed with work on the Mobile Lock.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	3	Completed, 100%
Market Review Presentation Preparation	3	Completed, 100%
Mobile Lock	5	20%
Total	**15**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	3	15%
Market Review Presentation Preparation	3	15%
Arduino I/O Testing	5	Completed, 100%
Mobile Lock	5	50%
Total	**20**	

E. Week 5

Team 2 Date: February 22, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	13
Adam Batakji	13
Joel Barrera	20
Kevin Nguyen	14
Total	

Group Meeting Minutes and synopsis of past week:

Monday February 16, 2015:

Team 2 met on Monday to go over what was to be expected for the upcoming week. This week there were no immediate assignments that needed to be addressed. We agreed to continue working on our respective parts and really start gearing towards a finished project. We received our XBee Wifi modules so that allows us to really focus on our mobile solution, which will be our focus for the following week. We also met with our instructor, who also gave us really good ideas on how to come up with and deliver a good Market Review presentation.

Wednesday February 18, 2015:

Team 2 met on Skype on Wednesday to finalize and assign projects for the team. Like previously stated, there were no immediate assignments that needed to be addressed. On the other hand, we managed to get a reliable connection between the client and the server, which was one of our biggest issues in our project. This news allows us to (hopefully) approach our mobile solution in the same manner. This will propel a lot of our test cases as well. For the coming week, our primary focus is our mobile solution as well as our Market Review presentation and document.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	40%
Market Review Presentation Preparation	20	40%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	25%
Total	**104**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	60%
Market Review Presentation Preparation	20	60%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	40%
Total	**104**	

Phillip Davis:

Mostly worked on testing hardware. Did have a couple of meetings on progress of the project as well as some of the upgrades we are working on. Testing arduino pins and expecting more of that for next week as well. Did more work on the market review portion of the project as well.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	50%
Market Review Report	3	60%
Market Review Presentation Preparation	3	60%
Testing on Hardware	5	20%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	75%
Market Review Report	3	75%
Market Review Presentation Preparation	3	75%
Testing on Hardware	3	30%
Total	**17**	

Adam Batakji:

Server and client have now established a reliable connection. Will now focus on sending different types of data to continue software testing.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	4	In-Progress, 50%
Server maintenance	3	Completed, 100%
SQL tables	2	In-Progress,50%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	3	75%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	75%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Total	**22**	

Joel Barrera:

This week has been kind of mellow, as there were no real assignments to turn in. We're taking advantage of this free time to try and wrap up our system and really gear towards a finished product. I've looked at the mobile app again and also SolidWorks. We're going to need an enclosure for the mobile solution, so it needs to be done within the next week. Next week we will complete our market review and start working on our presentation.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	25%
Market Review Presentation Preparation	4	25%
Mobile App	3	40%
Hardware/Software Test Cases	5	25%,
Total	**20**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Mobile App/SolidWorks	3	60%
Hardware/Software Test Cases	5	40%,
Total	**20**	

Kevin Nguyen:

Both XBee Wifi modules are in possession. Components of the door and cabinet locks are currently in the process of being implemented into the mobile lock, specifically the RFID reader at the moment.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock	10	50%
Total	**14**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Mobile Lock	5	100%
Total	**17**	

F. Week 6

Team 2 Date: March 1, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	
Adam Batakji	24
Joel Barrera	20
Kevin Nguyen	17
Total	

Group Meeting Minutes and synopsis of past week:

Monday February 23, 2015:

Team 2 met on Monday to go over what was to be expected for the upcoming week. This week we needed to complete our Market Review Report as well as have a presentation ready for next Monday. We came together as a team to review our thoughts about where exactly our device lies in the market, in order to come with cohesive points and to our surprise, new ideas and points we hadn't considered at the time. We had a few questions about our last meeting with our instructor regarding how to approach our presentation, so we met up with him on the same day. He gave us a lot of feedback and I think it really cleared up any questions we had that day.

Thursday February 26 , 2015:

Team 2 met on Skype on Thursday to finalize and assign projects for the team. Like previously stated, we were primarily focused on finalizing our Market Review Report and getting ready for our presentation next Monday. We also talked about the progress of our device and briefly touched on our focus for next week. We discovered that there was a 3D printer at our disposal and intend to use it immediately. We intend to get a prototype finished and hopefully printed by Monday in order to test out the quality and capabilities of the printer. We will have our Midterm Review in two weeks, so it's something to keep an eye on.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Market Review Report	20	Completed, 100%
Market Review Presentation Preparation	20	Completed, 100%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	40%
Total	**104**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Midterm Progress Review	20	20%
Device Testing Results	20	40%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	50%
Total	**104**	

Phillip Davis:

This week we focused on the presentation. I worked on my part of the presentation and we had meetings to assign which parts we were going to actually present. I worked more on some testing and putting equipment together. We also met on the final view of our project and what we wanted to work on going forward into the semester.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	75%
Market Review Report	3	75%
Market Review Presentation Preparation	3	75%
Testing on Hardware	3	30%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	Completed, 100%
Market Review Report	3	Completed, 100%
Market Review Presentation Preparation	3	Completed, 100%%
Testing on Hardware	3	50%
Total	**17**	

Adam Batakji:

Market Review preparation has been the highlight of this past week. As the project is almost complete, it is essential that our group is knowledgeable about where the project fits in the market. Any product that releases for the consumer is heavily sold based on marketing even though the technical aspects are what make it. Truth is, most people are either not interested or technologically sound to hear of the technicalities; hence, the market review.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	3	In-Progress, 75%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	In-Progress,75%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	85%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	85%
Total	**13**	

Joel Barrera:

This week, we have been primarily concerned and focused on the Market Review Report and Presentation. We talked amongst ourselves in order to get a clear understanding of where our device fits in the market. We had a few questions on the presentation portion and how we should address certain topics, but nothing too serious or far from what we originally had planned. We discovered that there is a 3D printer at our disposal, so we will also try and get a 3D model rough draft made so we can test for next week.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Mobile App/SolidWorks	3	60%
Hardware/Software Test Cases	5	40%,
Total	**20**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	4	25%
Device Testing Results	4	25%
Mobile App/SolidWorks	3	60%
Hardware/Software Test Cases	5	60%,
Total	**20**	

Kevin Nguyen:

This week I researched the competitions of our product that is available in the market. I analyzed their features and pricings, and compared it to our product for the market review report. A complete breakdown of the costs for our project was also compiled for each different lock (door, cabinet, and mobile). More work was also done in the mobile lock's development. Issue arose when uploading an Arduino sketch to the Lilypad via XBee Wifi. The problem turned out to be trivial. The part used to attach the Xbee only served as a breakout board for the XBee itself, and did not include the ATmega328 with the Arduino bootloader. Yet another component had to be order, increasing the price of the mobile lock.

Tasks Assigned – This Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Review Report	4	Completed, 100%
Market Review Presentation Preparation	4	Completed, 100%
Mobile Lock	5	60%
Total	**17**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming and Testing	15	100%
Total	**19**	

G. Week 7

Team 2 Date: March 8, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	13
Adam Batakji	13
Joel Barrera	20
Kevin Nguyen	29
Total	**75**

Group Meeting Minutes and synopsis of past week:

Monday February 23, 2015:

Monday, the team met before and after our presentation. We needed to wrap up our document, and go over what we were going to talk about in our presentation one last time. Our presentation covered all of our important points and also gave some insight about our future. Monday we set a date to meet up again and decided on Thursday.

Thursday February 26 , 2015:

We met again online on Thursday. We discussed the progress of our individual parts. We are ready to start more debugging, as we will have a dynamic webpage that will allow us more versatility. Also, we got our 3D prototype model finished, which will be printed on Monday. We will discuss the encasings for each of the locks and get accurate measurements of our components in order to properly encase them. Also, we will get our mobile system completed within the two weeks we have left before our Midterm Review. Time is running out, but we are progressing accordingly.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Midterm Progress Review	20	20%
Device Testing Results	20	40%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	50%
Total	**104**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Midterm Progress Review	20	50%
Device Testing Results	20	60%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	60%
Total	**104**	

Phillip Davis:

This past week we had our presentation, and our focus switched to the mid-term review. We have a results paper, a revision paper, and an oral presentation to work on. I have started putting together all of the hardware testing that I have done to see what else I need to work on as well as the results paper so I can check with what we said we were going to do.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Market Research	2	Completed, 100%
Market Review Report	3	Completed, 100%
Market Review Presentation Preparation	3	Completed, 100%%
Testing on Hardware	3	50%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	5	50%
Revised Test Plan Report	1	50%
Testing on Hardware	3	75%
Total	**12**	

Adam Batakji:

Progress is steady now as this week has consisted of software testing on the server-side. The web pages are about complete. Web hosting was done a couple weeks ago using Amazon Web Services, but the problem is that it has only been set up for static pages which means only html pages can be viewed online. This week, I will try to work with setting up a website that can host server-side pages so that the web pages can be fully viewed anywhere. Also, some 3d designs are ready to printed for Monday to see how the components for the mobile lock will fit.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	In-Progress, 85%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	In-Progress, 85%
Total	**13**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	95%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	95%
Total	**13**	

Joel Barrera:

Now that we have gotten our market review out of the way, our next step is to focus on a new important task. That is, it is Midterm time and our Midterm Progress Review will be due in two weeks from today. This week, I managed to finish our prototype for our 3D design and plan to have it printed for Monday, where we will all talk about our encasing for our system. We will need to address the housing for all of the locks; the mobile, cabinet and door lock.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	4	25%
Device Testing Results	4	25%
Mobile App/SolidWorks	3	60%
Hardware/Software Test Cases	5	60%,
Total	**20**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	4	50%
Device Testing Results	4	50%
Mobile App/SolidWorks	3	65%
Hardware/Software Test Cases	5	65%,
Total	**20**	

Kevin Nguyen:

This week the mobile components (Lilypad XBee, Lilypad Arduino, Power Supply, RFID reader) were soldered together in preparation for the Mid-Term Progress Review. The XBee Wifi modules were experiencing communication issues. I fell back to using the XBee S2 module to further develop the mobile lock while waiting on a second XBee USB dongle to debug the XBee Wifi modules' communications.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming and Testing	15	80%
Total	**19**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming and Testing	10	Completed, 100%
Mid-Term Progress Review Document	10	Completed, 100%
Mid-Term Progress Review Presentation	5	Completed, 100%
Total	**29**	

H. Week 8

Team 2 Date: March 15, 2015

Current Team Leader: Joel Barrera

Team Hour Summary:

Team Member	Hours
Phillip Davis	22
Adam Batakji	23
Joel Barrera	25
Kevin Nguyen	29
Total	**99**

Group Meeting Minutes and synopsis of past week:

Monday March 9, 2015:

Monday, the team met for a short while after class. We met with our advisor, but that was also cut short, as there were no urgent matters to attend. We primarily had to work out our test cases and get our report completed for next Monday. As a team, we discussed what we were gonna use for the encasing for the other solutions, since the mobile solution is already going to be 3D printed anyways. We aren't completely decided if we are going to use wood, but it was an option we were lingering on.

Thursday March 12 , 2015:

For Thursday, we had a short meeting online. We only met to understand what each part of the document we were addressing and to make sure we were all on the right page. Also, we talked about the progress of our device and we believe we are moving at an appropriate pace and believe that if all goes well, we will have a finished project within the next 5 weeks.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Midterm Progress Review	20	Completed, 100%
Device Testing Results	20	Completed, 100%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	90%
Total	**104**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Feature Presentation Report	20	50%
Feature Presentation Presentation	20	50%
Research	8	80%
Weekly Report	4	Completed, 100%
Test Cases	20	100%
Total	**104**	

Phillip Davis:

This week we had to focus on our presentation of the project. We met to see what we wanted to work on for the project, to decide what we wanted to have completed by the presentation date. Worked on hardware testing to meet all of the market review requirements for the hardware. I also started the report and will have it finished by tomorrow, so that we can show the things we tested for as well as the results of some of the things that could happen in our design. As we tested we also fixed some bugs that had been bothering us, such as the turning on alarm.

Tasks Assigned – This Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	7	80%
Revised Test Plan Report	3	50%
Testing on Hardware	5	95%
Lab meetings/class	3	Completed, 100%
Total	**4**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Project Beautification	5	60%
Team Member Evaluations	1	Completed, 100%
Feature Checking	4	25%
Total	**14**	

Adam Batakji:

Prototypes of the mobile lock have been 3d printed. The web page hosted by Amazon Web Services now has Apache, MySQL, and PHP installed on it. This now allows PHP applications on the website. The goal is to imitate the local host on the AWS page so that it can be viewed from any network.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	In-Progress, 95%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	In-Progress, 95%
Mid-Term Progress Review Document	5	Completed, 100%
Mid-Term Progress Review Presentation	5	Completed, 100%
Total	**23**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	100%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	100%
Feature Presentation preparation	6	100%
Total	**19**	

Joel Barrera:

A lot of progress has been made this week. We managed to 3D print prototypes for the mobile device but are still unsure of what we are gonna create for the other solutions. We will have to re-create the sliding mechanism we created for the mobile solution and hopefully get that printed for Monday. We mainly focused on getting the Midterm Progress Review Documentation and Presentation finished by Monday.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review Document	4	100%
Midterm Progress Review Presentation	5	100%
Device Testing Results	4	100%
Mobile App/SolidWorks	3	90%
Hardware/Software Test Cases	5	90%,
Total	**25**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Feature Presentation Report	4	100%
Feature Presentation Presentation	5	100%
Mobile App/SolidWorks	3	100%
Hardware/Software Test Cases	5	100%,
Total	**21**	

Kevin Nguyen:

This week's main focus was to complete the Mid-Term Progress Review document and preparing for the Mid-Term presentation. Testing was also carried out on the hardware components as described in the Device Test Plan document. The cause of the issue that prevented code to be uploaded to the Lilypad has been identified. The D0 pin of the RFID reader caused uploading issues when connected to the Lilypad. Disconnecting the wire and reconnecting after upload solved this issue.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming Main Functions	8	Completed, 100%
Testing	2	Completed, 100%
Mid-Term Progress Review Document	10	Completed, 100%
Mid-Term Progress Review Presentation	5	Completed, 100%
Total	**29**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	5	90%
Feature Presentation Report	10	50%
Feature Presentation Presentation	5	50%
Total	**24**	

I. Week 9

Team 2 Date: March 29, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	Hours
Phillip Davis	24
Adam Batakji	16
Joel Barrera	21
Kevin Nguyen	24
Total	**85**

Group Meeting Minutes and synopsis of past week:

Monday March 16, 2015:

A presentation of the project at its current status was shown to the advisor. After showing him current status on the features, questions were asked that brought insight upon which parts and features to improvise on. Collaboration was helpful in identifying what needs more work.

Thursday March 19 , 2015:

A checkpoint on everyone in the group was established to see if there were any issues or concerns. Roadblocks are surely imminent at this point; being able to communicate with each other when this happens is key. For now, each part is moving along in an exceptional manner. With the break taking place next week, productivity will either increase greatly or decrease. The former is highly desired with the semester approaching just over a month left.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Feature Presentation Report	20	50%
Feature Presentation	20	50%
Research	8	80%
Weekly Report	4	Completed, 100%
3D printing work	10	100%
Total	**94**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Feature Presentation Report	20	100%
Feature Presentation Presentation	20	100%
Research	8	80%
Weekly Report	4	Completed, 100%
Mobile application development	20	100%
Total	**104**	

Phillip Davis:

This week we had a break from school, but not from senior project. We worked on parts of the project we were trying to beautify as well as parts that needed to be put in place to make the mobile lock run off of battery instead of being plugged in. We blew up our lilypad and may need to go back to an arduino uno and since our 3d print is big enough, this may be our only expense/time option left. We also worked on some beautification of the project so that we will not just be showing wires and cords when we are in the alleyway. This includes getting a box for our door/cabinet lock and some other manual labor.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Midterm Progress Review	2	Completed 100%
Revised Test Plan Report	1	Completed 100%
Testing on Hardware	1	Completed 100%
Feature Checking Start	2	10%
Project Beautification	12	40%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Project Beautification	5	60%
Feature Checking	4	25%
Team Evaluations	2	Completed 100%
Total	**12**	

Adam Batakji:

An acquired IP address has been obtained for the server and its database on Amazon Web Services. After multiple tries with the 3D printer, a valid design was achieved. Hardware components will now be placed within for testing. Focus will now be aimed at mobile application development for this week.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	100%
Server maintenance/Testing	5	Completed, 100%
SQL tables	2	100%
3D prints	3	100%
Total	**16**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	8	100%
Server maintenance/Testing	5	Completed, 100%
Feature Presentation preparation	6	100%
Total	**23**	

Joel Barrera:

We just got done with our Midterm Progress Review. It went well I believe and it showed we had a clear plan to finish our design. This week me and Adam teamed up to get our 3D print finalized after multiple attempts! Our primary focus will now be centered around the mobile application, which still needs a lot of work.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Feature Presentation Report	4	50%
Feature Presentation Presentation	5	50%
Mobile App	3	50%
Hardware/Software Test Cases	5	50%,
Total	**21**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Feature Presentation Report	4	60%
Feature Presentation Presentation	5	60%
Mobile App	3	60%
Hardware/Software Test Cases	5	6%,
Total	**21**	

Kevin Nguyen:

This week, time was spent on further development on the mobile lock. Several lilypad components unfortunately blew up. A decision was made to fall back to using the Arduino Uno as it will still manage to fit in our mobile lock casing. Work was also done on building a full-sized door for demonstration purposes for our locking system.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	5	70%
Feature Presentation Report	5	20%
Feature Presentation Presentation	5	10%
Door Assembly	5	40%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	5	Completed, 100%
Feature Presentation Report	10	Completed, 100%
Feature Presentation Presentation	5	Completed, 100%
Total	**24**	

J. Week 10

Team 2 Date: April 5, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	Hours
Phillip Davis	16
Adam Batakji	25
Joel Barrera	21
Kevin Nguyen	24
Total	**86**

Group Meeting Minutes and synopsis of past week:

Monday March 30, 2015:

A discussion took place of what remains to be complete. All team members are busy completing their assigned tasks at this point. Since all hardware should be completed by April 20, that remains the biggest priority which is in collaboration by most of the group.

Thursday April 2 , 2015:

Each team member clarified which feature they will talk about during the feature presentation. It worked out that each person knew one of the features best for a selection. Questions and clarification about each feature were addressed as we prepare for next week.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Feature Presentation Report	20	100%
Feature Presentation	20	100%
Research	8	85%
Weekly Report	4	Completed, 100%
Mobile application development	20	30%
Total	**104**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Mobile lock configurations	10	50%
Door lock configurations	10	50%
Research	8	80%
Weekly Report	4	Completed, 100%
Mobile application development	20	50%
Total	**84**	

Phillip Davis:

This week we focused on our feature presentation as well as flaws we had in our original project design. We are working on another design for the mobile lock. One that will be more stable and rely less on other companies products. We also completed a large amount of our beautification and got well on our way to completing the wifi portion of our mobile lock since we had to change that at the last minute.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Feature Presentation Write Up	4	Completed 100%
Feature Presentation Powerpoint	3	Completed 100%
Project Beautification	4	60%
Wifi Coding Assist	1	30%
Total	**16**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Project Beautification	2	80%
Finalize Hardware Components	6	90%
Wifi Coding Assist	1	60%
Total	**13**	

Adam Batakji:

After spending some time with mobile application development, an accurate scope and objective has been set. Previously, it was thought that the mobile application would connect directly to the database to fetch data. It was learned that idea was a bad practice and the better approach would be to have JSON (JavaScript Object Notation) be the middleman in intercepting data through PHP pages. This week will set the task of accomplishing the JSON encoding.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Link html/php pages	2	100%
Server maintenance/Testing	5	Completed, 100%
Mobile app development	8	30%
Feature Presentation preparation	6	100%
Total	**25**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	15	100%
Server maintenance/Testing	5	Completed, 100%
Total	**24**	

Joel Barrera:

We are approaching our deadline for the hardware aspect of our project. We have only a few days left and are determined to finish our system as soon as possible. With that being said, we need to get the mobile lock completed. The upcoming weeks for me will be mainly focused on the mobile application. It turns out that we might have slightly underestimated the amount of time required to get a decent app up and running. Also my focus will be spent on finishing the mobile lock. We intend to use a push solenoid that will hopefully fix our headache we had previously.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Feature Presentation Report	4	100%
Feature Presentation Presentation	5	100%
Mobile App	3	65%
Hardware/Software Test Cases	5	60%,
Total	**21**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	60%
Hardware/Software Test Cases	5	60%,
Total	**19**	

Kevin Nguyen:

This week, time was spent on further development on the mobile lock. The Wifi shield functions as expected. The sensor shield functions normally while attached to the Wifi shield. The Ethernet shield solution previously explored experienced issues when both Ethernet and sensor shields are connected. This resulted in using two separate Arduinos for it to work. The Wifi solution minimizes the needed components to a single Arduino. The Feature Presentation report and powerpoint was also worked on. The feature I was responsible for was RFID.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	5	80%
Feature Presentation Report	6	Completed, 100%
Feature Presentation	5	Completed, 100%
Door Assembly	4	Completed, 100%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	5	Completed, 100%
Finalize Hardware Components	10	90%
Total	**19**	

K. Week 11

Team 2 Date: April 12, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	Hours
Phillip Davis	15
Adam Batakji	24
Joel Barrera	23
Kevin Nguyen	15
Total	**77**

Group Meeting Minutes and synopsis of past week:

Monday April 6, 2015:

The group was planning to do the feature presentations, but it was decided to wait until next week instead. Since we were all in one spot, we took advantage and discussed future plans; especially where we would like to be as 4/20 approaches. Everyone seemed to have an understanding of what they need to finish.

Thursday April 9, 2015:

We are busy with finishing up on hardware tasks as well as software development o the mobile application. This day was open to questions from any of the members for clarification on anything not clear. Group messages were sent throughout the day, which has been a great way to communicate for small questions.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Feature Presentation Report	20	100%
Feature Presentation	20	100%
Research	8	85%
Weekly Report	4	Completed, 100%
Mobile application development	20	50%
Total	**114**	

Forecast tasks for next week:

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Mobile lock configurations	10	50%
Door lock configurations	10	50%
Research	8	90%
Weekly Report	4	Completed, 100%
Mobile application development	20	60%
Total	**84**	

Phillip Davis:

This week we worked on getting all of the final parts for our hardware. We are waiting until next week to present so we have all of our paperwork finished. We have to put everything together and try to finish up the beautification by next monday so that we can keep all of our stuff to show off. We still need to work on some software things but we have a little longer to do those so hardware is the focus.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Project Beautification	2	80%
Finalize Hardware Components	6	90%
Wifi Coding Assist	1	60%
Project Beautification	2	80%
Total	**15**	

Adam Batakji:

From the beginning, I was not sure how complex development on the mobile application would be. As I have got more involved, it has been realized that there many tools to play with as well as many keywords. Other than that, the logic is simple to figure out. Encoding JSON has become more clear after working through some tutorials on the different ways to do it. The goal for this week will be to have multiple pages encoding with some data that comes from the database.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Server maintenance/Testing	5	Completed, 100%
Mobile app development	15	40%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	15	50%
Server maintenance/Testing	5	Completed, 100%
Total	**24**	

Joel Barrera:

We are now at our final week to get everything completed. There is definitely pressure to perform and finish, but we are still on track. We will finish the mobile lock and make everything completed for grading. We are primarily focused on getting everything finished, or in other words the Deployable Prototype Review and Presentation.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	65%
Hardware/Software Test Cases	5	65%,
Feature Presentation	4	Completed, 100%
Total	**23**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	60%
Finalize Hardware	10	Completed, 100%
Deployable Prototype Review Presentation	5	Completed, 100%
Hardware/Software Test Cases	5	60%,
Total	**34**	

Kevin Nguyen:

This week, time was spent on finalizing all hardware components in preparation for the Deployable Prototype Review and Presentation. Hardware components are currently in process on being placed on the physical door. The main program code was also further modified to display other messages. Code was also added to lockdown the system in response to multiple failed attempts of entry.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile Lock Programming	4	Completed, 100%
Finalize Hardware Components	7	90%
Total	**15**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Hardware Components	10	Completed, 100%
Deployable Prototype Review Presentation	5	Completed, 100%
Total	**19**	

L. Week 12

Team 2 Date: April 19, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	Hours
Phillip Davis	24
Adam Batakji	24
Joel Barrera	34
Kevin Nguyen	24
Total	102

Group Meeting Minutes and synopsis of past week:

Monday April 13, 2015:

This day was reserved for our feature presentations. The members of our group showcased their true knowledge of the feature they worked most with. In other news, next week is showcasing the hardware. Arrangements were made to finalize our set up.

Thursday April 16 , 2015:

Joel, Phillip and Kevin were busy finalizing the hardware. They met up to complete their task. Adam continued working on the mobile application with steady progress. Everything is looking up for our project.

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Mobile lock configurations	10	50%
Door lock configurations	10	50%
Research	8	90%
Weekly Report	4	Completed, 100%
Mobile application development	20	60%
Total	**84**	

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Deployable Prototype Report	20	100%
Deployable Prototype Presentation	20	100%
Research	8	85%
Weekly Report	4	Completed, 100%
Mobile application development	20	70%
Total	**114**	

Phillip Davis:

We had our presentation this week, and were introduced to other solutions. At this point we would not be able to use them in our project, but it is still good knowledge going forward in engineering. We worked on more of the hardware, and decided to make some last minute changes such as attempting to build the cabinet lock so it too can be on display. We had decided to only do one of the locks and then have it communicate but now we are displaying all three. We need to finish the mobile lock build and the cabinet lock build by this week. Daunting but not an impossible task.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Project Beautification	5	100%
Finalize Hardware Components	6	100%
Wifi Coding Assist	1	80%
Project Beautification	5	100%
Deploy Presentation	3	100%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software Coding	5	90%
End of Project Documentation	7	50%
Total	**16**	

Adam Batakji:

With so many hours spent in the mobile application development environment, it does not look foreign anymore. Most of the coding is done in the multiple controller classes. Each class represents a viewable page on the application. The overview will depict of a login page along with any linked data pages that are associated with privileges granted to the user or admin. It is not expected that the application will look pretty but it will be functional.

Tasks Assigned – This Week:

Task	**Hours**	**Status and % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Server maintenance/Testing	5	Completed, 100%
Mobile app development	15	50%
Total	**24**	

Tasks Assigned – Next Week:

Task	**Hours**	**Projected % Completion**
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	15	60%
Server maintenance/Testing	5	Completed, 100%
Total	**24**	

Joel Barrera:

As for the last week before we get evaluated, we needed to be complete with the project. We made sure we have/had all of the hardware completed one way or another. Sacrifices needed to be made to an extent, but all for a good cause. Met up with the rest of the team to finalize the hardware aspect during the weekend. Making sure we will address everything we said we will deliver. The software of the project still needs work, but that can be put off to a certain extent.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	60%
Finalize Hardware	10	Completed, 100%
Deployable Prototype Review Presentation	5	Completed, 100%
Hardware/Software Test Cases	5	60%,
Total	**34**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	70%
Finalize Hardware	10	Completed, 100%
Finalize Software	5	Completed, 100%
Hardware/Software Test Cases	5	70%,
Total	**34**	

Kevin Nguyen:

This week, time was spent on further finalizing all hardware components in preparation for the Deployable Prototype Review and Presentation. The door lock was neatly assembled and placed inside a casing that is mounted on the door. The solenoid lock was interfaced with the door lock system as well to demonstrate the cabinet lock without the need to duplicate all hardware components.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Hardware Components	10	Completed, 100%
Finalize Arduino Code	5	80%
Deployable Prototype Review Presentation	5	Completed, 100%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	10	90%
End of Project Documentation	5	50%
Total	**19**	

M. Week 13

Team 2 Date: April 26, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	Hours
Phillip Davis	18
Adam Batakji	24
Joel Barrera	24
Kevin Nguyen	19
Total	**85**

Group Meeting Minutes and synopsis of past week:

Monday April 20, 2015:

This was an important day of transitioning for our group. We had presented our project as a whole for our deployable prototype. The mobile lock, door lock and cabinet lock were shown in its functionality with beautification in order. Instructor had some questions about the alarm and it gave us more ideas of what can be done to fix.

Thursday April 23 , 2015:

At this point, documentation is what remains. Everyone agreed to get started on putting the document together. Also, possible plans of painting the door for the trade show was discussed. A poster board to greet passer bys will be done to bring attention to the trade show as well.

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Deployable Prototype Report	20	100%
Deployable Prototype Presentation	20	100%
Research	8	85%
Weekly Report	4	Completed, 100%
Mobile application development	20	80%
Total	**114**	

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Mobile lock configurations	10	70%
Door lock configurations	10	50%
Research	8	90%
Weekly Report	4	Completed, 100%
Mobile application development	20	60%
Total	**84**	

Phillip Davis:

Since my focus has been hardware on this project there is a lot I am finished with. I am still assisting Kevin with coding and working with Joel to build the cabinet lock from parts that are not just a cardboard box, but there is still a decent amount of software coding to finish up. For most of the week I worked on the beginning of the project documentation and have more to finish for my section of the report.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software Coding	5	90%
End of Project Documentation	7	50%
Project Beautification	2	100%
Total	**18**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software Coding	5	950%
End of Project Documentation	7	Complated, 100%
Total	**16**	

Adam Batakji:

As we come close to the end, I look forward to not having to do another weekly report. To stay on topic, this week consisted of deployable prototype presentation as a group. That went better than expected. As a solo member, I continued to add to the mobile application with the combination of Joel's development and my code. Basically, panel shifts with JSON encoding is the vision of the final outcome.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Server maintenance/Testing	5	Completed, 100%
Mobile app development	15	60%
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	15	70%
Server maintenance/Testing	5	Completed, 100%
Total	**24**	

Joel Barrera:

We just finished our hardware for the most part. We will need to create a cabinet soon, so that will be one of our main focuses. Also, our code needs to be finalized. I will be working primarily on XCode and making sure that the layout of the application resembles somewhat that of the website to keep things consistent and less confusing for the clients.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	70%
Finalize Hardware	10	Completed, 100%
Finalize Software	5	Completed, 100%
Hardware/Software Test Cases	5	70%,
Total	**34**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	70%
Finalize Software	5	Completed, 100%
Hardware/Software Test Cases	5	70%,
Total	**24**	

Kevin Nguyen:

This week, time was spent on further finalizing the Arduino code for the door/cabinet and mobile lock. A lockdown feature was implemented to lock the system down after three unauthorized attempts of access. A master key will be required to reset the system. As suggested for the mobile lock, when the lock is breached, the light sensor will also set the lock into lockdown. This will prevent the perpetrator from simply covering the sensor with their finger to turn off the alarm sound. A master key will also be required to disarm the alarm. Parts of the End of Project Documentation are beginning to be pieced together.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	10	90%
End of Project Documentation	5	50%
Total	**19**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	10	95%
End of Project Documentation	5	Completed, 100%
Total	**19**	

N. Week 14

Team 2 Date: May 3, 2015

Current Team Leader: Adam Batakji

Team Hour Summary:

Team Member	**Hours**
Phillip Davis	16
Adam Batakji	27
Joel Barrera	24
Kevin Nguyen	19
Total	

Group Meeting Minutes and synopsis of past week:

Monday April 27, 2015:

Completed course evaluations

Thursday April 30 , 2015:

Group is currently working on the end of project documentation

Synopsis:

Summary of Team Activities:

Task	Task Hours This Week	Task Status
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed, 100%
Deployable Prototype Report	20	100%
Deployable Prototype Presentation	20	100%
Research	8	100%
Weekly Report	4	Completed, 100%
Mobile application development	20	90%
Total	**114**	

Task	Task Hour Forecast	Task Status Forecast
Lab meetings/class	12	Completed, 100%
Independent Work	20	Completed 100%
Mobile lock configurations	10	100%
Door lock configurations	10	100%
Research	8	100%
Weekly Report	4	Completed, 100%
Mobile application development	20	80%
Total	**84**	

Phillip Davis:

This week the main focus was trying to get the end of project documentation finished, as well as working on the mobile application and some arduino software adjustments. This documentation a big portion of our project and as we are coming down to an end it has been difficult keeping everything together, especially with the demands of our other classes.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Software Coding	5	950%
End of Project Documentation	7	Completed, 100%
Total	**16**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	10	Completed, 100%
Deployable Prototype Presentation	10	Completed, 100%
Total	**24**	

Adam Batakji:

Mobile application development is almost complete and all that remains is the end of project documentation along with any necessary plans to prepare for the trade show.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
End of project documentation	8	Completed, 100%
Mobile app development	15	80%
Total	**27**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile app development	15	90%
End of project documentation	8	Completed, 100%
Total	**27**	

Joel Barrera:

Our main focus for the week was finalizing our software. The End of Project Documentation was also on our mind the majority of the time as it will need to be completed fairly shortly. For this coming week, it will remain the same with an emphasis on finalizing the mobile application. We are relatively close to finishing, so we aren't completely stressed about it yet.

Tasks Assigned – This Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	70%
Finalize Software	5	Completed, 100%
Hardware/Software Test Cases	5	70%,
Total	**24**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Mobile App	10	70%
Finalize Software	5	Completed, 100%
Hardware/Software Test Cases	5	70%,
Total	**24**	

Kevin Nguyen:

The End of Project Documentation was the primary focus for this week. The document was assembled according to the required specifications. The final code will be completed and preparations for the Deployable Prototype Presentation will be done for the coming week.

Tasks Assigned – This Week:

Task	Hours	Status and % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	5	95%
End of Project Documentation	10	Completed, 100%
Total	**19**	

Tasks Assigned – Next Week:

Task	Hours	Projected % Completion
Lab meetings/class	3	Completed, 100%
Weekly Report	1	Completed, 100%
Finalize Arduino Code	10	Completed, 100%
Deployable Prototype Presentation	10	Completed, 100%
Total	**24**	

IX. User Manual

A. How to use web page:

The hardware (locking system) successfully connects to the web server that is run on Apache software. As of now, the server runs on a local host. This means that only devices on the same network as the host can connect.

The purpose of the web server is to create a system of communication with the devices. One of the most crucial aspects to hardware is that it can relay data to a network that can be viewed anywhere.

One of the tasks for next semester will be to move from a local host to a web domain. An IP address will be reserved for the server and thus it can be accessed from any network.

The local host currently has some web pages composed of HTML and PHP. The HTML serves as a basic UI (user interface) to help the user navigate easily through the pages to do as they wish. As of now, a user can register an account, login and attempt to view data. The attempt to view data takes the user to an empty table for now, since serial communication has not been fully established yet. This is what the PHP is responsible for. The script language is that pathway from user/device interactions to the database.

The current device that the RFID reader is pinned to (Arduino UNO) executes a PHP file with an insert query to the database. This relays the RFID and a timestamp to the database.

Currently, the database contains several tables, which keep track of the following data:

- User accounts
- Attempted logins
- RFID check ins

When a user signs up for an account, their information (email address, username and password) will be stored in a table. Logging in compares the user input to fields in the table to look for a match.

Every time a password is entered incorrectly for a specified username, a session will begin keeping track of failed logins and eventually lock the account for a certain amount of time for security if failed logins reach a certain number.

RFID check in keeps track of when the user has interacted with their lock. This can be a good reference if needed to check for a time and date of last use or as another security measure. When the mobile application is complete, one of its features will be a notification system for each time the lock has been used. The user will be the first to know if their lock is supposedly being hacked.

B. Mobile Application

Using the mobile application is quite similar to the webpage. Just as someone uses a mobile application of a website they log on to from their computer, the same is done with this application.

When first opening the application, the user is greeted with a login page that asks for a username and password. If they have not yet acquired a username, there is an option to register for Loksys.

Clicking “Register” takes the user to another page that asks for some basic information such as name, email, username, password, etc. After that, depending if the registered user is a basic user or administrator, they login in the appropriate directory.

Functionality is obvious after login as the user will be able to see data that shows their past uses of their locking systems, their personal RFID code, what and how many locks are registered and much more.

One last important feature that is present on the application is the alarm mode. When the lock is set with the alarm, the user’s phone will be notified if any tampering is done to the lock. This is quite useful in situations when the mobile lock is in use and the user is close enough to get to their belongings and avoid any losses.

C. Door Lock

This product will be installed by a professional electrician. They are meant to be connected to power lines in the home and will not be safe if left alone for the user to do.

New User

Since the software comes preloaded, all you need to do to get started is have your user account information as well as your computer or handheld device to add yourself as a user. After you have added yourself to the website as an admin, or the admin has added you, then you are able to begin using the device.

Use

The lock will be opened with the code from the device that you have purchased. There are multiple keys types that will open the lock including buttons, cards, or fobs. Since you have a plethora of choices, please choose a key that is to your liking and works for your situation. After a 1.5 second delay you will hear the lock disengage and you will be able to enter. The lock will resume its previous locked position after a timer of 5 seconds. The door will still be able to be closed after the allotted time, but if you have had a delay of over 5 seconds in opening the door, you will have to use your key once again to unlock it. The Users name as well as clearance level will appear on the LCD screen. This will allow you to see what the admin has input for your information and lets you know which places you have access to in the house.

Alarm

This lock comes with an alerting alarm. It will allow the user to know if they do not have access to the lock. This means that if the incorrect card is used the alarm will go off and the incorrect card usage will be uploaded to the database for future reference. This is not a replacement for house alarm systems.

Warning:

1. During power outages you will not be able to use the lock. If you are inside then you will still be able to get out, but if you are outside you will not be able to get in. This is why Team Loksys suggests a backup generator if these problems are frequent.

2. During internet outages you will not be able to add or remove users. This can be circumvented by using the mobile application to add and remove users, if there is cellphone reception, or just simply waiting until the internet is back up. During this time period people who come and go will not be logged into the database and some data could be lost.

3. The card type necessary for the reader is the lower frequency of 125kHz. This means if you get high frequency cards, there will be reading problems with the system and you will not get the results that you expect.

4. This alarm is not intended to replace any home security system.

D. Cabinet Lock

This product will be installed by a professional electrician. They are meant to be connected to power lines in the home and will not be safe if left alone for the user to do.

New User

Since the software comes preloaded, all you need to do to get started is have your user account information as well as your computer or handheld device to add yourself as a user. After you have added yourself to the website as an admin, or the admin has added you, then you are able to begin using the device.

Use

The lock will be opened with the code from the device that you have purchased. There are multiple keys types that will open the lock including buttons, cards, or fobs. Since you have a plethora of choices, please choose a key that is to your liking and works for your situation. After a 1.5 second delay you will hear the lock disengage and you will be able to enter. The lock will resume its previous locked position after a timer of 5 seconds. The door will still be able to be closed after the allotted time, but if you have had a delay of over 5 seconds in opening the door, you will have to use your key once again to unlock it.

Alarm

This lock comes with a clearance level system. If the level is not high enough, the alarm will beep to notify the user that they are unable to enter the lock. This will happen an infinite number of times and if the admin or another user is in the area, they will be able to prevent someone from trying to use a lock they are not permitted to use. The alarm will also go off in the case of an incorrect card or user trying to access the lock.

Warning:

1. During power outages you will not be able to use the lock. This is why Team Loksys suggests a backup generator if these types of problems are frequent.

2. During internet outages you will not be able to add or remove users. This can be circumvented by using the mobile application to add and remove users, if there is cellphone reception, or just simply waiting until the internet is back up. During this time period people who come and go will not be logged into the database and some data could be lost.

3. The card type necessary for the reader is the lower frequency of 125kHz. This means if you get high frequency cards, there will be reading problems with the system and you will not get the results that you expect.

4. This alarm is not intended to replace any home security system.

E. Mobile Lock

This product will be assembled and ready for use out of box. Simply attach the required batteries and the mobile lock will be ready for configuration.

New User

Since the software comes preloaded, all you need to do to get started is have your user account information as well as your computer or handheld device to add yourself as a user. After you have added yourself to the website as an admin, or the admin has added you, then you are able to begin using the device.

Use

The mobile lock is used to secure any type of locker that would normally be secured with a traditional padlock or combination lock. Once secured, the lock will remain locked until an authorized user's RFID tag is read. After the authorized tag is read, the shackle will release, allowing access to the locker with removal of the lock.

Alarm

The lock will play a short sound when an unauthorized tag is read. After three failed attempts, the lock will lock down the system, preventing any more attempts to unlock the locker. Lockdown will also be activated when the lock has been physically breached and broken open. The lock will remain in lockdown until the owner disarms it himself by scanning their RFID tag or disarming it remotely via mobile application.

Warning:

1. Battery will need to be replaced when low on charge. Lock will remain in locked position when without power.

2. During internet outages you will not be able to add or remove users. This can be circumvented by using the mobile

application to add and remove users, if there is cellphone reception, or just simply waiting until the internet is back up. During this time period people who come and go will not be logged into the database and some data could be lost.

3. The card type necessary for the reader is the lower frequency of 125kHz. This means if you get high frequency cards, there will be reading problems with the system and you will not get the results that you expect.

X. HARDWARE BLOCK DIAGRAM

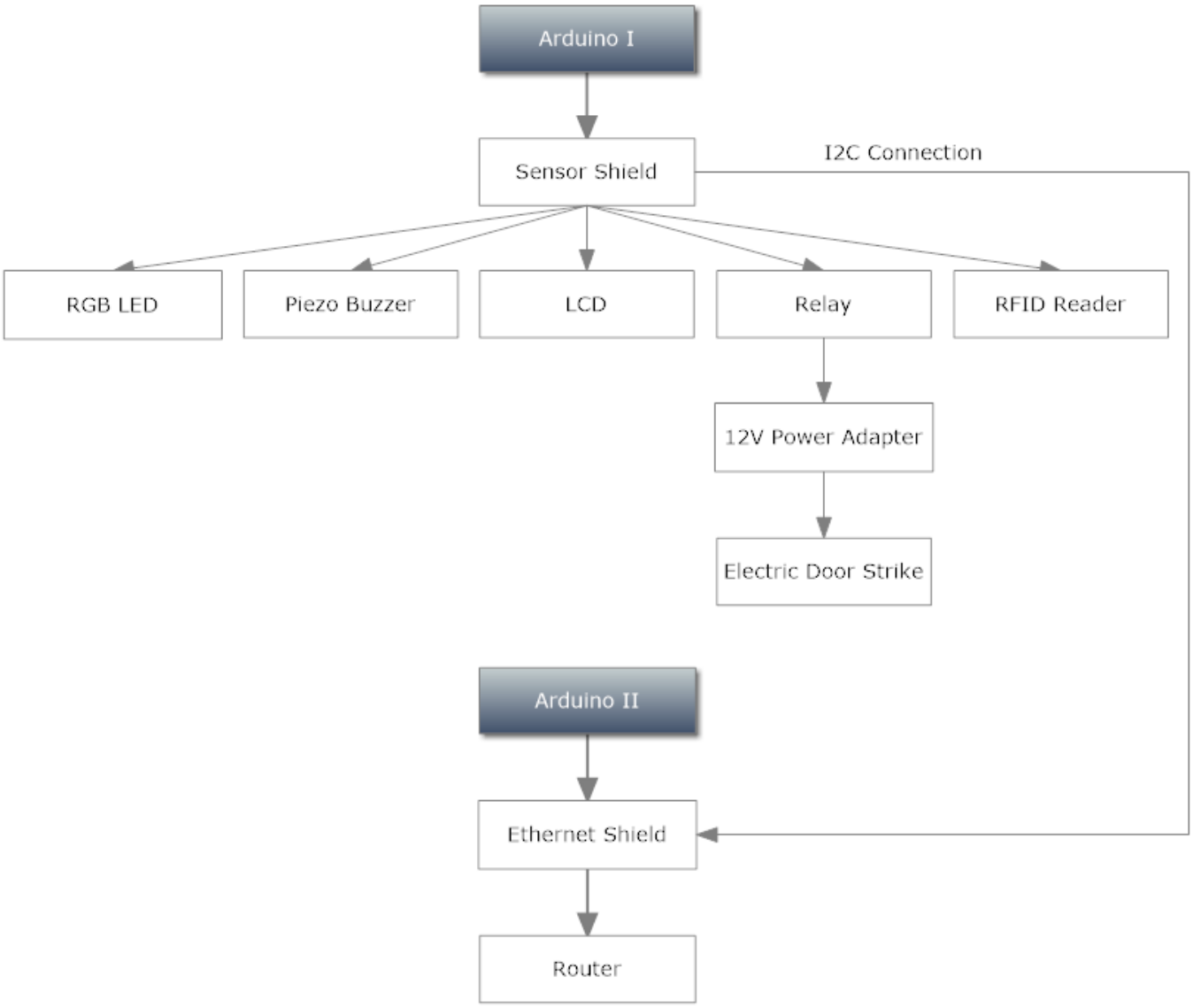


Fig. 9. Hardware block diagram.

XI. SOFTWARE FLOWCHART

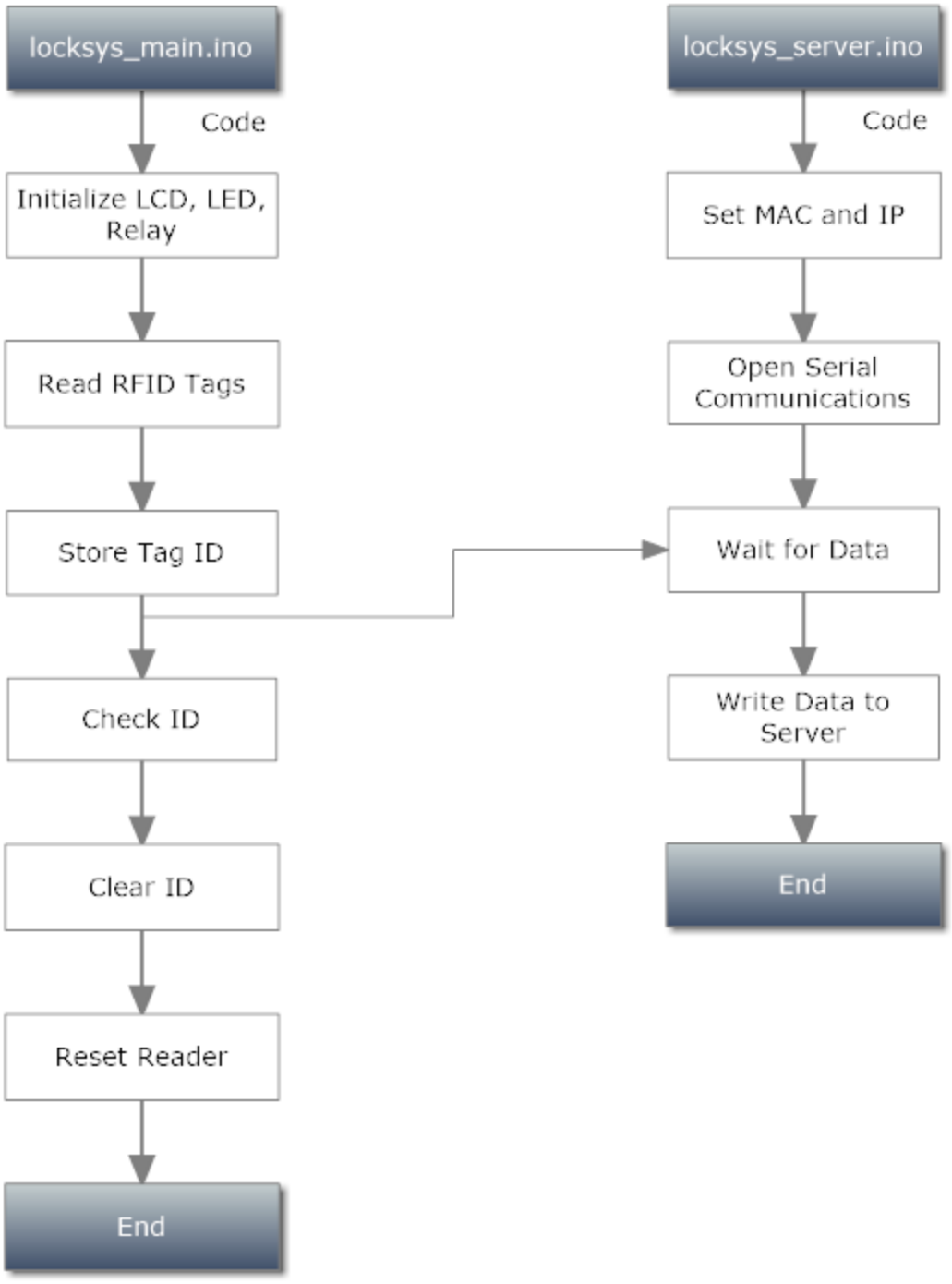


Fig. 10. Software flowchart.

XII. PHOTO DOCUMENTATION

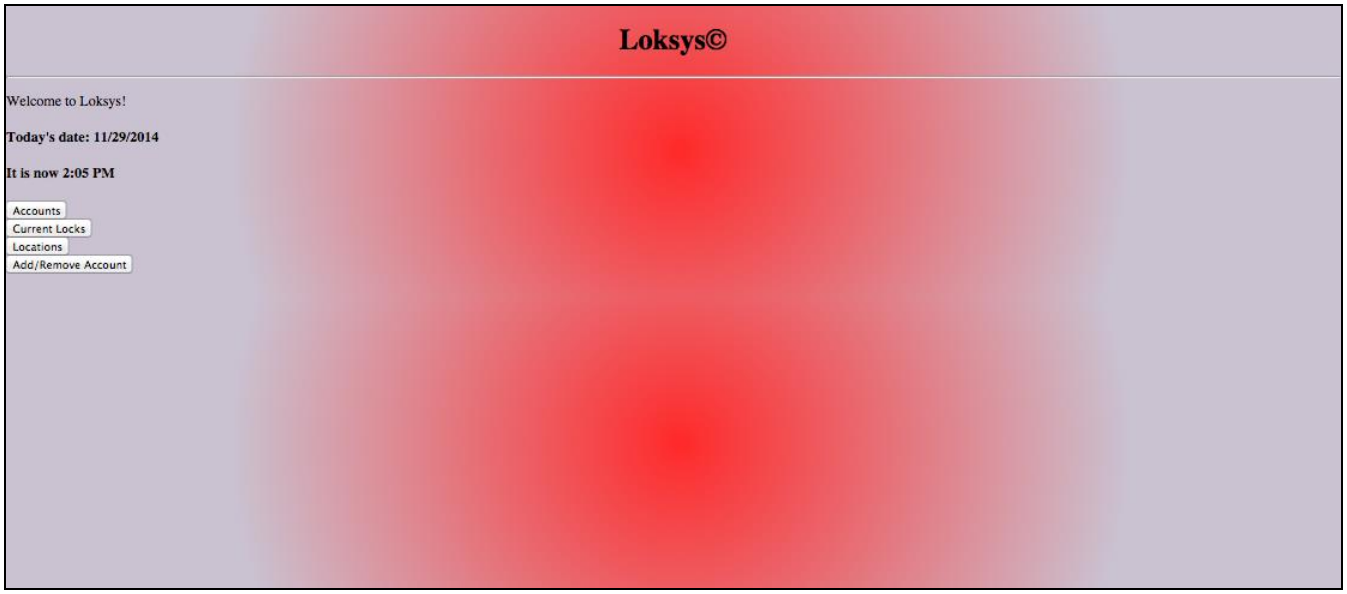


Fig. 11. Admin Personal Homepage.

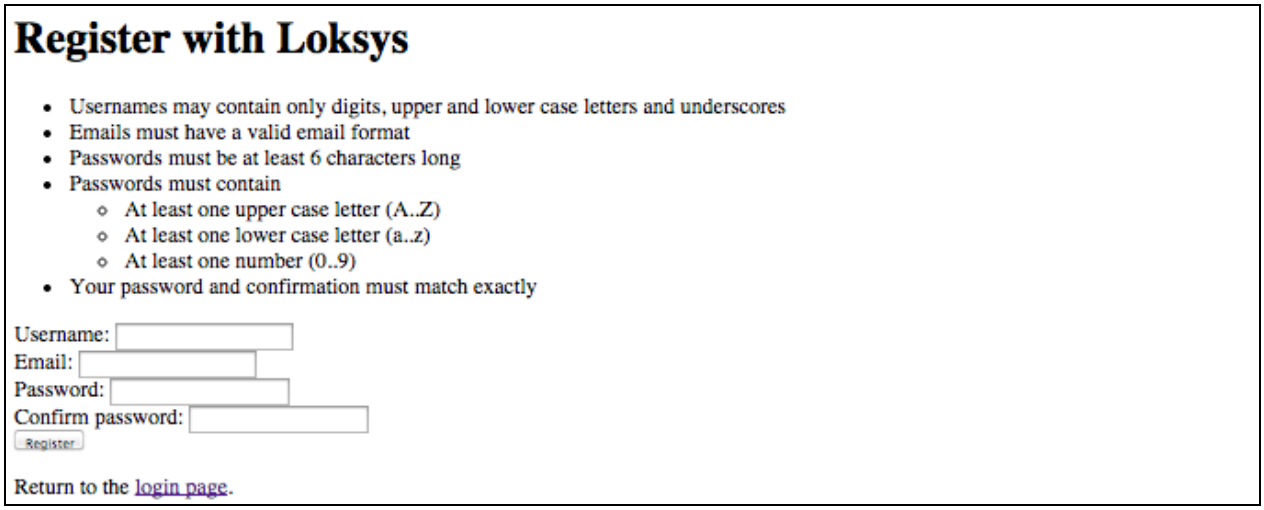


Fig. 11. Registration Page (currently functions).

Fig. 12. Login Page (currently functions).

XIII. MECHANICAL

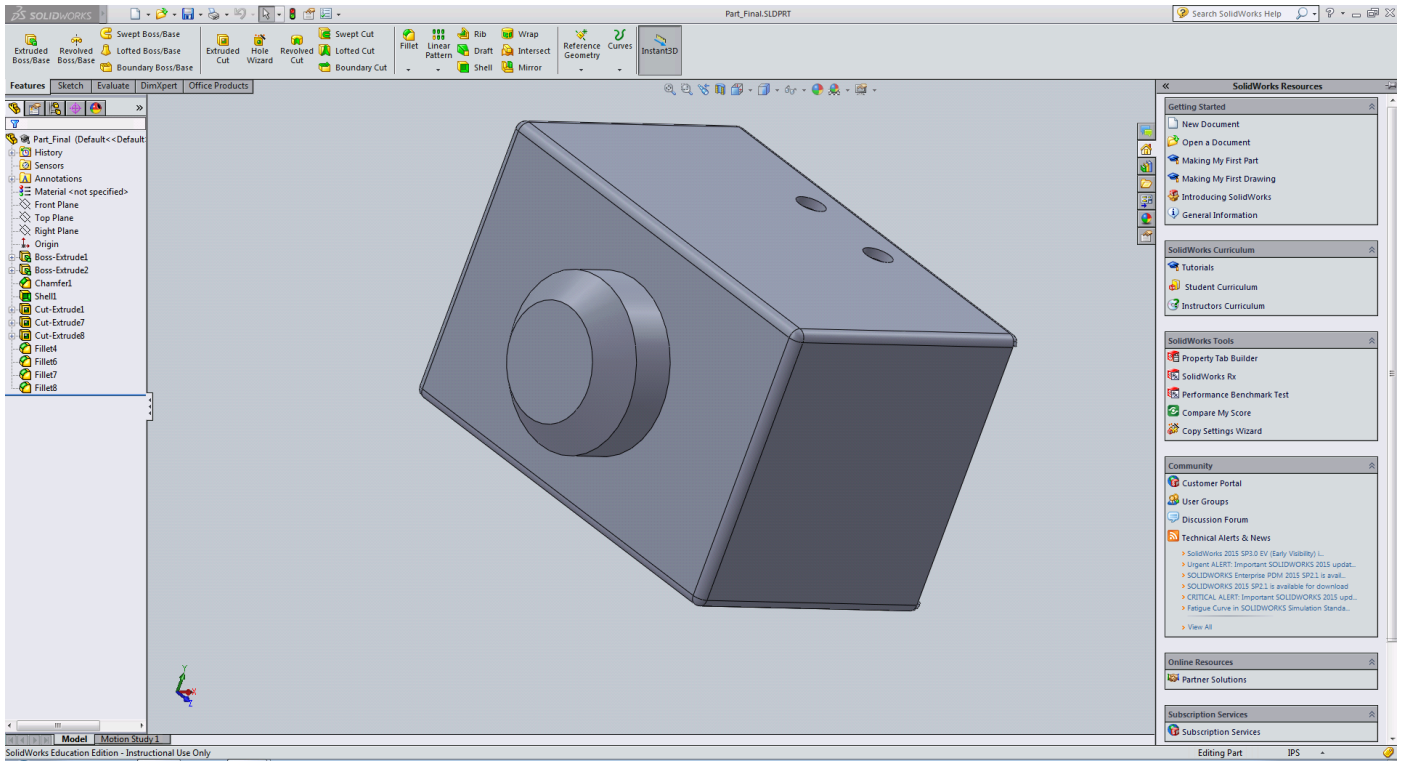


Fig. 13. SolidWorks Lock Design (1).

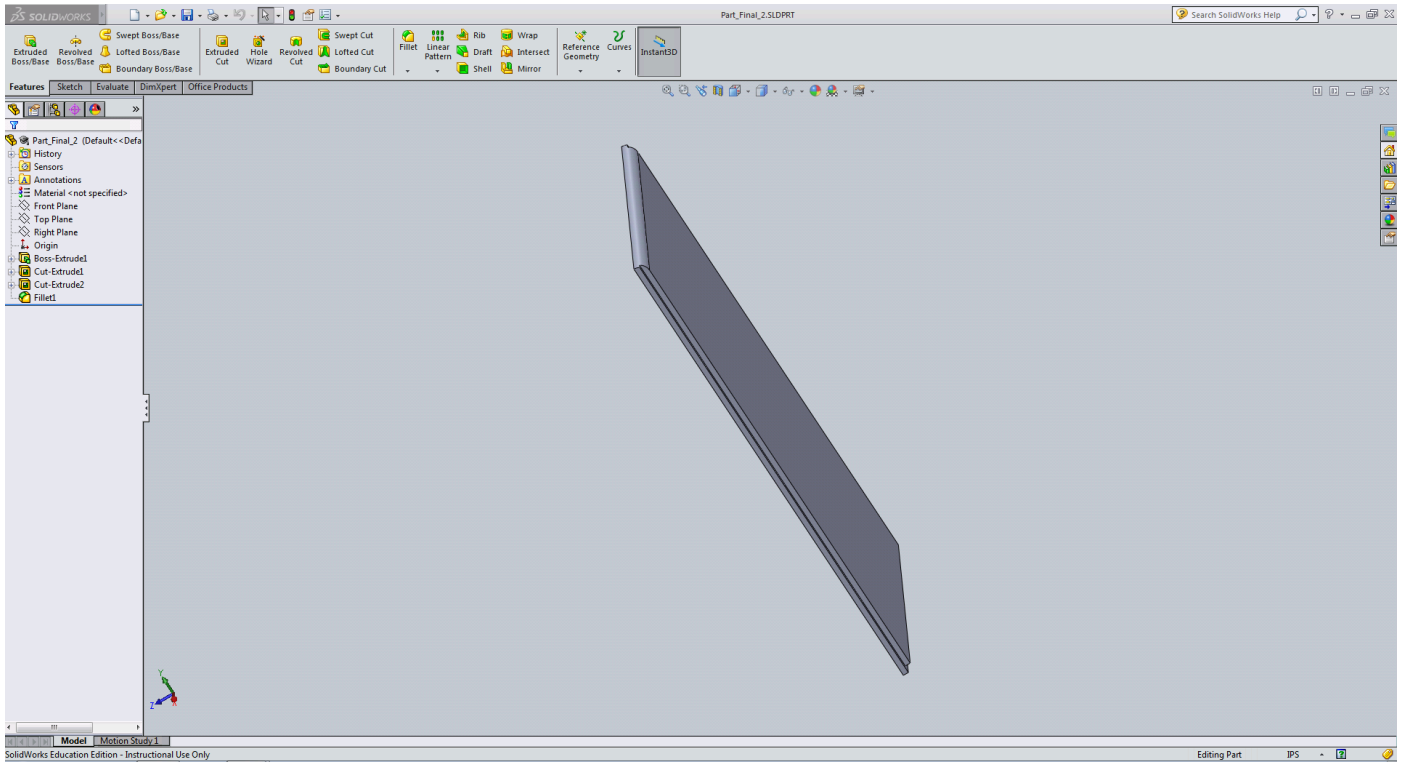


Fig. 14. SolicWorks Lock Design (2).

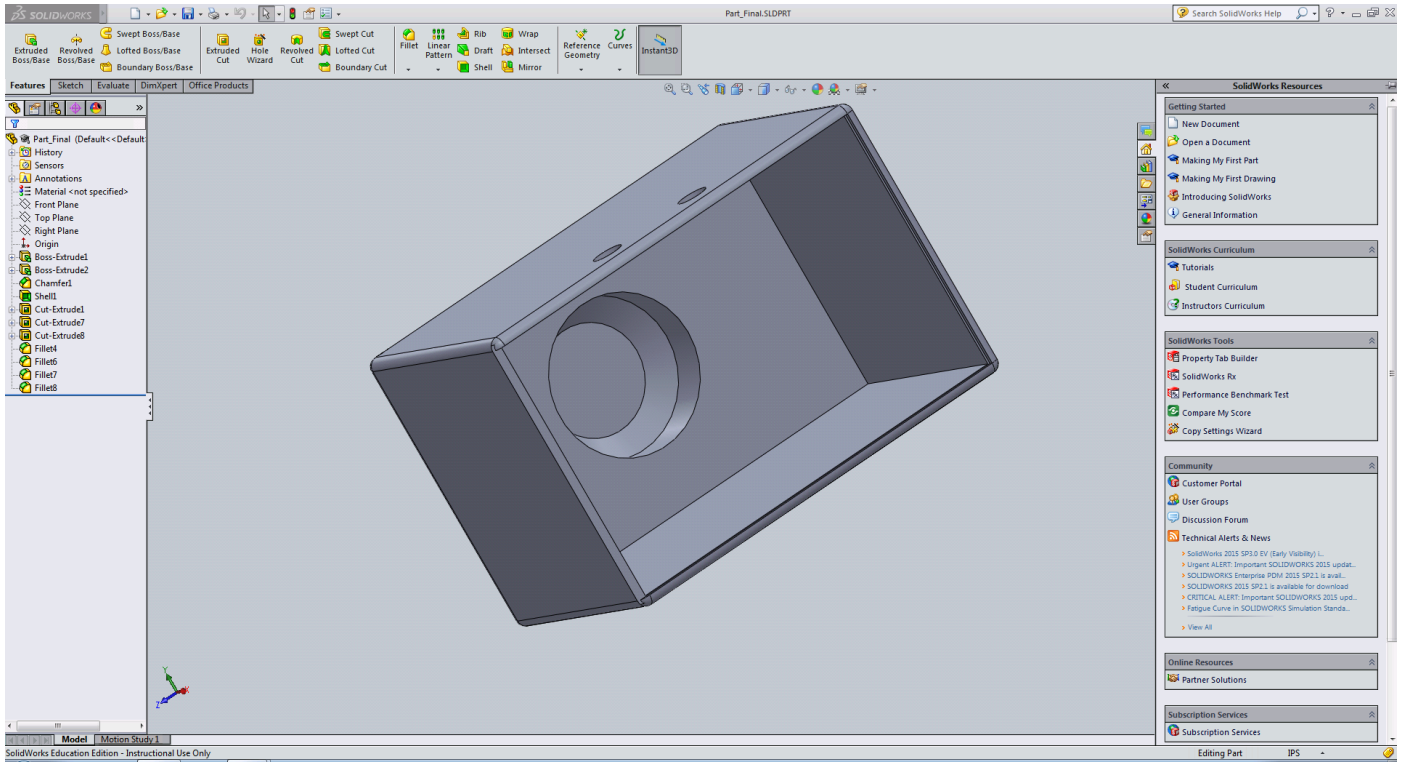
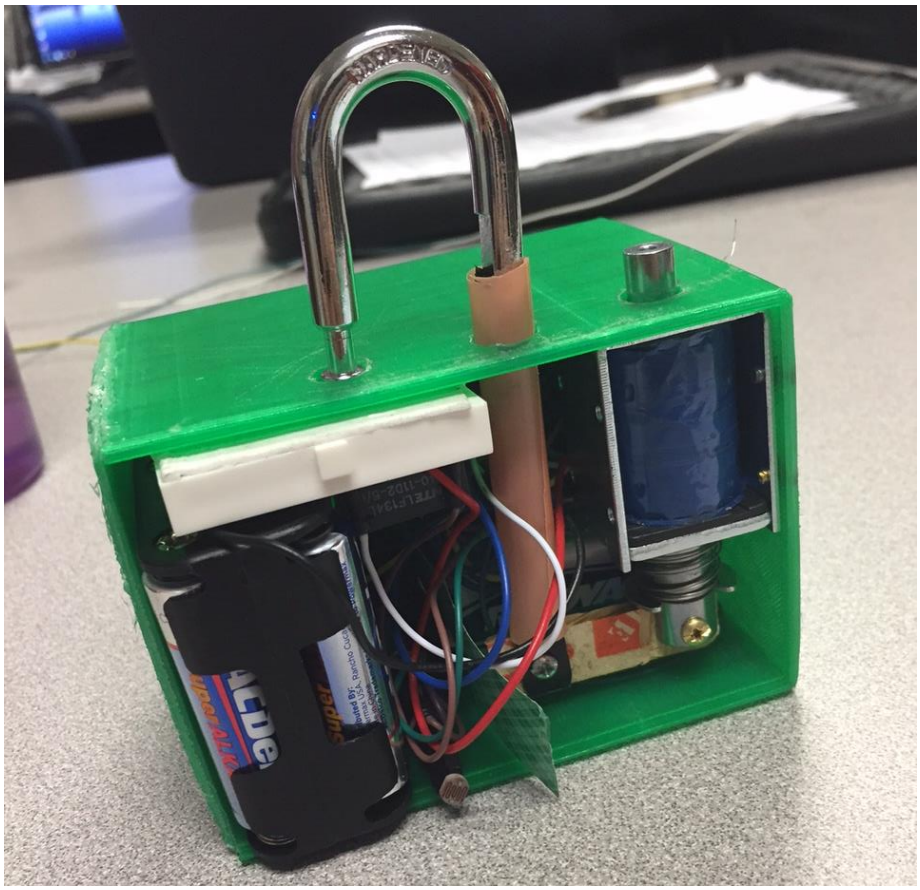


Fig. 15. SolidWorks Lock Design (3).



Figures 13-15 depicts the design we have for our portable lock. It was created using SolidWorks. The idea behind this CAD (Computer-Aided Design) is to essentially use it as a model to create a 3D printed padlock to hold our portable solution. When we use 3D printing, we know it might not be of the strongest material (plastic) but with our resources and time, it was the most efficient route and would only be intended for demonstration purposes. The printed result will serve as a shell and essentially be used to hold our solution without having to run into the issue of metal interfering with our RFID reader. One of the main reasons we opted for 3D printing was to avoid the specific conflict with RFID technology and metal surfaces. That is, metal surfaces block most, if not all, radio frequencies, which would ultimately reduce the effectiveness of our project.

3D printing, also labeled as additive manufacturing, is trending technology that ultimately layers particular material using the assistance of a computer. The process is relatively simple. You design your project/object and save it in a .dwg or required file with the proper extension that would be able to be read by the appropriate 3D printer. 3D printers are available at some of the Sacramento Libraries, which make it very convenient for any designer or engineer.

For our specific design, our sketch consists of cylindrical extrusions from the center box, for the shackle of the padlock. A circular extrusion with fileted edges for the dial center of the design. And lastly, there is the square box for the body of the padlock, which will be hollow to allow proper fitment of all of the hardware. All of the 3D bodies have measured dimensions as well as fileted edges to give it a rounder look to the object.

Figure 16 depicts the finalized product for the mobile lock with all of its components.

XIV. Hardware Test Plan Results

To test the hardware, Team 2 decided to check through multiple predetermined test cases, using the testing document written a few weeks ago. Hardware testing included testing the firmware for our microcontroller, testing card reading, and using stress tests in case of extreme, but possible, events.

A. Arduino Microcontroller Testing

We tested the pins of the microcontroller to see if they were giving out the expected values in terms of voltage.

Component	**Voltage**
LCD	4.96V
A0-5	4.91V ± 0.03V
Reader Power	4.95V
RGB LED	4.99V ± 0.01V
LilyPad Power	4.97V
LilyPad Output Pins	4.93V ± 0.05V

We found that all pins were within our expected values, except for our red LED which we had found to be broken. Our options are to buy a new multicolored LED or to say that while mass production would have the red color, we have a clear color to represent incorrect passwords and the alarm state.

B. Solenoid and Electric Door Strike Testing

We tested the necessary power for the door strike as well as our solenoid cabinet lock. We were trying to see the necessary power to run the system as well as whether there were any problems with the system, as well as making sure that everything is safe. We found that our door would be able to still be open from the inside in the case of a dangerous event, such as a fire, where a card might not be readily on hand. We needed found that 11.93V is what our locks were receiving to open and that the mobile lock only needed the 4.97 volts to open the lock since we have a partial mechanical solution.

C. RFID Reader Testing

We used the Arduino serial monitor to output card information to supplement our testing results. The first reader test we performed was a speed reading test. We wanted to see how long between card reads, and if the data would be input properly. We found there is about a 7.5 second delay in between card reads, and that leaving the card on the reader would work most of the time, while periodically causing errors of the wrong card being read. It is not expected that the user leave the card in front of the reader, so this should not be a problem. We then tested reading multiple cards at the same time. At first we found that the closer card would be read first, but then we found a problem in the code. After we fixed the problem, we found, using the serial monitor, that the reader will read both cards at the same time. Since it reads both, it will not accept either card as a user and sets off the alarm.

D. Power Adaptor Voltage Measurements

We were checking the voltage and saw that we had put the wrong testing status for our adaptor. We wanted 12 volts from the adaptor instead of the written 9.

No Load	Load
17.5V	11.93V

The results came out as we expected for the loaded circuit. Without the load we were a little confused while testing and with the help of a EEE student we figured out our problem. This amount of power was needed for our end devices, the door strike and solenoid to open and close.

E. Other Components/Sensor Testing

For this, we tested the code for problems as well as checked to see if there were any problems with our equipment. We found that after extensive use our LCD

would cease to work and our system would need a reset. To counter this we made sure to have a clear command before each LCD command to reset the LCD properly. This has ended the problem so far and it was honestly a big step in making our project presentable. We also had a problem with our startup code, and the alarm going off unwarrantedly. We found out that since we did not have an initialization of the card code, when it was checked, and found not empty, the alarm would go off as if it had read a bad card. To mitigate this problem we simply cleared the card value before we went through the setup and reading process. Our LED was having problems with the state it should be in at certain times, and we found through the process of checking the voltage as well as changing values in the code, that our red setting on our LED is no longer working. Since we have put a large amount of money into this project already, we have decided to refrain from buying another one, but simply noting that if it was mass production the red LED would work properly. For now we use clear to symbolize our red LED.

F. Power Outage

We checked what would happen to the system during a power outage. Since there would be no power to open locks, or communicate with the server we found that our system would cease to work. There will still be availability to leave the door, and the mobile lock will open and close, but not communicate without Wi-Fi. We also found that after the power returns our code still exists and it should not create a problem with our devices.

XV. Software Test Plan Results

Testing software requires going through all cases. This consists of achieving the expected and unexpected results. To achieve the latter is surely a good thing so that the clients do not discover it for you. Basically, our team is in the alpha stage of testing, which means that the developers of the project are their own testers. After exhausting all our testing techniques, it is possible we will move to the beta stage where we will have others, generally potential customers, test the system. Software testing is still not complete as was expected; although, the results of each test plan so far is reported below:

A. User Interface

The user interface consists of the graphical components that are seen by a user when accessing any part of the webpage. The pages are compilations of HTML or PHP. Being accessed in different ways to see if it continues to display properly each time will test each HTML page. Since most of the web server reflects upon the database, most of the webpages will be PHP. Those specific pages will need to be tested for sampling the text fields and various buttons for correct functionality.

The user interface will continually receive more functionality as time passes. The complexity of the testing will increase as more data tracking is involved along with its algorithms. Currently, there are over 10 unique web pages mainly consisting of PHP. The system is currently in the process of being moved to a public IP. The first thing a user will see when accessing the web page is shown in Figure 3a.

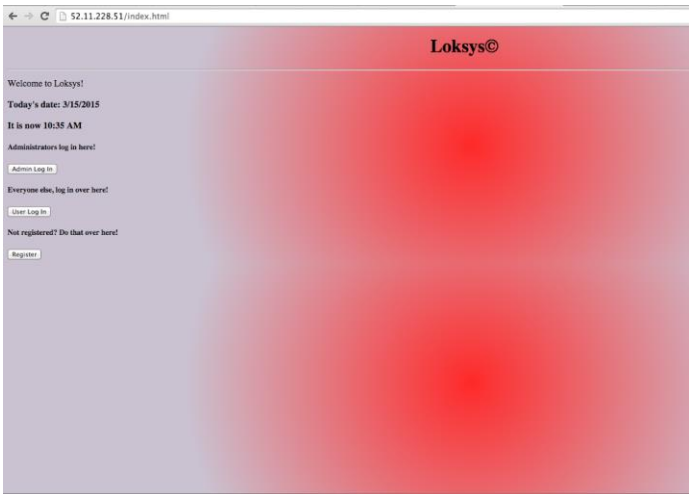


Figure 3a: Index page

Testing of the user interface begins from this page. The goal is to transition around the pages upon the action of a button. The database is also involved when any input is asked from the user. Going through these motions several times, the web pages have not seem to broke yet, although the database had issues with connectivity/interaction on several pages such as the two separate login pages for the admin and user. An admin is considered a registered person who is assigned a special pin. Without that, they cannot access the privileges of an admin. Several tweaks have been made in the code to reflect this. Aside from this as previously stated, user interface testing will be ongoing.

B. Security

Security can be a tricky component to test. Basically, there needs to be attempts to hack or break the web server. When a web page asks for any user input of data, the submission of it usually means a SQL query will be executed. In this instance, SQL injection will be tested to see if any of the data can be exposed. Also, it needs to be certain that only administrators can access the master home page, which gives access to customer information and other admin privileges.

Ten cases have been tested so far to check for any SQL injection vulnerabilities. Injection is a common case of hacking private information found in a database. This is generally done in the login page. The following cases were executed in the user login and admin login pages:

Case 1: Type some random SQL for the input value and see if the server returns an unexpected message.

Email: SELECT email FROM members WHERE id=1

Password: SELECT MD5(password) FROM members WHERE id=1

Case 2: Type an asterisk(*) for the input value

Email: *
Password: (Leave blank)

Case 3: Type a known email as the input and suffix the comment command with --
Email: admin@gmail.com'--
Password: (Leave blank)

Case 4: Type a known email as the input and suffix the comment command with #
Email: admin@gmail.com'#
Password: (Leave blank)

Case 5: Type the following for the password: 'or 1=1-- and use an email you know exists
Email: admin@gmail.com
Password: 'or 1=1

Case 6: Depending on the system, try inputting comment syntax
Email: admin@gmail.com
Password: 'or 1=1 --connect

Case 7: Type the following for the password:' or"='
Email: admin@gmail.com
Password:' or"='

Case 8: Some scripts cannot interpret a newline, check for another query or does the script trim the last line submitted?
Email: admin@gmail.com
Password: ' (new line) OR 1=1--

Case 9: Type the following: %27%20or%20%27%27%3D%27 for the password
Email: admin@gmail.com
Password: %27%20or%20%27%27%3D%27

Case 10: Check that scripts do not validate on password alone
Email: guest@gmail.com
Password: (password that exists in system)

These cases have returned a negative result of SQL injection. Although, these cases are few of many. There will be many more cases to try over time to reinforce the system with better security.

C. *Stress Testing*

In a case where many users are using the product at the same time, there needs to be assurance that the web server can handle the activity simultaneously. Depending on which way the test is simulated; an oscilloscope can be a good determinant. By hooking it up to the breadboard where the RFID reader is connected, frequency can be increased to show a change of state within a period. This will show a clear picture of how well the database can collect and display the data.

Currently, this test is on standby. Other software tests have been placed as higher priority before this one will be tested. It has been decided that a frequency of about 20 Hz (every 20ms) would give a good confirmation that the system can handle stress.

D. *Browser Capability*

There are many browsers to access the Internet. Some are more popular that others, nonetheless, all of them should be able to display the web pages properly. Opening the web pages on each browser will simply test this. If there are any significant issues with display, further testing can be done to see what needs to be done to bring it to acceptable terms.

The web page has been tested on Google Chrome, Mozilla Firefox, Apple Safari and Internet Explorer. Except for Firefox, the web page has executed as expected on the different browsers. Firefox does not seem to accept the .css file for formatting and color. Research is being done to figure out how to correct that flaw. When the issue with Firefox is solved, the testing will continue to other browsers that are not as popular to help us learn about more issues. This is another test will continue to be tested as the web page has more complicated functions.

E. *Error Handling*

A website is capable of producing many errors. It can be user related or system related. In a case where the user is at fault, they need to be redirected to correct themselves. An example would be registering on the website. If there is a requirement on length or what the username should contain, they should be told to try again without kicking them off the page. A system related issue would be if the website has over exceeded its limit on activity. If so, the user needs to notified so they can understand the situation. There will always be many ways to be of error on a website.

To handle errors, the system needs to be ready to receive multiple sorts of wrong inputs. Whether it be length or wrong use of characters, the system needs to know not to accept it and prompt the user that the attempt failed and allow another try. To test this, every function was tested on the web page with different types of input. If a user tries to log in through the admin login page, they will be rejected. The same result occurs if an admin tries to login as a user. Currently, where those logins are rejected, they are taken to an error screen with no indication as to why, but only that the login has failed. Security-wise, it may be better not to give a specific indication as to why, because the system would be giving great feedback to an attempt of hacking. The solution to the error page is currently in thought as how to approach this. Also, the user/admin should be redirected to the homepage after a fail. Some tweaks in the code are being made to correct this.

It was also realized that the requirements of what a username and password should contain were not implemented correctly. For example, the system will register someone who chooses 'admin' for username and 'admin' for password. This was alright in the testing stage for correct

data insertion, but now since that phase has passed, this will be corrected in the code.

In summary, the testing has come a long way so far but there is much more to do. The software system will become much stronger over time as testing is implemented.

A. Data Sheets

1) LCD2004 (HD44780)

HD44780U (LCD-II)

(Dot Matrix Liquid Crystal Display Controller/Driver)

HITACHI

ADE-207-272(Z)
'99.9
Rev. 0.0

Description

The HD44780U dot-matrix liquid crystal display controller and driver LSI displays alphanumerics, Japanese kana characters, and symbols. It can be configured to drive a dot-matrix liquid crystal display under the control of a 4- or 8-bit microprocessor. Since all the functions such as display RAM, character generator, and liquid crystal driver, required for driving a dot-matrix liquid crystal display are internally provided on one chip, a minimal system can be interfaced with this controller/driver.

A single HD44780U can display up to one 8-character line or two 8-character lines.

The HD44780U has pin function compatibility with the HD44780S which allows the user to easily replace an LCD-II with an HD44780U. The HD44780U character generator ROM is extended to generate 208 5 × 8 dot character fonts and 32 5 × 10 dot character fonts for a total of 240 different character fonts.

The low power supply (2.7V to 5.5V) of the HD44780U is suitable for any portable battery-driven product requiring low power dissipation.

Features

- 5 × 8 and 5 × 10 dot matrix possible
- Low power operation support:
  — 2.7 to 5.5V
- Wide range of liquid crystal display driver power
  — 3.0 to 11V
- Liquid crystal drive waveform
  — A (One line frequency AC waveform)
- Correspond to high speed MPU bus interface
  — 2 MHz (when V_{CC} = 5V)
- 4-bit or 8-bit MPU interface enabled
- 80 × 8-bit display RAM (80 characters max.)
- 9,920-bit character generator ROM for a total of 240 character fonts
  — 208 character fonts (5 × 8 dot)
  — 32 character fonts (5 × 10 dot)

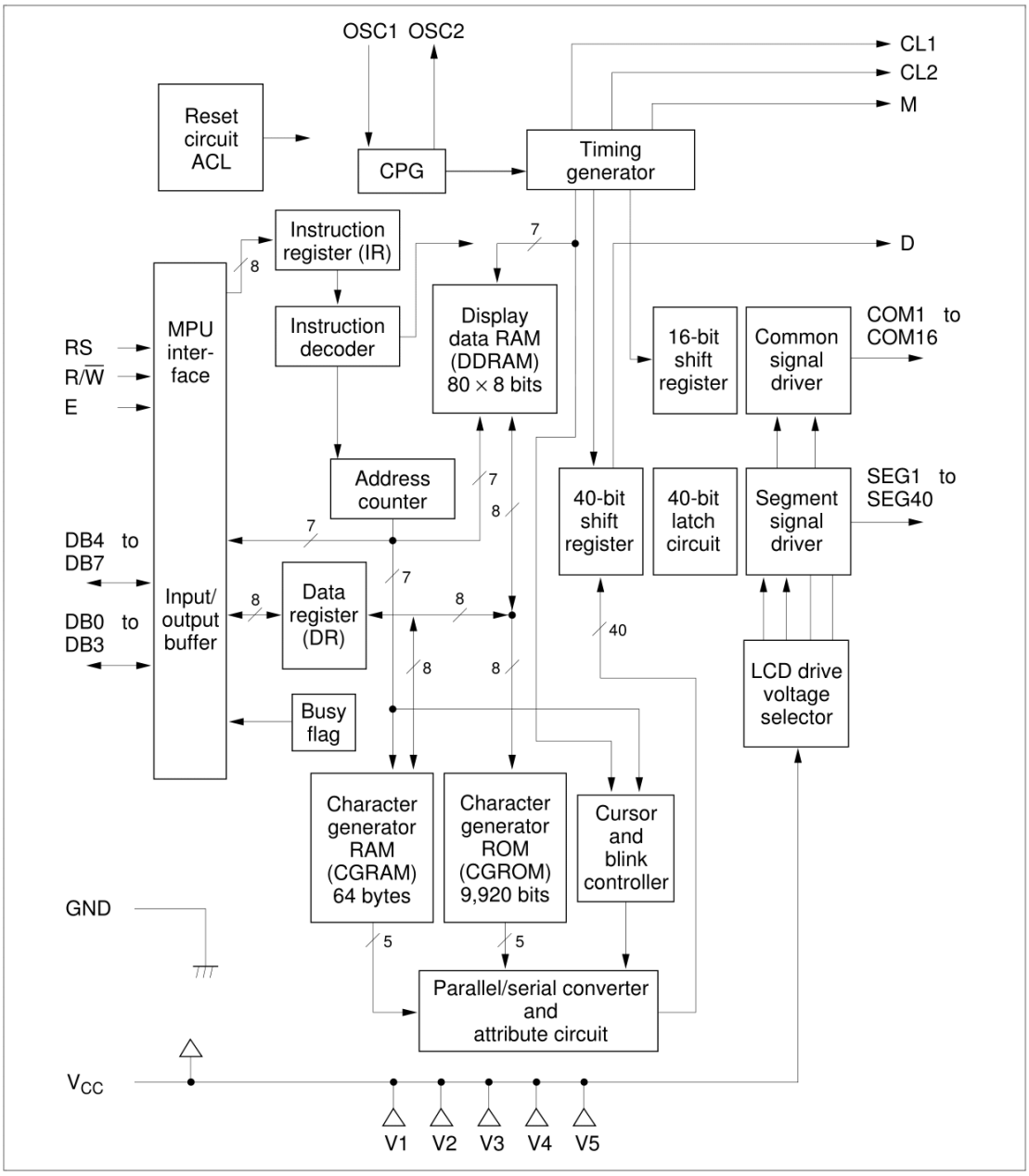
- 64 × 8-bit character generator RAM
  - — 8 character fonts (5 × 8 dot)
  - — 4 character fonts (5 × 10 dot)
- 16-common × 40-segment liquid crystal display driver
- Programmable duty cycles
  - — 1/8 for one line of 5 × 8 dots with cursor
  - — 1/11 for one line of 5 × 10 dots with cursor
  - — 1/16 for two lines of 5 × 8 dots with cursor
- Wide range of instruction functions:
  - — Display clear, cursor home, display on/off, cursor on/off, display character blink, cursor shift, display shift
- Pin function compatibility with HD44780S
- Automatic reset circuit that initializes the controller/driver after power on
- Internal oscillator with external resistors
- Low power consumption

Ordering Information

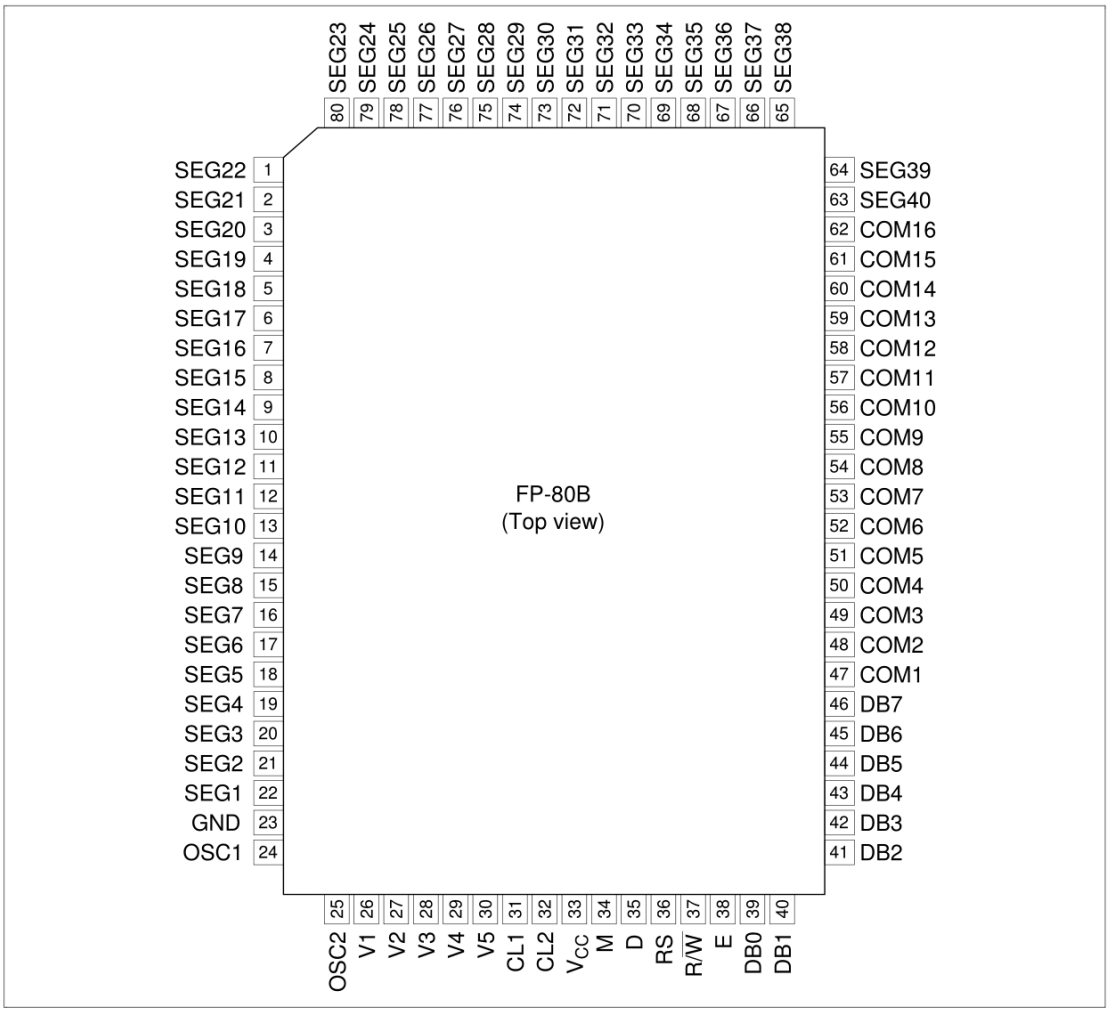
Type No.	Package	CGROM
HD44780UA00FS HCD44780UA00 HD44780UA00TF	FP-80B Chip TFP-80F	Japanese standard font
HD44780UA02FS HCD44780UA02 HD44780UA02TF	FP-80B Chip TFP-80F	European standard font
HD44780UBxxFS HCD44780UBxx HD44780UBxxTF	FP-80B Chip TFP-80F	Custom font

Note: xx: ROM code No.

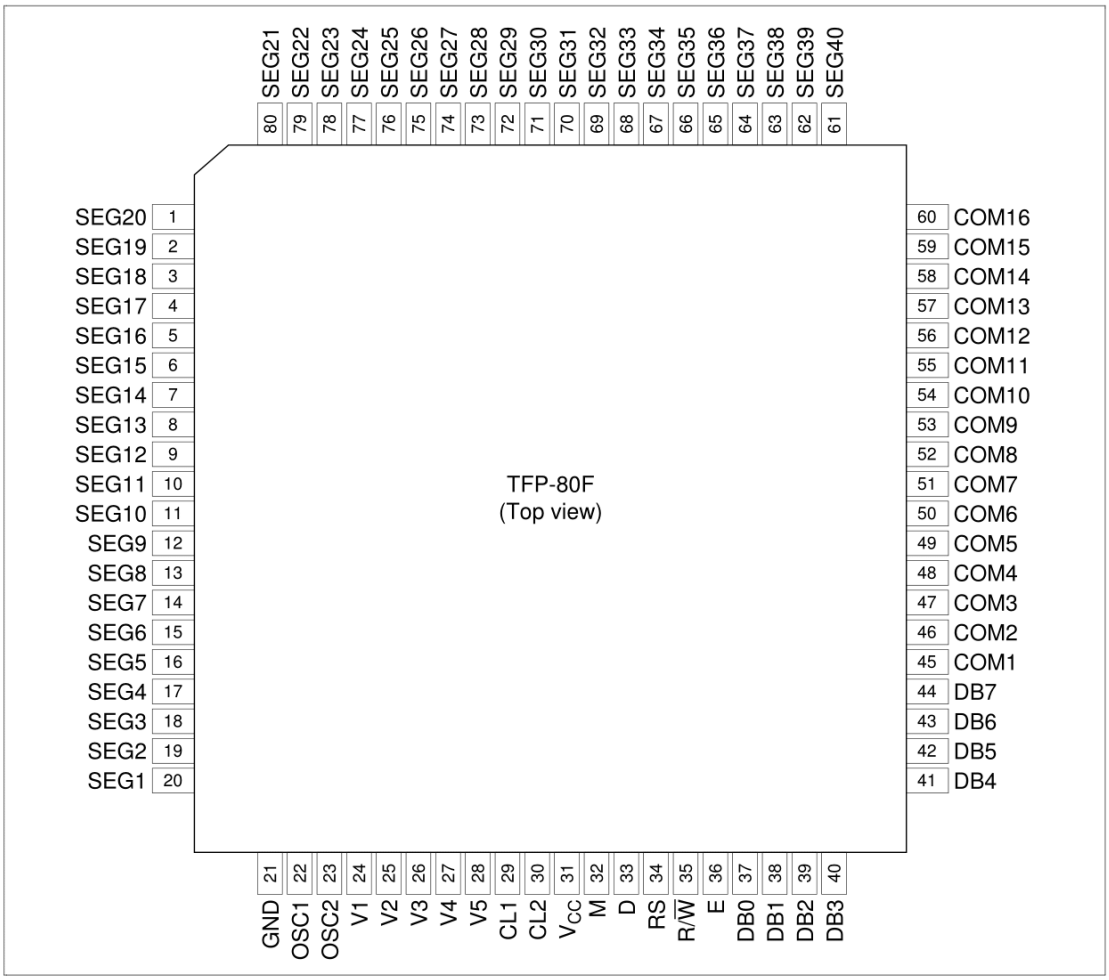
HD44780U Block Diagram



HD44780U Pin Arrangement (FP-80B)

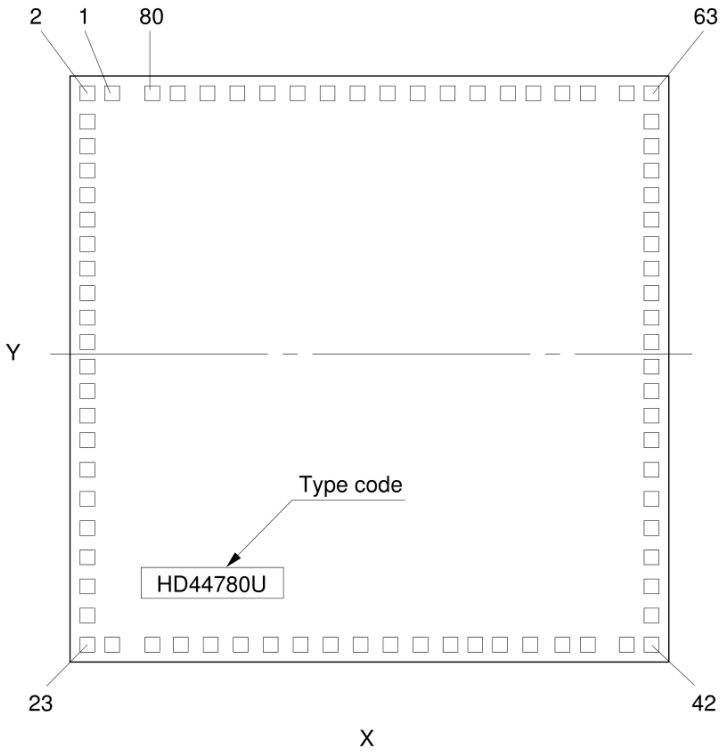


HD44780U Pin Arrangement (TFP-80F)



HD44780U Pad Arrangement

Chip size: 4.90×4.90 mm^2
Coordinate: Pad center (μm)
Origin: Chip center
Pad size: 114×114 μm^2



HCD44780U Pad Location Coordinates

		Coordinate				Coordinate	
Pad No.	Function	X (um)	Y (um)	Pad No.	Function	X (um)	Y (um)
1	SEG22	–2100	2313	41	DB2	2070	–2290
2	SEG21	–2280	2313	42	DB3	2260	–2290
3	SEG20	–2313	2089	43	DB4	2290	–2099
4	SEG19	–2313	1833	44	DB5	2290	–1883
5	SEG18	–2313	1617	45	DB6	2290	–1667
6	SEG17	–2313	1401	46	DB7	2290	–1452
7	SEG16	–2313	1186	47	COM1	2313	–1186
8	SEG15	–2313	970	48	COM2	2313	–970
9	SEG14	–2313	755	49	COM3	2313	–755
10	SEG13	–2313	539	50	COM4	2313	–539
11	SEG12	–2313	323	51	COM5	2313	–323
12	SEG11	–2313	108	52	COM6	2313	–108
13	SEG10	–2313	–108	53	COM7	2313	108
14	SEG9	–2313	–323	54	COM8	2313	323
15	SEG8	–2313	–539	55	COM9	2313	539
16	SEG7	–2313	–755	56	COM10	2313	755
17	SEG6	–2313	–970	57	COM11	2313	970
18	SEG5	–2313	–1186	58	COM12	2313	1186
19	SEG4	–2313	–1401	59	COM13	2313	1401
20	SEG3	–2313	–1617	60	COM14	2313	1617
21	SEG2	–2313	–1833	61	COM15	2313	1833
22	SEG1	–2313	–2073	62	COM16	2313	2095
23	GND	–2280	–2290	63	SEG40	2296	2313
24	OSC1	–2080	–2290	64	SEG39	2100	2313
25	OSC2	–1749	–2290	65	SEG38	1617	2313
26	V1	–1550	–2290	66	SEG37	1401	2313
27	V2	–1268	–2290	67	SEG36	1186	2313
28	V3	–941	–2290	68	SEG35	970	2313
29	V4	–623	–2290	69	SEG34	755	2313
30	V5	–304	–2290	70	SEG33	539	2313
31	CL1	–48	–2290	71	SEG32	323	2313
32	CL2	142	–2290	72	SEG31	108	2313
33	V_{CC}	309	–2290	73	SEG30	–108	2313
34	M	475	–2290	74	SEG29	–323	2313
35	D	665	–2290	75	SEG28	–539	2313
36	RS	832	–2290	76	SEG27	–755	2313
37	R/$\overline{W}$	1022	–2290	77	SEG26	–970	2313
38	E	1204	–2290	78	SEG25	–1186	2313
39	DB0	1454	–2290	79	SEG24	–1401	2313
40	DB1	1684	–2290	80	SEG23	–1617	2313

Pin Functions

Signal	No. of Lines	I/O	Device Interfaced with	Function
RS	1	I	MPU	Selects registers. 0: Instruction register (for write) Busy flag: address counter (for read) 1: Data register (for write and read)
R/$\overline{W}$	1	I	MPU	Selects read or write. 0: Write 1: Read
E	1	I	MPU	Starts data read/write.
DB4 to DB7	4	I/O	MPU	Four high order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. DB7 can be used as a busy flag.
DB0 to DB3	4	I/O	MPU	Four low order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. These pins are not used during 4-bit operation.
CL1	1	O	Extension driver	Clock to latch serial data D sent to the extension driver
CL2	1	O	Extension driver	Clock to shift serial data D
M	1	O	Extension driver	Switch signal for converting the liquid crystal drive waveform to AC
D	1	O	Extension driver	Character pattern data corresponding to each segment signal
COM1 to COM16	16	O	LCD	Common signals that are not used are changed to non-selection waveforms. COM9 to COM16 are non-selection waveforms at 1/8 duty factor and COM12 to COM16 are non-selection waveforms at 1/11 duty factor.
SEG1 to SEG40	40	O	LCD	Segment signals
V1 to V5	5	—	Power supply	Power supply for LCD drive V_{CC} –V5 = 11 V (max)
V_{CC}, GND	2	—	Power supply	V_{CC}: 2.7V to 5.5V, GND: 0V
OSC1, OSC2	2	—	Oscillation resistor clock	When crystal oscillation is performed, a resistor must be connected externally. When the pin input is an external clock, it must be input to OSC1.

Function Description

Registers

The HD44780U has two 8-bit registers, an instruction register (IR) and a data register (DR).

The IR stores instruction codes, such as display clear and cursor shift, and address information for display data RAM (DDRAM) and character generator RAM (CGRAM). The IR can only be written from the MPU.

The DR temporarily stores data to be written into DDRAM or CGRAM and temporarily stores data to be read from DDRAM or CGRAM. Data written into the DR from the MPU is automatically written into DDRAM or CGRAM by an internal operation. The DR is also used for data storage when reading data from DDRAM or CGRAM. When address information is written into the IR, data is read and then stored into the DR from DDRAM or CGRAM by an internal operation. Data transfer between the MPU is then completed when the MPU reads the DR. After the read, data in DDRAM or CGRAM at the next address is sent to the DR for the next read from the MPU. By the register selector (RS) signal, these two registers can be selected (Table 1).

Busy Flag (BF)

When the busy flag is 1, the HD44780U is in the internal operation mode, and the next instruction will not be accepted. When RS = 0 and R/$\overline{W}$ = 1 (Table 1), the busy flag is output to DB7. The next instruction must be written after ensuring that the busy flag is 0.

Address Counter (AC)

The address counter (AC) assigns addresses to both DDRAM and CGRAM. When an address of an instruction is written into the IR, the address information is sent from the IR to the AC. Selection of either DDRAM or CGRAM is also determined concurrently by the instruction.

After writing into (reading from) DDRAM or CGRAM, the AC is automatically incremented by 1 (decremented by 1). The AC contents are then output to DB0 to DB6 when RS = 0 and R/$\overline{W}$ = 1 (Table 1).

Table 1 Register Selection

RS	R/$\overline{W}$	Operation
0	0	IR write as an internal operation (display clear, etc.)
0	1	Read busy flag (DB7) and address counter (DB0 to DB6)
1	0	DR write as an internal operation (DR to DDRAM or CGRAM)
1	1	DR read as an internal operation (DDRAM or CGRAM to DR)

Display Data RAM (DDRAM)

Display data RAM (DDRAM) stores display data represented in 8-bit character codes. Its extended capacity is 80 × 8 bits, or 80 characters. The area in display data RAM (DDRAM) that is not used for display can be used as general data RAM. See Figure 1 for the relationships between DDRAM addresses and positions on the liquid crystal display.

The DDRAM address (A_{DD}) is set in the address counter (AC) as hexadecimal.

- 1-line display (N = 0) (Figure 2)
  - When there are fewer than 80 display characters, the display begins at the head position. For example, if using only the HD44780, 8 characters are displayed. See Figure 3.

    When the display shift operation is performed, the DDRAM address shifts. See Figure 3.

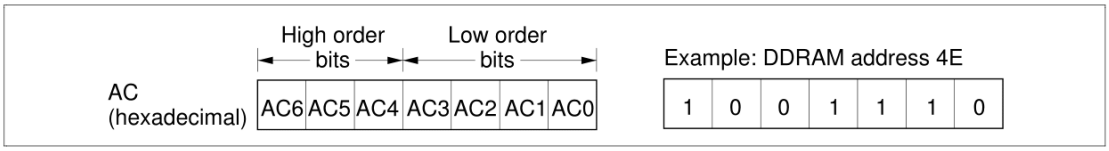


Figure 1 DDRAM Address

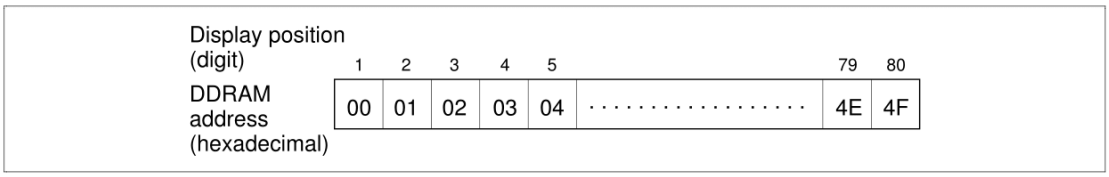


Figure 2 1-Line Display

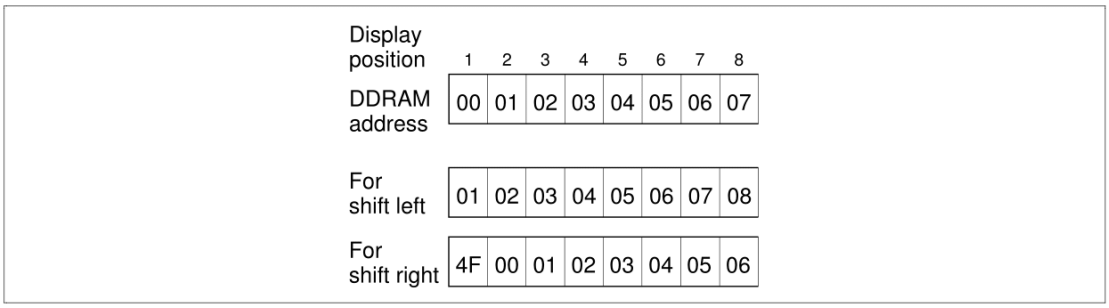


Figure 3 1-Line by 8-Character Display Example

- 2-line display (N = 1) (Figure 4)

  — Case 1: When the number of display characters is less than 40 × 2 lines, the two lines are displayed from the head. Note that the first line end address and the second line start address are not consecutive. For example, when just the HD44780 is used, 8 characters × 2 lines are displayed. See Figure 5.

  When display shift operation is performed, the DDRAM address shifts. See Figure 5.

Display position	1	2	3	4	5	·····	39	40
DDRAM address (hexadecimal)	00	01	02	03	04	·····	26	27
	40	41	42	43	44	·····	66	67

Figure 4 2-Line Display

Display position	1	2	3	4	5	6	7	8
DDRAM address	00	01	02	03	04	05	06	07
	40	41	42	43	44	45	46	47
For shift left	01	02	03	04	05	06	07	08
	41	42	43	44	45	46	47	48
For shift right	27	00	01	02	03	04	05	06
	67	40	41	42	43	44	45	46

Figure 5 2-Line by 8-Character Display Example

— Case 2: For a 16-character × 2-line display, the HD44780 can be extended using one 40-output extension driver. See Figure 6.

When display shift operation is performed, the DDRAM address shifts. See Figure 6.

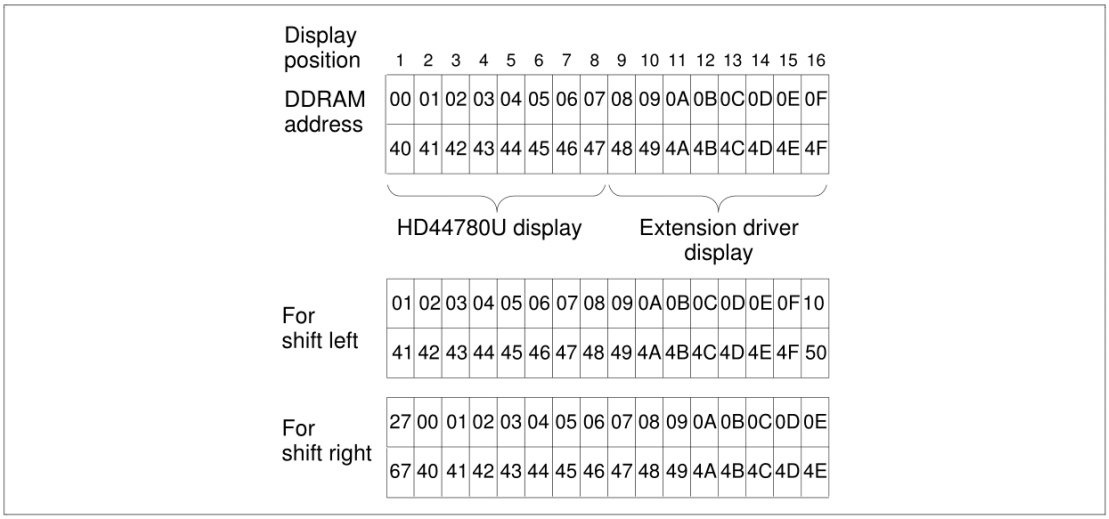


Figure 6 2-Line by 16-Character Display Example

Character Generator ROM (CGROM)

The character generator ROM generates 5 × 8 dot or 5 × 10 dot character patterns from 8-bit character codes (Table 4). It can generate 208 5 × 8 dot character patterns and 32 5 × 10 dot character patterns. User-defined character patterns are also available by mask-programmed ROM.

Character Generator RAM (CGRAM)

In the character generator RAM, the user can rewrite character patterns by program. For 5 × 8 dots, eight character patterns can be written, and for 5 × 10 dots, four character patterns can be written.

Write into DDRAM the character codes at the addresses shown as the left column of Table 4 to show the character patterns stored in CGRAM.

See Table 5 for the relationship between CGRAM addresses and data and display patterns.

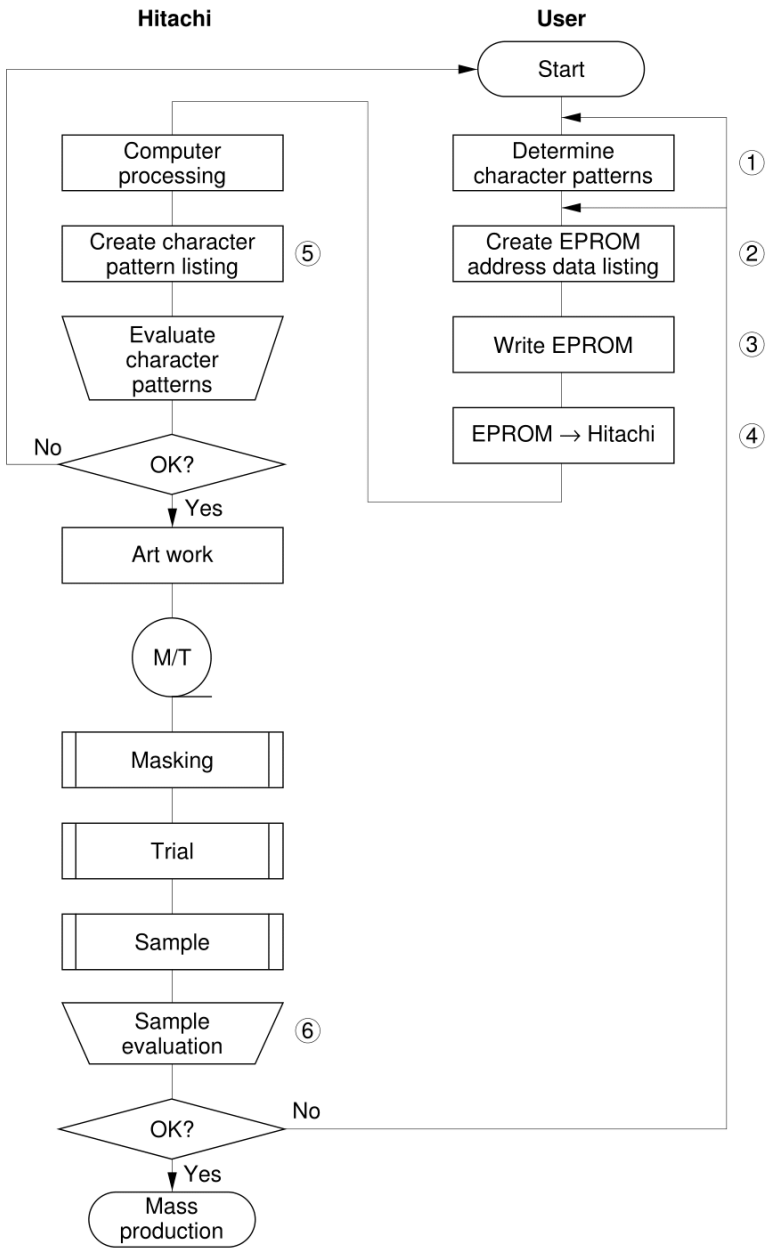
Areas that are not used for display can be used as general data RAM.

Modifying Character Patterns

- Character pattern development procedure

The following operations correspond to the numbers listed in Figure 7:

1. Determine the correspondence between character codes and character patterns.
2. Create a listing indicating the correspondence between EPROM addresses and data.
3. Program the character patterns into the EPROM.
4. Send the EPROM to Hitachi.
5. Computer processing on the EPROM is performed at Hitachi to create a character pattern listing, which is sent to the user.
6. If there are no problems within the character pattern listing, a trial LSI is created at Hitachi and samples are sent to the user for evaluation. When it is confirmed by the user that the character patterns are correctly written, mass production of the LSI proceeds at Hitachi.



Note: For a description of the numbers used in this figure, refer to the preceding page.

Figure 7 Character Pattern Development Procedure

- Programming character patterns

  This section explains the correspondence between addresses and data used to program character patterns in EPROM. The HD44780U character generator ROM can generate 208 5 × 8 dot character patterns and 32 5 × 10 dot character patterns for a total of 240 different character patterns.

  — Character patterns

  EPROM address data and character pattern data correspond with each other to form a 5 × 8 or 5 × 10 dot character pattern (Tables 2 and 3).

Table 2 Example of Correspondence between EPROM Address Data and Character Pattern (5 × 8 Dots)

EPROM Address												Data					
A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	O4	O3	O2	O1	LSB O0	
								0	0	0	0	1	0	0	0	0	
								0	0	0	1	1	0	0	0	0	
								0	0	1	0	1	0	1	1	0	
								0	0	1	1	1	1	0	0	1	
								0	1	0	0	1	0	0	0	1	
								0	1	0	1	1	0	0	0	1	
								0	1	1	0	1	1	1	1	0	
0	1	1	0	0	0	1	0	0	1	1	1	0	0	0	0	0	← Cursor position
								1	0	0	0	0	0	0	0	0	
								1	0	0	1	0	0	0	0	0	
								1	0	1	0	0	0	0	0	0	
								1	0	1	1	0	0	0	0	0	
								1	1	0	0	0	0	0	0	0	
								1	1	0	1	0	0	0	0	0	
								1	1	1	0	0	0	0	0	0	
								1	1	1	1	0	0	0	0	0	
Character code								Line position									

Notes:
1. EPROM addresses A11 to A4 correspond to a character code.
2. EPROM addresses A3 to A0 specify a line position of the character pattern.
3. EPROM data O4 to O0 correspond to character pattern data.
4. EPROM data O5 to O7 must be specified as 0.
5. A lit display position (black) corresponds to a 1.
6. Line 9 and the following lines must be blanked with 0s for a 5 × 8 dot character fonts.

— Handling unused character patterns

1. EPROM data outside the character pattern area: Always input 0s.
2. EPROM data in CGRAM area: Always input 0s. (Input 0s to EPROM addresses 00H to FFH.)
3. EPROM data used when the user does not use any HD44780U character pattern: According to the user application, handled in one of the two ways listed as follows.
   a. When unused character patterns are not programmed: If an unused character code is written into DDRAM, all its dots are lit. By not programing a character pattern, all of its bits become lit. (This is due to the EPROM being filled with 1s after it is erased.)
   b. When unused character patterns are programmed as 0s: Nothing is displayed even if unused character codes are written into DDRAM. (This is equivalent to a space.)

Table 3 Example of Correspondence between EPROM Address Data and Character Pattern (5 × 10 Dots)

EPROM Address												Data					
A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	O4	O3	O2	O1	LSB O0	
								0	0	0	0	0	0	0	0	0	
								0	0	0	1	0	0	0	0	0	
								0	0	1	0	0	1	1	0	1	
								0	0	1	1	1	0	0	1	1	
								0	1	0	0	1	0	0	0	1	
								0	1	0	1	1	0	0	0	1	
								0	1	1	0	0	1	1	1	1	
0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	
								1	0	0	0	0	0	0	0	1	
								1	0	0	1	0	0	0	0	1	
								1	0	1	0	0	0	0	0	0	← Cursor position
								1	0	1	1	0	0	0	0	0	
								1	1	0	0	0	0	0	0	0	
								1	1	0	1	0	0	0	0	0	
								1	1	1	0	0	0	0	0	0	
								1	1	1	1	0	0	0	0	0	

Character code — Line position

Notes: 1. EPROM addresses A11 to A3 correspond to a character code.
2. EPROM addresses A3 to A0 specify a line position of the character pattern.
3. EPROM data O4 to O0 correspond to character pattern data.
4. EPROM data O5 to O7 must be specified as 0.
5. A lit display position (black) corresponds to a 1.
6. Line 11 and the following lines must be blanked with 0s for a 5 × 10 dot character fonts.

Table 4 Correspondence between Character Codes and Character Patterns (ROM Code: A00)

Upper 4 Bits / Lower 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)															
xxxx0001	(2)															
xxxx0010	(3)															
xxxx0011	(4)															
xxxx0100	(5)															
xxxx0101	(6)															
xxxx0110	(7)															
xxxx0111	(8)															
xxxx1000	(1)															
xxxx1001	(2)															
xxxx1010	(3)															
xxxx1011	(4)															
xxxx1100	(5)															
xxxx1101	(6)															
xxxx1110	(7)															
xxxx1111	(8)															

Note: The user can specify any pattern for character-generator RAM.

Table 4 Correspondence between Character Codes and Character Patterns (ROM Code: A02)

Lower 4 Bits \ Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)															
xxxx0001	(2)															
xxxx0010	(3)															
xxxx0011	(4)															
xxxx0100	(5)															
xxxx0101	(6)															
xxxx0110	(7)															
xxxx0111	(8)															
xxxx1000	(1)															
xxxx1001	(2)															
xxxx1010	(3)															
xxxx1011	(4)															
xxxx1100	(5)															
xxxx1101	(6)															
xxxx1110	(7)															
xxxx1111	(8)															

Table 5 Relationship between CGRAM Addresses, Character Codes (DDRAM) and Character Patterns (CGRAM Data)

For 5 × 8 dot character patterns

Character Codes (DDRAM data)	CGRAM Address	Character Patterns (CGRAM data)	
7 6 5 4 3 2 1 0 High Low	5 4 3 2 1 0 High Low	7 6 5 4 3 2 1 0 High Low	
0 0 0 0 * 0 0 0	0 0 0	0 0 0 * * * 1 1 1 1 0 0 0 1 1 0 0 0 1 0 1 0 1 0 0 0 1 0 1 1 1 1 1 1 0 1 0 0 1 0 1 0 0 1 0 1 1 0 0 1 0 1 1 0 1 0 0 0 1 1 1 1 * * * 0 0 0 0 0	Character pattern (1) Cursor position
0 0 0 0 * 0 0 1	0 0 1	0 0 0 * * * 1 0 0 0 1 0 0 1 0 1 0 1 0 0 1 0 1 1 1 1 1 0 1 1 0 0 1 0 0 1 0 0 1 1 1 1 1 1 0 1 0 0 1 0 0 1 1 0 0 0 1 0 0 1 1 1 * * * 0 0 0 0 0	Character pattern (2) Cursor position
	0 0 0 0 0 1	* * *	
0 0 0 0 * 1 1 1	1 1 1 1 0 0 1 0 1 1 1 0 1 1 1	* * *	

Notes: 1. Character code bits 0 to 2 correspond to CGRAM address bits 3 to 5 (3 bits: 8 types).

2. CGRAM address bits 0 to 2 designate the character pattern line position. The 8th line is the cursor position and its display is formed by a logical OR with the cursor.

   Maintain the 8th line data, corresponding to the cursor display position, at 0 as the cursor display.

   If the 8th line data is 1, 1 bits will light up the 8th line regardless of the cursor presence.

3. Character pattern row positions correspond to CGRAM data bits 0 to 4 (bit 4 being at the left).

4. As shown Table 5, CGRAM character patterns are selected when character code bits 4 to 7 are all 0. However, since character code bit 3 has no effect, the R display example above can be selected by either character code 00H or 08H.

5. 1 for CGRAM data corresponds to display selection and 0 to non-selection.

* Indicates no effect.

Table 5 Relationship between CGRAM Addresses, Character Codes (DDRAM) and Character Patterns (CGRAM Data) (cont)

For 5 × 10 dot character patterns

Character Codes (DDRAM data)	CGRAM Address	Character Patterns (CGRAM data)	
7 6 5 4 3 2 1 0	5 4 3 2 1 0	7 6 5 4 3 2 1 0	
High Low	High Low	High Low	
0 0 0 0 * 0 0 *	0 0 0 0 0 0	* * * 0 0 0 0 0	
	0 0 0 1	0 0 0 0 0	
	0 0 1 0	1 0 1 1 0	
	0 0 1 1	1 1 0 0 1	
	0 1 0 0	1 0 0 0 1	Character pattern
	0 1 0 1	1 0 0 0 1	
	0 1 1 0	1 1 1 1 0	
	0 1 1 1	1 0 0 0 0	
	1 0 0 0	1 0 0 0 0	
	1 0 0 1	1 0 0 0 0	
	1 0 1 0	* * * 0 0 0 0 0	Cursor position
	1 0 1 1	* * * * * * * *	
	1 1 0 0		
	1 1 0 1		
	1 1 1 0		
	1 1 1 1	* * * * * * * *	
	0 0 0 0	* * *	
	0 0 0 1		
0 0 0 0 * 1 1 *	1 1 1 0 0 1		
	1 0 1 0	* * *	
	1 0 1 1	* * * * * * * *	
	1 1 0 0		
	1 1 0 1		
	1 1 1 0		
	1 1 1 1	* * * * * * * *	

Notes: 1. Character code bits 1 and 2 correspond to CGRAM address bits 4 and 5 (2 bits: 4 types).

2. CGRAM address bits 0 to 3 designate the character pattern line position. The 11th line is the cursor position and its display is formed by a logical OR with the cursor.

   Maintain the 11th line data corresponding to the cursor display positon at 0 as the cursor display.

   If the 11th line data is “1”, “1” bits will light up the 11th line regardless of the cursor presence.

   Since lines 12 to 16 are not used for display, they can be used for general data RAM.

3. Character pattern row positions are the same as 5 × 8 dot character pattern positions.

4. CGRAM character patterns are selected when character code bits 4 to 7 are all 0.

   However, since character code bits 0 and 3 have no effect, the P display example above can be selected by character codes 00H, 01H, 08H, and 09H.

5. 1 for CGRAM data corresponds to display selection and 0 to non-selection.

* Indicates no effect.

Timing Generation Circuit

The timing generation circuit generates timing signals for the operation of internal circuits such as DDRAM, CGROM and CGRAM. RAM read timing for display and internal operation timing by MPU access are generated separately to avoid interfering with each other. Therefore, when writing data to DDRAM, for example, there will be no undesirable interferences, such as flickering, in areas other than the display area.

Liquid Crystal Display Driver Circuit

The liquid crystal display driver circuit consists of 16 common signal drivers and 40 segment signal drivers. When the character font and number of lines are selected by a program, the required common signal drivers automatically output drive waveforms, while the other common signal drivers continue to output non-selection waveforms.

Sending serial data always starts at the display data character pattern corresponding to the last address of the display data RAM (DDRAM).

Since serial data is latched when the display data character pattern corresponding to the starting address enters the internal shift register, the HD44780U drives from the head display.

Cursor/Blink Control Circuit

The cursor/blink control circuit generates the cursor or character blinking. The cursor or the blinking will appear with the digit located at the display data RAM (DDRAM) address set in the address counter (AC).

For example (Figure 8), when the address counter is 08H, the cursor position is displayed at DDRAM address 08H.

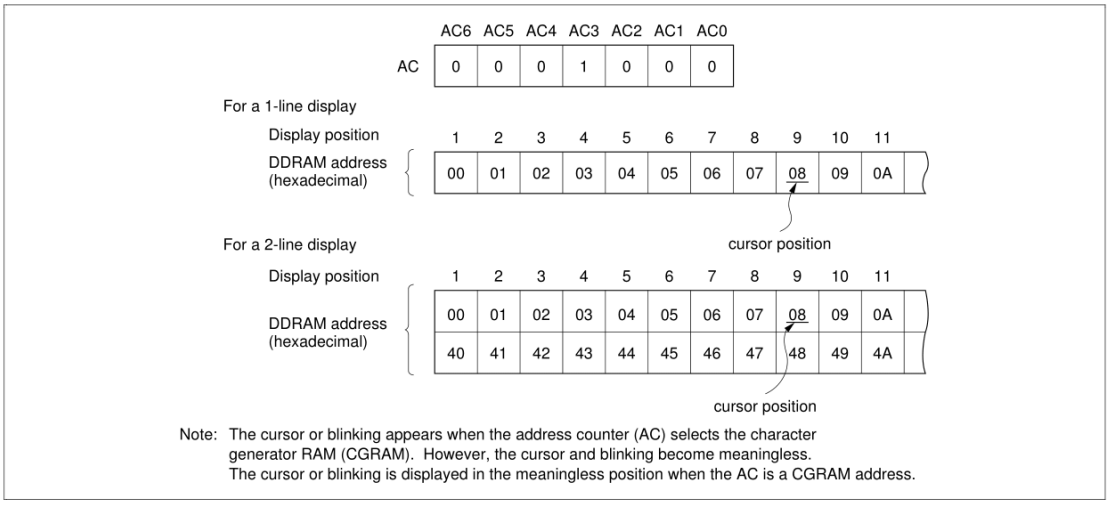


Figure 8 Cursor/Blink Display Example

Interfacing to the MPU

The HD44780U can send data in either two 4-bit operations or one 8-bit operation, thus allowing interfacing with 4- or 8-bit MPUs.

- For 4-bit interface data, only four bus lines (DB4 to DB7) are used for transfer. Bus lines DB0 to DB3 are disabled. The data transfer between the HD44780U and the MPU is completed after the 4-bit data has been transferred twice. As for the order of data transfer, the four high order bits (for 8-bit operation, DB4 to DB7) are transferred before the four low order bits (for 8-bit operation, DB0 to DB3).

  The busy flag must be checked (one instruction) after the 4-bit data has been transferred twice. Two more 4-bit operations then transfer the busy flag and address counter data.
- For 8-bit interface data, all eight bus lines (DB0 to DB7) are used.

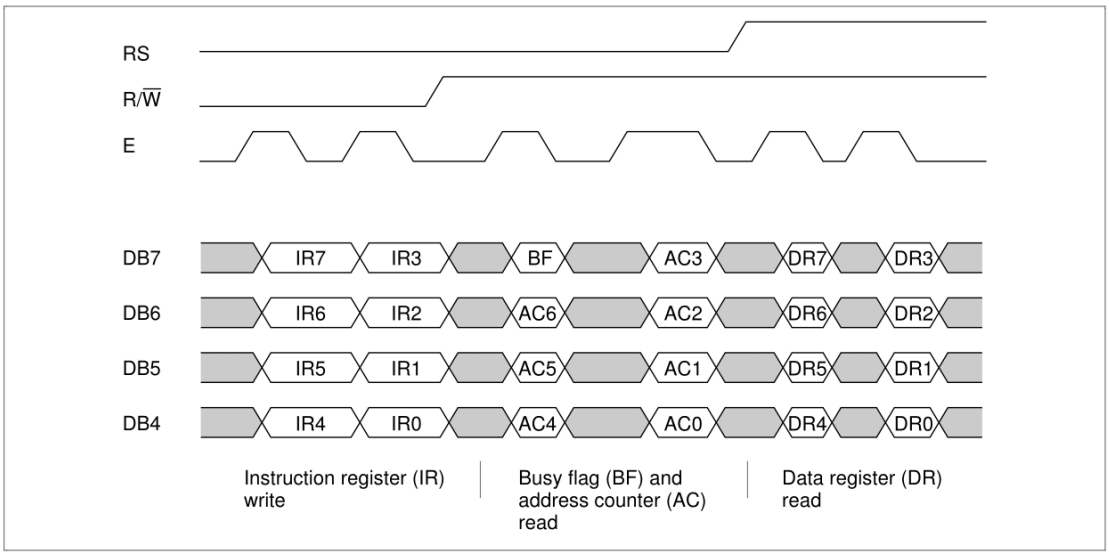


Figure 9 4-Bit Transfer Example

Reset Function

Initializing by Internal Reset Circuit

An internal reset circuit automatically initializes the HD44780U when the power is turned on. The following instructions are executed during the initialization. The busy flag (BF) is kept in the busy state until the initialization ends (BF = 1). The busy state lasts for 10 ms after V_{CC} rises to 4.5 V.

1. Display clear
2. Function set:
   DL = 1; 8-bit interface data
   N = 0; 1-line display
   F = 0; 5 × 8 dot character font
3. Display on/off control:
   D = 0; Display off
   C = 0; Cursor off
   B = 0; Blinking off
4. Entry mode set:
   I/D = 1; Increment by 1
   S = 0; No shift

Note: If the electrical characteristics conditions listed under the table Power Supply Conditions Using Internal Reset Circuit are not met, the internal reset circuit will not operate normally and will fail to initialize the HD44780U. For such a case, initial-ization must be performed by the MPU as explained in the section, Initializing by Instruction.

Instructions

Outline

Only the instruction register (IR) and the data register (DR) of the HD44780U can be controlled by the MPU. Before starting the internal operation of the HD44780U, control information is temporarily stored into these registers to allow interfacing with various MPUs, which operate at different speeds, or various peripheral control devices. The internal operation of the HD44780U is determined by signals sent from the MPU. These signals, which include register selection signal (RS), read/

write signal (R/$\overline{W}$), and the data bus (DB0 to DB7), make up the HD44780U instructions (Table 6). There are four categories of instructions that:

- Designate HD44780U functions, such as display format, data length, etc.
- Set internal RAM addresses
- Perform data transfer with internal RAM
- Perform miscellaneous functions

Normally, instructions that perform data transfer with internal RAM are used the most. However, auto-incrementation by 1 (or auto-decrementation by 1) of internal HD44780U RAM addresses after each data write can lighten the program load of the MPU. Since the display shift instruction (Table 11) can perform concurrently with display data write, the user can minimize system development time with maximum programming efficiency.

When an instruction is being executed for internal operation, no instruction other than the busy flag/address read instruction can be executed.

Because the busy flag is set to 1 while an instruction is being executed, check it to make sure it is 0 before sending another instruction from the MPU.

Note: Be sure the HD44780U is not in the busy state (BF = 0) before sending an instruction from the MPU to the HD44780U. If an instruction is sent without checking the busy flag, the time between the first instruction and next instruction will take much longer than the instruction time itself. Refer to Table 6 for the list of each instruc-tion execution time.

Table 6 Instructions

	Code											Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Instruction	**RS**	**R/$\overline{W}$**	**DB7**	**DB6**	**DB5**	**DB4**	**DB3**	**DB2**	**DB1**	**DB0**	**Description**	
Clear display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address 0 in address counter.	
Return home	0	0	0	0	0	0	0	0	1	—	Sets DDRAM address 0 in address counter. Also returns display from being shifted to original position. DDRAM contents remain unchanged.	1.52 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 µs
Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B).	37 µs
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	—	—	Moves cursor and shifts display without changing DDRAM contents.	37 µs
Function set	0	0	0	0	1	DL	N	F	—	—	Sets interface data length (DL), number of display lines (N), and character font (F).	37 µs
Set CGRAM address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM address. CGRAM data is sent and received after this setting.	37 µs
Set DDRAM address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM address. DDRAM data is sent and received after this setting.	37 µs
Read busy flag & address	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Reads busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0 µs

Table 6 Instructions (cont)

Instruction	Code RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (max) (when f_{cp} or f_{OSC} is 270 kHz)
Write data to CG or DDRAM	1	0	Write data								Writes data into DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
Read data from CG or DDRAM	1	1	Read data								Reads data from DDRAM or CGRAM.	37 µs t_{ADD} = 4 µs*
	I/D = 1: Increment I/D = 0: Decrement S = 1: Accompanies display shift S/C = 1: Display shift S/C = 0: Cursor move R/L = 1: Shift to the right R/L = 0: Shift to the left DL = 1: 8 bits, DL = 0: 4 bits N = 1: 2 lines, N = 0: 1 line F = 1: 5 × 10 dots, F = 0: 5 × 8 dots BF = 1: Internally operating BF = 0: Instructions acceptable										DDRAM: Display data RAM CGRAM: Character generator RAM ACG: CGRAM address ADD: DDRAM address (corresponds to cursor address) AC: Address counter used for both DD and CGRAM addresses	Execution time changes when frequency changes Example: When f_{cp} or f_{OSC} is 250 kHz, $37\ \mu s \times \frac{270}{250} = 40\ \mu s$

Note: — indicates no effect.

* After execution of the CGRAM/DDRAM data write or read instruction, the RAM address counter is incremented or decremented by 1. The RAM address counter is updated after the busy flag turns off. In Figure 10, t_{ADD} is the time elapsed after the busy flag turns off until the address counter is updated.

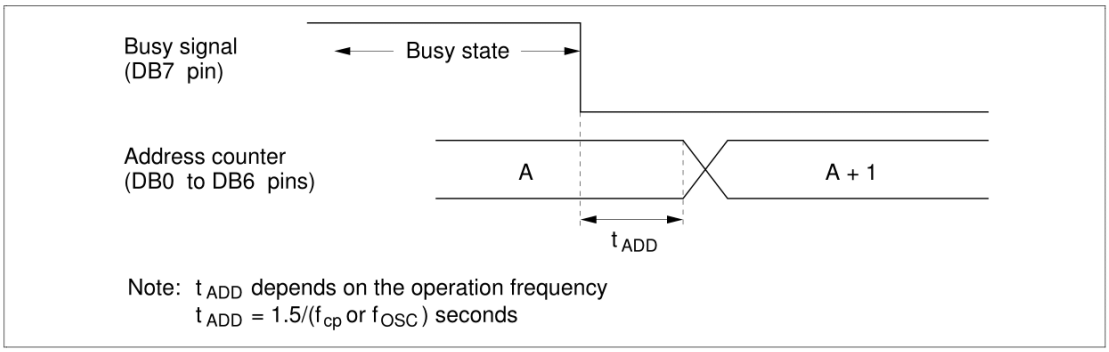


Figure 10 Address Counter Update

Instruction Description

Clear Display

Clear display writes space code 20H (character pattern for character code 20H must be a blank pattern) into all DDRAM addresses. It then sets DDRAM address 0 into the address counter, and returns the display to its original status if it was shifted. In other words, the display disappears and the cursor or blinking goes to the left edge of the display (in the first line if 2 lines are displayed). It also sets I/D to 1 (increment mode) in entry mode. S of entry mode does not change.

Return Home

Return home sets DDRAM address 0 into the address counter, and returns the display to its original status if it was shifted. The DDRAM contents do not change.

The cursor or blinking go to the left edge of the display (in the first line if 2 lines are displayed).

Entry Mode Set

I/D: Increments (I/D = 1) or decrements (I/D = 0) the DDRAM address by 1 when a character code is written into or read from DDRAM.

The cursor or blinking moves to the right when incremented by 1 and to the left when decremented by 1. The same applies to writing and reading of CGRAM.

S: Shifts the entire display either to the right (I/D = 0) or to the left (I/D = 1) when S is 1. The display does not shift if S is 0.

If S is 1, it will seem as if the cursor does not move but the display does. The display does not shift when reading from DDRAM. Also, writing into or reading out from CGRAM does not shift the display.

Display On/Off Control

D: The display is on when D is 1 and off when D is 0. When off, the display data remains in DDRAM, but can be displayed instantly by setting D to 1.

C: The cursor is displayed when C is 1 and not displayed when C is 0. Even if the cursor disappears, the function of I/D or other specifications will not change during display data write. The cursor is displayed using 5 dots in the 8th line for 5 × 8 dot character font selection and in the 11th line for the 5 × 10 dot character font selection (Figure 13).

B: The character indicated by the cursor blinks when B is 1 (Figure 13). The blinking is displayed as switching between all blank dots and displayed characters at a speed of 409.6-ms intervals when f_{cp} or f_{OSC} is 250 kHz. The cursor and blinking can be set to display simultaneously. (The blinking frequency changes according to f_{OSC} or the reciprocal of f_{cp}. For example, when f_{cp} is 270 kHz, 409.6 × 250/270 = 379.2 ms.)

Cursor or Display Shift

Cursor or display shift shifts the cursor position or display to the right or left without writing or reading display data (Table 7). This function is used to correct or search the display. In a 2-line display, the cursor moves to the second line when it passes the 40th digit of the first line. Note that the first and second line displays will shift at the same time.

When the displayed data is shifted repeatedly each line moves only horizontally. The second line display does not shift into the first line position.

The address counter (AC) contents will not change if the only action performed is a display shift.

Function Set

DL: Sets the interface data length. Data is sent or received in 8-bit lengths (DB7 to DB0) when DL is 1, and in 4-bit lengths (DB7 to DB4) when DL is 0.When 4-bit length is selected, data must be sent or received twice.

N: Sets the number of display lines.

F: Sets the character font.

Note: Perform the function at the head of the program before executing any instructions (except for the read busy flag and address instruction). From this point, the function set instruction cannot be executed unless the interface data length is changed.

Set CGRAM Address

Set CGRAM address sets the CGRAM address binary AAAAAA into the address counter.

Data is then written to or read from the MPU for CGRAM.

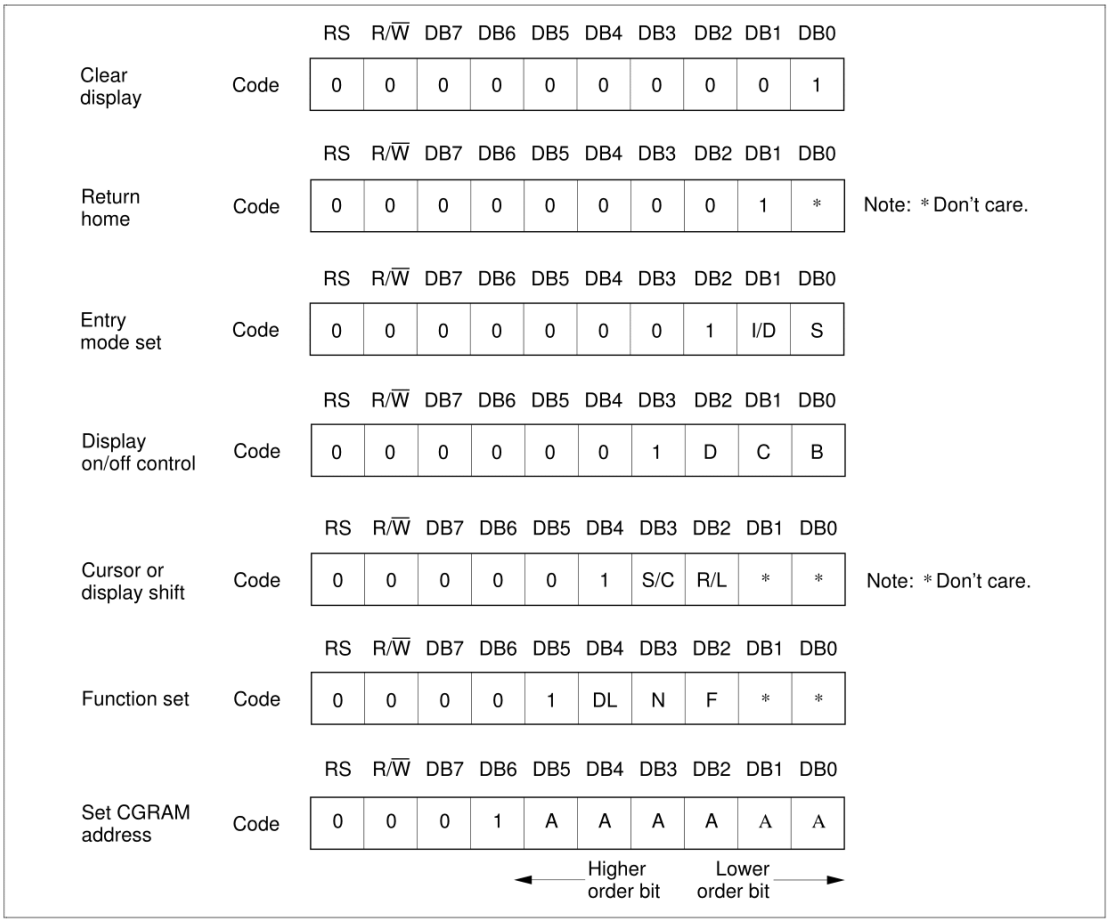


Figure 11 Instruction (1)

Set DDRAM Address

Set DDRAM address sets the DDRAM address binary AAAAAAA into the address counter.

Data is then written to or read from the MPU for DDRAM.

However, when N is 0 (1-line display), AAAAAAA can be 00H to 4FH. When N is 1 (2-line display), AAAAAAA can be 00H to 27H for the first line, and 40H to 67H for the second line.

Read Busy Flag and Address

Read busy flag and address reads the busy flag (BF) indicating that the system is now internally operating on a previously received instruction. If BF is 1, the internal operation is in progress. The next instruction will not be accepted until BF is reset to 0. Check the BF status before the next write operation. At the same time, the value of the address counter in binary AAAAAAA is read out. This address counter is used by both CG and DDRAM addresses, and its value is determined by the previous instruction. The address contents are the same as for instructions set CGRAM address and set DDRAM address.

Table 7 Shift Function

S/C	R/L	
0	0	Shifts the cursor position to the left. (AC is decremented by one.)
0	1	Shifts the cursor position to the right. (AC is incremented by one.)
1	0	Shifts the entire display to the left. The cursor follows the display shift.
1	1	Shifts the entire display to the right. The cursor follows the display shift.

Table 8 Function Set

N	F	No. of Display Lines	Character Font	Duty Factor	Remarks
0	0	1	5 × 8 dots	1/8	
0	1	1	5 × 10 dots	1/11	
1	*	2	5 × 8 dots	1/16	Cannot display two lines for 5 × 10 dot character font

Note: * Indicates don't care.

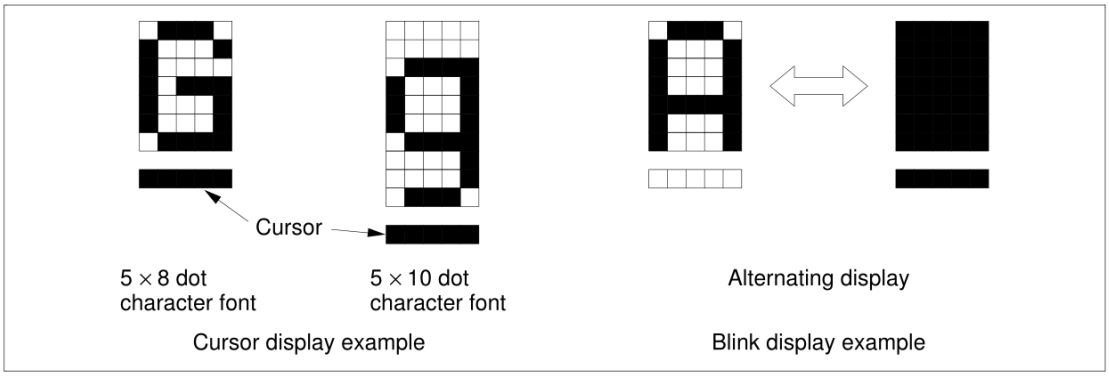


Figure 12 Cursor and Blinking

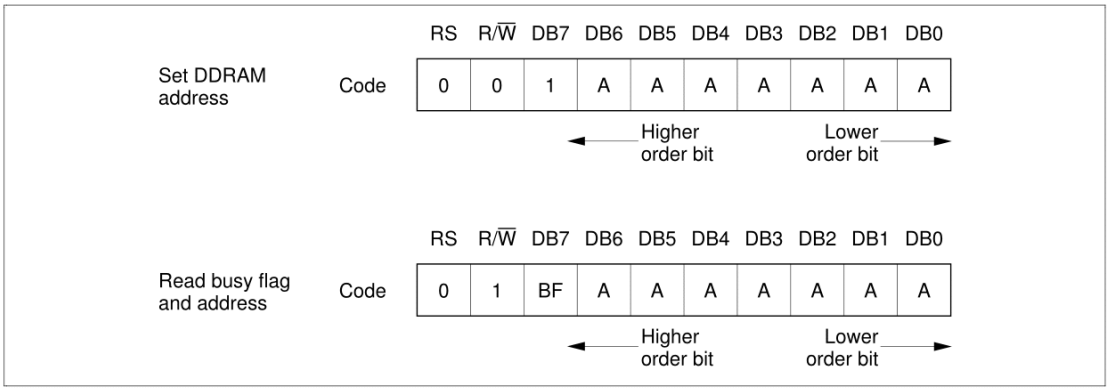


Figure 13 Instruction (2)

Write Data to CG or DDRAM

Write data to CG or DDRAM writes 8-bit binary data DDDDDDDD to CG or DDRAM.

To write into CG or DDRAM is determined by the previous specification of the CGRAM or DDRAM address setting. After a write, the address is automatically incremented or decremented by 1 according to the entry mode. The entry mode also determines the display shift.

Read Data from CG or DDRAM

Read data from CG or DDRAM reads 8-bit binary data DDDDDDDD from CG or DDRAM.

The previous designation determines whether CG or DDRAM is to be read. Before entering this read instruction, either CGRAM or DDRAM address set instruction must be executed. If not executed, the first read data will be invalid. When serially executing read instructions, the next address data is normally read from the second read. The address set instructions need not be executed just before this read instruction when shifting the cursor by the cursor shift instruction (when reading out DDRAM). The operation of the cursor shift instruction is the same as the set DDRAM address instruction.

After a read, the entry mode automatically increases or decreases the address by 1. However, display shift is not executed regardless of the entry mode.

Note: The address counter (AC) is automatically incremented or decremented by 1 after the write instructions to CGRAM or DDRAM are executed. The RAM data selected by the AC cannot be read out at this time even if read instructions are executed. Therefore, to correctly read data, execute either the address set instruction or cursor shift instruction (only with DDRAM), then just before reading the desired data, execute the read instruction from the second time the read instruction is sent.

Figure 14 Instruction (3)

Interfacing the HD44780U

Interface to MPUs

- Interfacing to an 8-bit MPU

  See Figure 16 for an example of using a I/O port (for a single-chip microcomputer) as an interface device.

  In this example, P30 to P37 are connected to the data bus DB0 to DB7, and P75 to P77 are connected to E, R/$\overline{W}$, and RS, respectively.

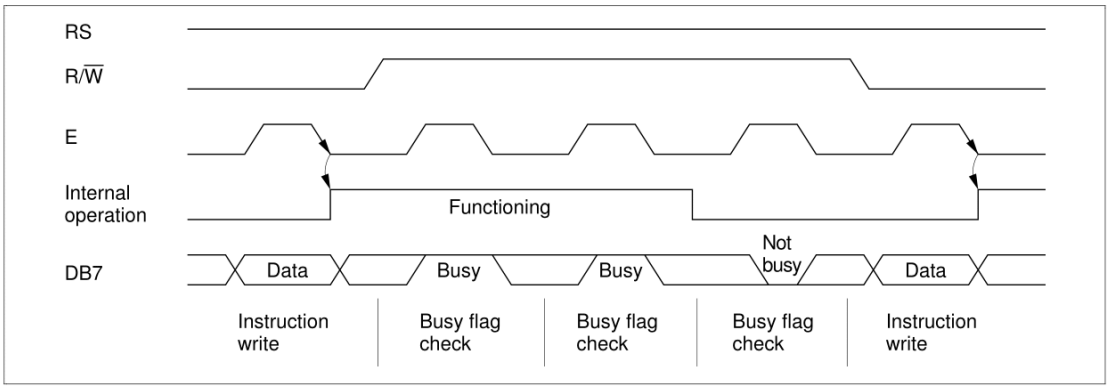


Figure 15 Example of Busy Flag Check Timing Sequence

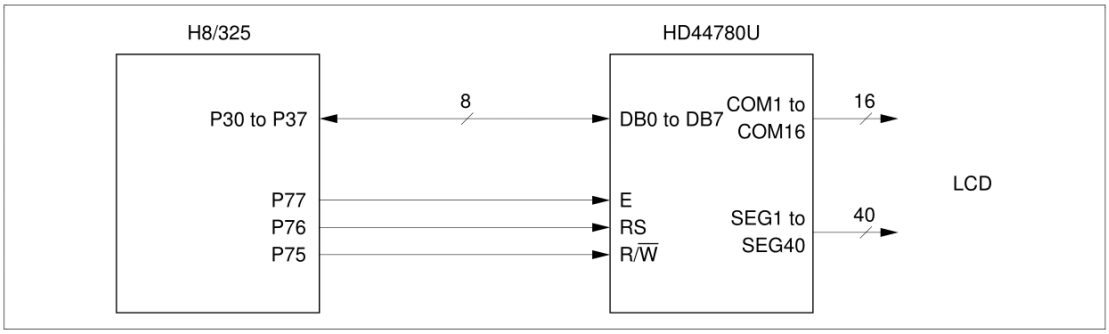


Figure 16 H8/325 Interface (Single-Chip Mode)

- Interfacing to a 4-bit MPU

  The HD44780U can be connected to the I/O port of a 4-bit MPU. If the I/O port has enough bits, 8-bit data can be transferred. Otherwise, one data transfer must be made in two operations for 4-bit data. In this case, the timing sequence becomes somewhat complex. (See Figure 17.)

  See Figure 18 for an interface example to the HMCS4019R.

  Note that two cycles are needed for the busy flag check as well as for the data transfer. The 4-bit operation is selected by the program.

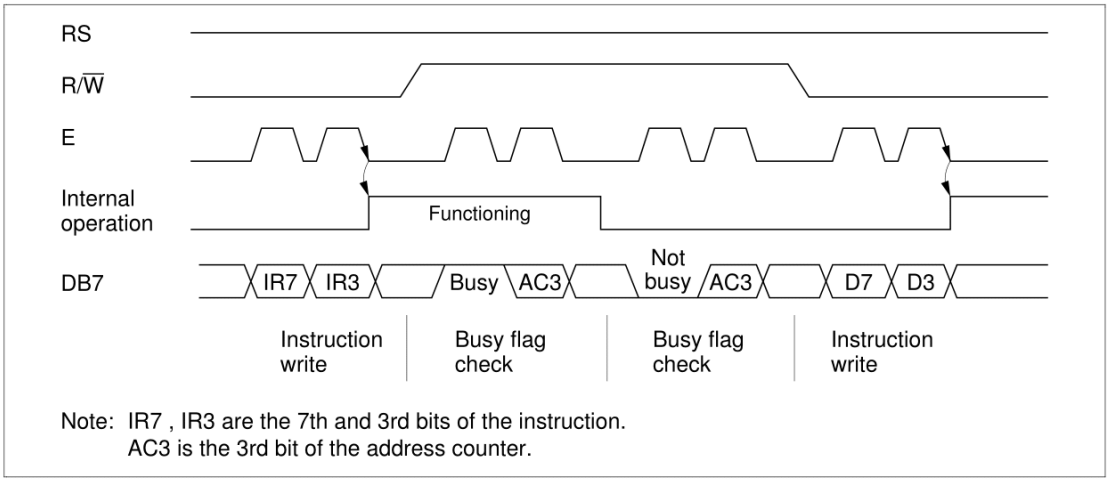


Figure 17 Example of 4-Bit Data Transfer Timing Sequence

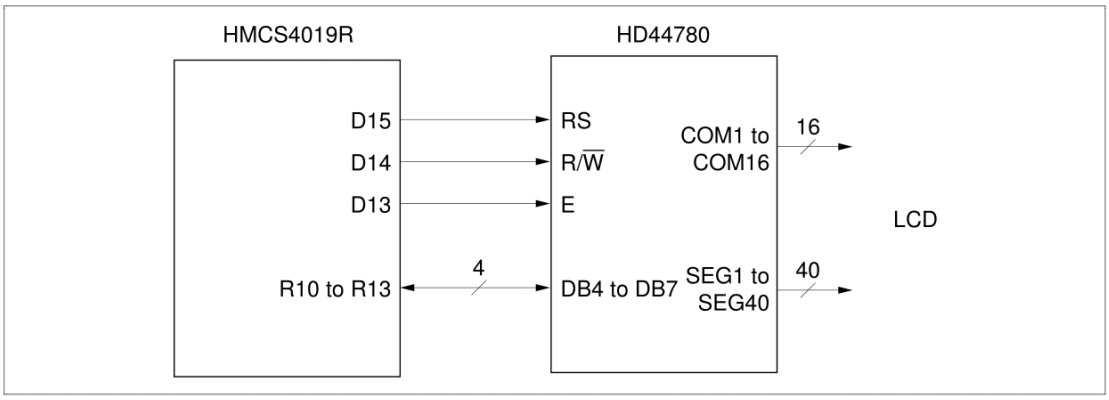


Figure 18 Example of Interface to HMCS4019R

Interface to Liquid Crystal Display

Character Font and Number of Lines: The HD44780U can perform two types of displays, 5 × 8 dot and 5 × 10 dot character fonts, each with a cursor.

Up to two lines are displayed for 5 × 8 dots and one line for 5 × 10 dots. Therefore, a total of three types of common signals are available (Table 9).

The number of lines and font types can be selected by the program. (See Table 6, Instructions.)

Connection to HD44780 and Liquid Crystal Display: See Figure 19 for the connection examples.

Table 9 Common Signals

Number of Lines	Character Font	Number of Common Signals	Duty Factor
1	5 × 8 dots + cursor	8	1/8
1	5 × 10 dots + cursor	11	1/11
2	5 × 8 dots + cursor	16	1/16

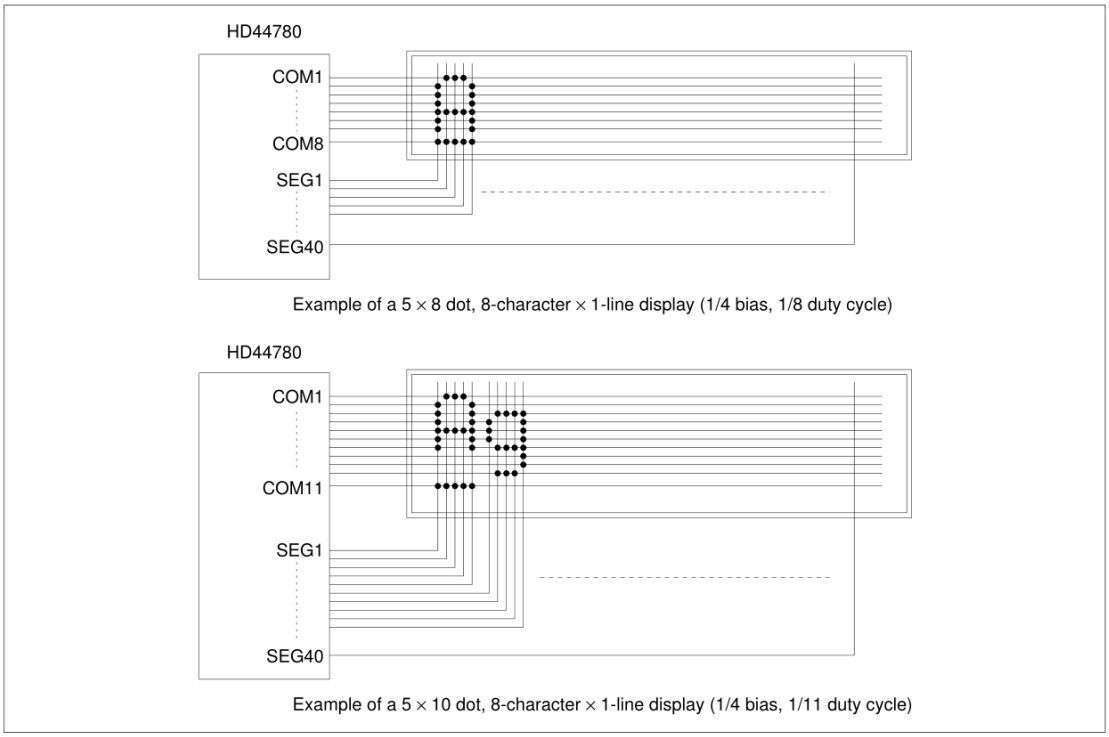


Figure 19 Liquid Crystal Display and HD44780 Connections

Since five segment signal lines can display one digit, one HD44780U can display up to 8 digits for a 1-line display and 16 digits for a 2-line display.

The examples in Figure 19 have unused common signal pins, which always output non-selection waveforms. When the liquid crystal display panel has unused extra scanning lines, connect the extra scanning lines to these common signal pins to avoid any undesirable effects due to crosstalk during the floating state.

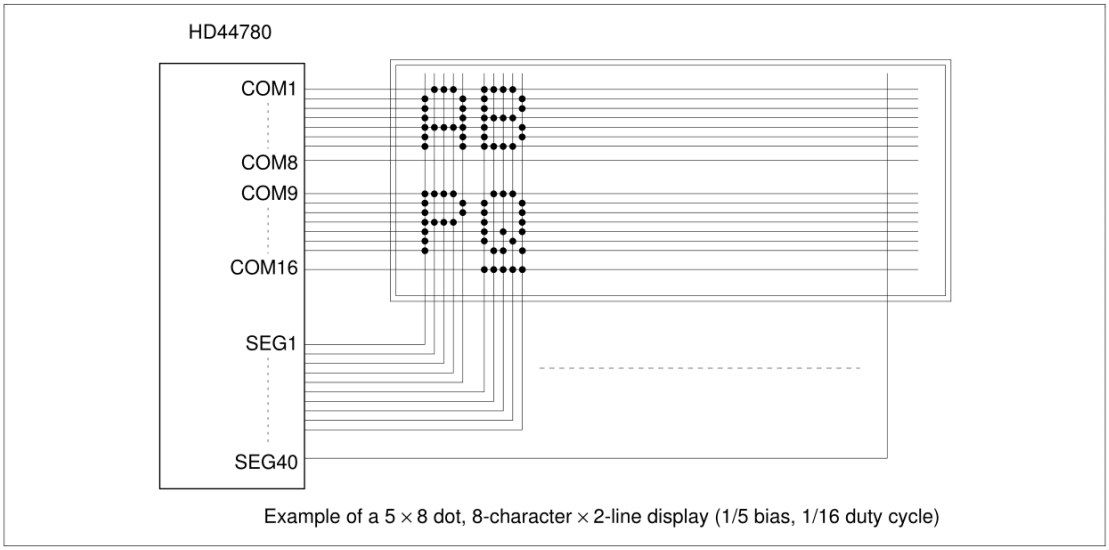


Figure 19 Liquid Crystal Display and HD44780 Connections (cont)

Connection of Changed Matrix Layout: In the preceding examples, the number of lines correspond to the scanning lines. However, the following display examples (Figure 20) are made possible by altering the matrix layout of the liquid crystal display panel. In either case, the only change is the layout. The display characteristics and the number of liquid crystal display characters depend on the number of common signals or on duty factor. Note that the display data RAM (DDRAM) addresses for 4 characters × 2 lines and for 16 characters × 1 line are the same as in Figure 19.

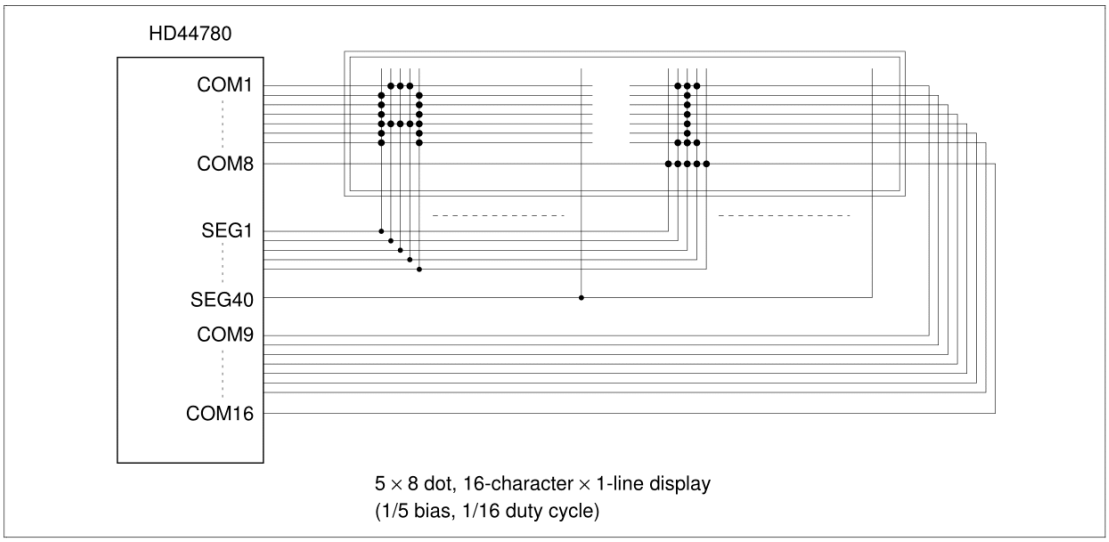


Figure 20 Changed Matrix Layout Displays

Power Supply for Liquid Crystal Display Drive

Various voltage levels must be applied to pins V1 to V5 of the HD44780U to obtain the liquid crystal display drive waveforms. The voltages must be changed according to the duty factor (Table 10).

VLCD is the peak value for the liquid crystal display drive waveforms, and resistance dividing provides voltages V1 to V5 (Figure 21).

Table 10 Duty Factor and Power Supply for Liquid Crystal Display Drive

	Duty Factor	
	1/8, 1/11	1/16
		Bias
Power Supply	1/4	1/5
V1	V_{CC}–1/4 VLCD	V_{CC}–1/5 VLCD
V2	V_{CC}–1/2 VLCD	V_{CC}–2/5 VLCD
V3	V_{CC}–1/2 VLCD	V_{CC}–3/5 VLCD
V4	V_{CC}–3/4 VLCD	V_{CC}–4/5 VLCD
V5	V_{CC}–VLCD	V_{CC}–VLCD

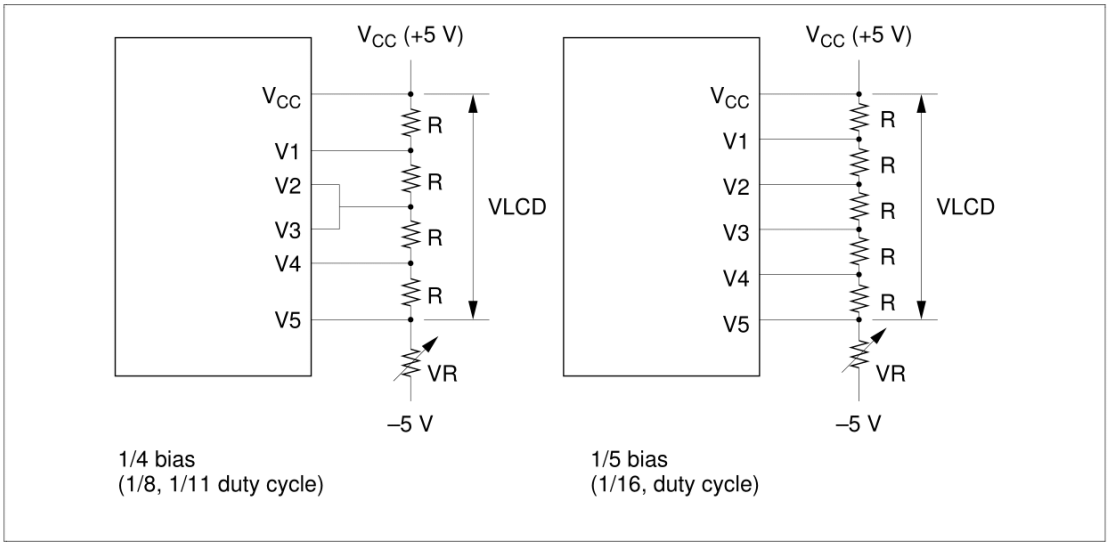


Figure 21 Drive Voltage Supply Example

Relationship between Oscillation Frequency and Liquid Crystal Display Frame Frequency

The liquid crystal display frame frequencies of Figure 22 apply only when the oscillation frequency is 270 kHz (one clock pulse of 3.7 µs).

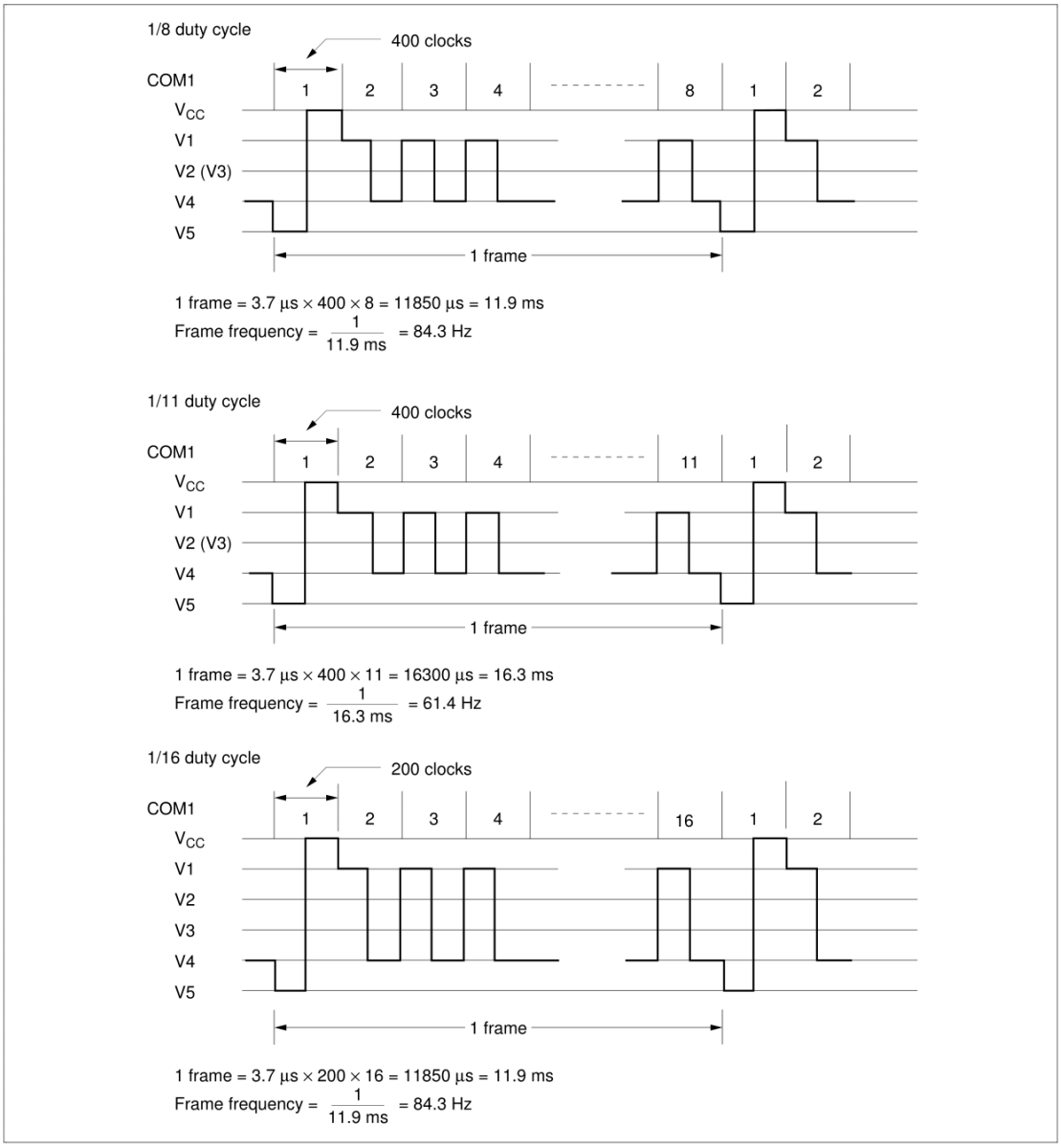


Figure 22 Frame Frequency

Instruction and Display Correspondence

- 8-bit operation, 8-digit × 1-line display with internal reset

  Refer to Table 11 for an example of an 8-digit × 1-line display in 8-bit operation. The HD44780U functions must be set by the function set instruction prior to the display. Since the display data RAM can store data for 80 characters, as explained before, the RAM can be used for displays such as for advertising when combined with the display shift operation.

  Since the display shift operation changes only the display position with DDRAM contents unchanged, the first display data entered into DDRAM can be output when the return home operation is performed.

- 4-bit operation, 8-digit × 1-line display with internal reset

  The program must set all functions prior to the 4-bit operation (Table 12). When the power is turned on, 8-bit operation is automatically selected and the first write is performed as an 8-bit operation. Since DB0 to DB3 are not connected, a rewrite is then required. However, since one operation is completed in two accesses for 4-bit operation, a rewrite is needed to set the functions (see Table 12). Thus, DB4 to DB7 of the function set instruction is written twice.

- 8-bit operation, 8-digit × 2-line display

  For a 2-line display, the cursor automatically moves from the first to the second line after the 40th digit of the first line has been written. Thus, if there are only 8 characters in the first line, the DDRAM address must be again set after the 8th character is completed. (See Table 13.) Note that the display shift operation is performed for the first and second lines. In the example of Table 13, the display shift is performed when the cursor is on the second line. However, if the shift operation is performed when the cursor is on the first line, both the first and second lines move together. If the shift is repeated, the display of the second line will not move to the first line. The same display will only shift within its own line for the number of times the shift is repeated.

Note: When using the internal reset, the electrical characteristics in the Power Supply Conditions Using Internal Reset Circuit table must be satisfied. If not, the HD44780U must be initialized by instructions. See the section, Initializing by Instruction.

Table 11 8-Bit Operation, 8-Digit × 1-Line Display Example with Internal Reset

Step No.	Instruction RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Display	Operation
1	Power supply on (the HD44780U is initialized by the internal reset circuit)											Initialized. No display.
2	Function set	0	0	0	0	1	1	0	0	*	*	Sets to 8-bit operation and selects 1-line display and 5 × 8 dot character font. (Number of display lines and character fonts cannot be changed after step #2.)
3	Display on/off control	0	0	0	0	0	0	1	1	1	0	Turns on display and cursor. Entire display is in space mode because of initialization.
4	Entry mode set	0	0	0	0	0	0	0	1	1	0	Sets mode to increment the address by one and to shift the cursor to the right at the time of write to the DD/CGRAM. Display is not shifted.
5	Write data to CGRAM/DDRAM	1	0	0	1	0	0	1	0	0	0	Writes H. DDRAM has already been selected by initialization when the power was turned on. The cursor is incremented by one and shifted to the right.
6	Write data to CGRAM/DDRAM	1	0	0	1	0	0	1	0	0	1	Writes I.
7	. . . . .					. . . . .					. . . . .	
8	Write data to CGRAM/DDRAM	1	0	0	1	0	0	1	0	0	1	Writes I.
9	Entry mode set	0	0	0	0	0	0	0	1	1	1	Sets mode to shift display at the time of write.
10	Write data to CGRAM/DDRAM	1	0	0	0	1	0	0	0	0	0	Writes a space.

Table 11 8-Bit Operation, 8-Digit × 1-Line Display Example with Internal Reset (cont)

Step No.	Instruction RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Display	Operation
11	Write data to CGRAM/DDRAM										TACHI M_	Writes M.
	1	0	0	1	0	0	1	1	0	1		
12						.					.	
						.					.	
						.					.	
						.					.	
						.					.	
13	Write data to CGRAM/DDRAM										MICROKO_	Writes O.
	1	0	0	1	0	0	1	1	1	1		
14	Cursor or display shift										MICROKO	Shifts only the cursor position to the left.
	0	0	0	0	0	1	0	0	*	*		
15	Cursor or display shift										MICROKO	Shifts only the cursor position to the left.
	0	0	0	0	0	1	0	0	*	*		
16	Write data to CGRAM/DDRAM										ICROCO	Writes C over K. The display moves to the left.
	1	0	0	1	0	0	0	0	1	1		
17	Cursor or display shift										MICROCO	Shifts the display and cursor position to the right.
	0	0	0	0	0	1	1	1	*	*		
18	Cursor or display shift										MICROCO_	Shifts the display and cursor position to the right.
	0	0	0	0	0	1	0	1	*	*		
19	Write data to CGRAM/DDRAM										ICROCOM_	Writes M.
	1	0	0	1	0	0	1	1	0	1		
20						.					.	
						.					.	
						.					.	
						.					.	
						.					.	
21	Return home										HITACHI	Returns both display and cursor to the original position (address 0).
	0	0	0	0	0	0	0	0	1	0		

Table 13 8-Bit Operation, 8-Digit × 2-Line Display Example with Internal Reset

Step No.	Instruction RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Display	Operation
1	Power supply on (the HD44780U is initialized by the internal reset circuit)											Initialized. No display.
2	Function set											Sets to 8-bit operation and selects 2-line display and 5 × 8 dot character font.
	0	0	0	0	1	1	1	0	*	*		
3	Display on/off control											Turns on display and cursor. All display is in space mode because of initialization.
	0	0	0	0	0	0	1	1	1	0	_	
4	Entry mode set											Sets mode to increment the address by one and to shift the cursor to the right at the time of write to the DD/CGRAM. Display is not shifted.
	0	0	0	0	0	0	0	1	1	0	_	
5	Write data to CGRAM/DDRAM											Writes H. DDRAM has already been selected by initialization when the power was turned on. The cursor is incremented by one and shifted to the right.
	1	0	0	1	0	0	1	0	0	0	H_	
6	.				.						.	
	.				.						.	
	.				.						.	
	.				.						.	
	.				.						.	
7	Write data to CGRAM/DDRAM											Writes I.
	1	0	0	1	0	0	1	0	0	1	HITACHI_	
8	Set DDRAM address											Sets DDRAM address so that the cursor is positioned at the head of the second line.
	0	0	1	1	0	0	0	0	0	0	HITACHI _	

Table 13 8-Bit Operation, 8-Digit × 2-Line Display Example with Internal Reset (cont)

Step No.	Instruction RS	R/$\overline{W}$	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Display	Operation
9	Write data to CGRAM/DDRAM 1	0	0	1	0	0	1	1	0	1	HITACHI M_	Writes M.
10	.	.	.	.	.	.	.	.	.	.	.	
11	Write data to CGRAM/DDRAM 1	0	0	1	0	0	1	1	1	1	HITACHI MICROCO_	Writes O.
12	Entry mode set 0	0	0	0	0	0	0	1	1	1	HITACHI MICROCO_	Sets mode to shift display at the time of write.
13	Write data to CGRAM/DDRAM 1	0	0	1	0	0	1	1	0	1	ITACHI ICROCOM_	Writes M. Display is shifted to the left. The first and second lines both shift at the same time.
14	.	.	.	.	.	.	.	.	.	.	.	
15	Return home 0	0	0	0	0	0	0	0	1	0	HITACHI MICROCOM	Returns both display and cursor to the original position (address 0).

Initializing by Instruction

If the power supply conditions for correctly operating the internal reset circuit are not met, initialization by instructions becomes necessary.

Refer to Figures 23 and 24 for the procedures on 8-bit and 4-bit initializations, respectively.

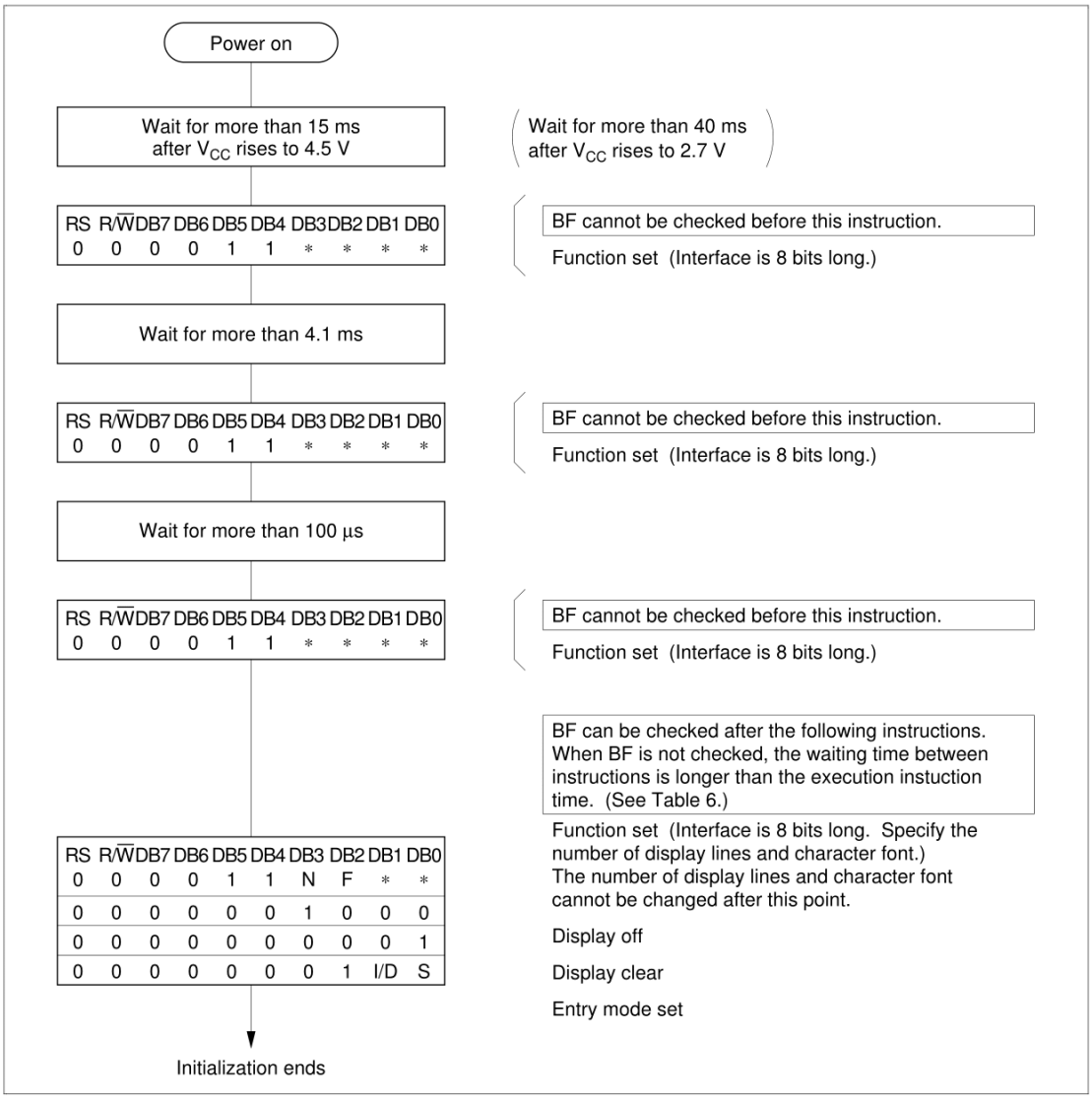


Figure 23 8-Bit Interface

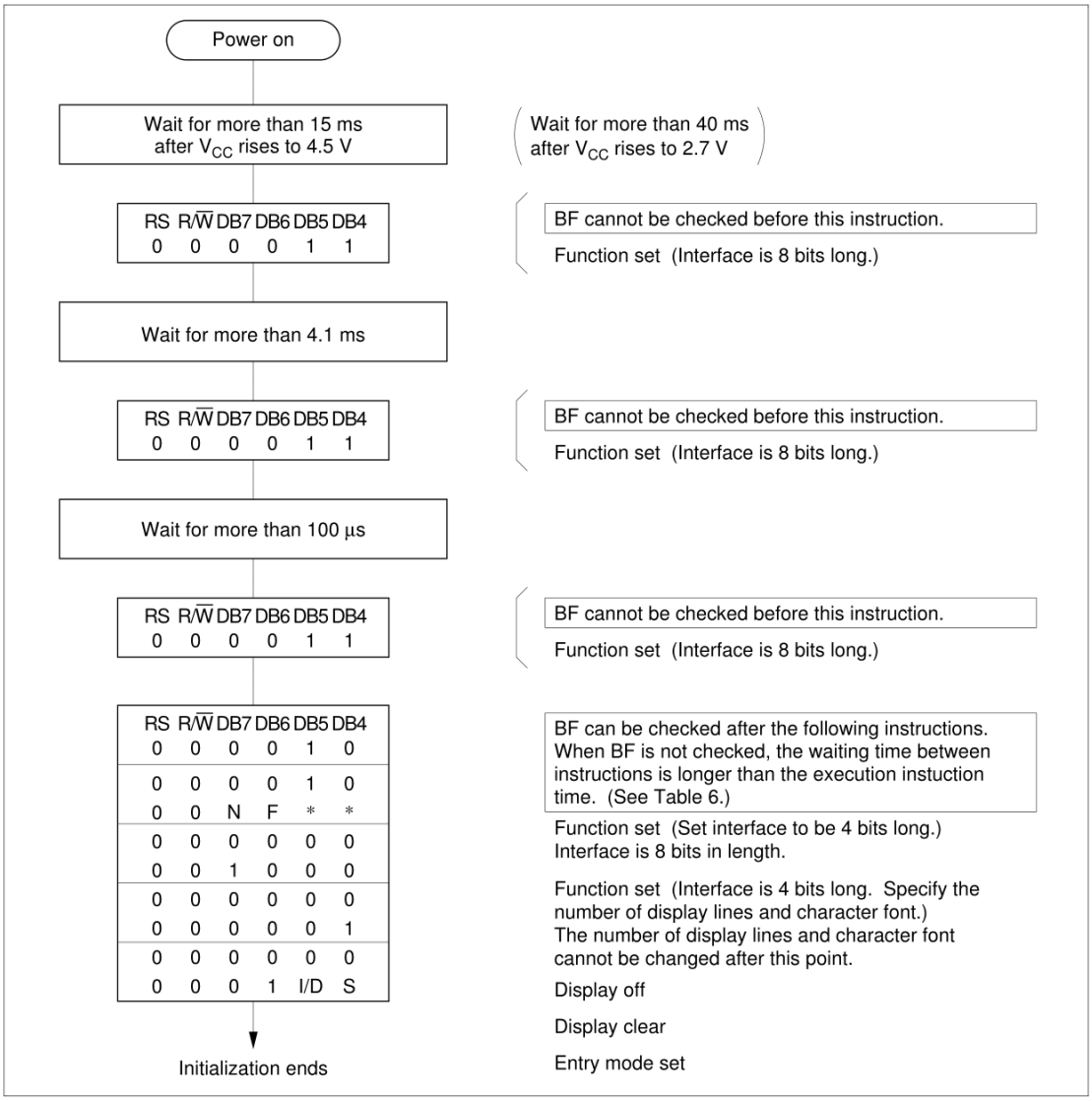


Figure 24 4-Bit Interface

Absolute Maximum Ratings*

Item	Symbol	Value	Unit	Notes
Power supply voltage (1)	V_{CC}–GND	–0.3 to +7.0	V	1
Power supply voltage (2)	V_{CC}–V5	–0.3 to +13.0	V	1, 2
Input voltage	Vt	–0.3 to V_{CC} +0.3	V	1
Operating temperature	T_{opr}	–30 to +75	°C	
Storage temperature	T_{stg}	–55 to +125	°C	4

Note: * If the LSI is used above these absolute maximum ratings, it may become permanently damaged. Using the LSI within the following electrical characteristic limits is strongly recommended for normal operation. If these electrical characteristic conditions are also exceeded, the LSI will malfunction and cause poor reliability.

DC Characteristics (V_{CC} = 2.7 to 4.5 V, T_a = –30 to +75°C*[3])

Item	Symbol	Min	Typ	Max	Unit	Test Condition	Notes*
Input high voltage (1) (except OSC1)	VIH1	$0.7V_{CC}$	—	V_{CC}	V		6
Input low voltage (1) (except OSC1)	VIL1	–0.3	—	0.55	V		6
Input high voltage (2) (OSC1)	VIH2	$0.7V_{CC}$	—	V_{CC}	V		15
Input low voltage (2) (OSC1)	VIL2	—	—	$0.2V_{CC}$	V		15
Output high voltage (1) (DB0–DB7)	VOH1	$0.75V_{CC}$	—	—	V	$–I_{OH}$ = 0.1 mA	7
Output low voltage (1) (DB0–DB7)	VOL1	—	—	$0.2V_{CC}$	V	I_{OL} = 0.1 mA	7
Output high voltage (2) (except DB0–DB7)	VOH2	$0.8V_{CC}$	—	—	V	$–I_{OH}$ = 0.04 mA	8
Output low voltage (2) (except DB0–DB7)	VOL2	—	—	$0.2V_{CC}$	V	I_{OL} = 0.04 mA	8
Driver on resistance (COM)	R_{COM}	—	2	20	kΩ	±Id = 0.05 mA, VLCD = 4 V	13
Driver on resistance (SEG)	R_{SEG}	—	2	30	kΩ	±Id = 0.05 mA, VLCD = 4 V	13
Input leakage current	I_{LI}	–1	—	1	µA	VIN = 0 to V_{CC}	9
Pull-up MOS current (DB0–DB7, RS, R/$\overline{W}$)	$–I_p$	10	50	120	µA	V_{CC} = 3 V	
Power supply current	I_{CC}	—	150	300	µA	R_f oscillation, external clock V_{CC} = 3 V, f_{OSC} = 270 kHz	10, 14
LCD voltage	VLCD1	3.0	—	11.0	V	V_{CC}–V5, 1/5 bias	16
	VLCD2	3.0	—	11.0	V	V_{CC}–V5, 1/4 bias	16

Note: * Refer to the Electrical Characteristics Notes section following these tables.

AC Characteristics (V_{CC} = 2.7 to 4.5 V, T_a = –30 to +75°C*[3])

Clock Characteristics

Item		Symbol	Min	Typ	Max	Unit	Test Condition	Note*
External clock operation	External clock frequency	f_{cp}	125	250	350	kHz		11
	External clock duty	Duty	45	50	55	%		
	External clock rise time	t_{rcp}	—	—	0.2	µs		
	External clock fall time	t_{fcp}	—	—	0.2	µs		
R_f oscillation	Clock oscillation frequency	f_{OSC}	190	270	350	kHz	R_f = 75 kΩ, V_{CC} = 3 V	12

Note: * Refer to the Electrical Characteristics Notes section following these tables.

Bus Timing Characteristics

Write Operation

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Enable cycle time	t_{cycE}	1000	—	—	ns	Figure 25
Enable pulse width (high level)	PW_{EH}	450	—	—		
Enable rise/fall time	t_{Er}, t_{Ef}	—	—	25		
Address set-up time (RS, R/$\overline{W}$ to E)	t_{AS}	60	—	—		
Address hold time	t_{AH}	20	—	—		
Data set-up time	t_{DSW}	195	—	—		
Data hold time	t_H	10	—	—		

Read Operation

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Enable cycle time	t_{cycE}	1000	—	—	ns	Figure 26
Enable pulse width (high level)	PW_{EH}	450	—	—		
Enable rise/fall time	t_{Er}, t_{Ef}	—	—	25		
Address set-up time (RS, R/$\overline{W}$ to E)	t_{AS}	60	—	—		
Address hold time	t_{AH}	20	—	—		
Data delay time	t_{DDR}	—	—	360		
Data hold time	t_{DHR}	5	—	—		

Interface Timing Characteristics with External Driver

Item		Symbol	Min	Typ	Max	Unit	Test Condition
Clock pulse width	High level	t_{CWH}	800	—	—	ns	Figure 27
	Low level	t_{CWL}	800	—	—		
Clock set-up time		t_{CSU}	500	—	—		
Data set-up time		t_{SU}	300	—	—		
Data hold time		t_{DH}	300	—	—		
M delay time		t_{DM}	–1000	—	1000		
Clock rise/fall time		t_{ct}	—	—	200		

Power Supply Conditions Using Internal Reset Circuit

Item	Symbol	Min	Typ	Max	Unit	Test Condition
Power supply rise time	t_{rCC}	0.1	—	10	ms	Figure 28
Power supply off time	t_{OFF}	1	—	—		

DC Characteristics (V_{CC} = 4.5 to 5.5 V, T_a = –30 to +75°C*[3])

Item	Symbol	Min	Typ	Max	Unit	Test Condition	Notes*
Input high voltage (1) (except OSC1)	VIH1	2.2	—	V_{CC}	V		6
Input low voltage (1) (except OSC1)	VIL1	–0.3	—	0.6	V		6
Input high voltage (2) (OSC1)	VIH2	V_{CC}–1.0	—	V_{CC}	V		15
Input low voltage (2) (OSC1)	VIL2	—	—	1.0	V		15
Output high voltage (1) (DB0–DB7)	VOH1	2.4	—	—	V	$–I_{OH}$ = 0.205 mA	7
Output low voltage (1) (DB0–DB7)	VOL1	—	—	0.4	V	I_{OL} = 1.2 mA	7
Output high voltage (2) (except DB0–DB7)	VOH2	0.9 V_{CC}	—	—	V	$–I_{OH}$ = 0.04 mA	8
Output low voltage (2) (except DB0–DB7)	VOL2	—	—	0.1 V_{CC}	V	I_{OL} = 0.04 mA	8
Driver on resistance (COM)	RCOM	—	2	20	kΩ	±Id = 0.05 mA, VLCD = 4 V	13
Driver on resistance (SEG)	RSEG	—	2	30	kΩ	±Id = 0.05 mA, VLCD = 4 V	13
Input leakage current	I_{LI}	–1	—	1	µA	VIN = 0 to V_{CC}	9
Pull-up MOS current (DB0–DB7, RS, R/$\overline{W}$)	$–I_p$	50	125	250	µA	V_{CC} = 5 V	
Power supply current	I_{CC}	—	350	600	µA	R_f oscillation, external clock V_{CC} = 5 V, f_{OSC} = 270 kHz	10, 14
LCD voltage	VLCD1	3.0	—	11.0	V	V_{CC}–V5, 1/5 bias	16
	VLCD2	3.0	—	11.0	V	V_{CC}–V5, 1/4 bias	16

Note: * Refer to the Electrical Characteristics Notes section following these tables.

2) RFID 20-LA

ID-innovations
Advanced RFID design

ID-innovations
Advanced RFID design

ID-2LA, ID-12LA, ID-20LA

Low Voltage Series

Reader Modules

Datasheet Version1.0 Date 09/01/13

Content

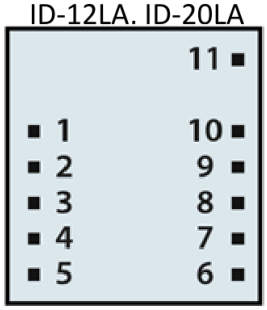
1. Overview 3
2. Pin Out for ID12-LA and ID20-LA 3
ID-12LA, ID-20LA 3
ID-2LA 3
3. Device Operational Characteristics 4
4. Data Formats 4
5. Magnetic Emulation Waveforms 5
6. Pin Description & Output Data Formats 6
7. Absolute Maximum Ratings 7
8. Circuit Diagram 8
8.1 Circuit Diagram for ID-12LA, ID-20LA 8
8.2 Circuit Diagram for ID-2LA 8
9. Dimensions (mm) 9
10. Connection direct to a computer 10
10.1 Connection to a Processor UART 11
10.2 Connecting a Read LED 11
11. Useful information 11
12. Choice of Power Supply 12
13. Designing Coils for ID-2LA 13
14. Fine Tuning the ID-2LA 14
15. ID-2LA Compatibility Issues with the ID-2 15
16. ID-12LA, ID-20LA Compatibility Issues with ID-12, ID-20 15
17. Layout Precautions for ID-12LA and ID-20LA 15
18. Contact Information 16
Disclaimer 17

1. Overview

ID2-LA, ID12-LA and the ID20-LA series are small footprint 2.8-5.0volt reader modules that support ASCII, Wiegand26 and Magnetic ABA Track2 data formats. The modules are pin and function compatible with the ID2/12/20 series.

2. Pin Out for ID12-LA and ID20-LA

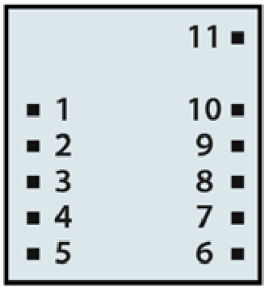
ID-12LA. ID-20LA



Bottom View

1. GND
2. RES (Reset Bar)
3. NC
4. NC
5. CP
6. Tag in Range
7. Format Selector
8. D1 (Data Pin 1)
9. D0 (Data Pin 0)
10. Read (LED / Beeper)
11. +2.8V thru +5.0V

ID-2LA



Bottom View

1. GND
2. RES (Reset Bar)
3. ANT (Antenna)
4. ANT (Antenna)
5. CP
6. Tag in Range
7. Format Selector
8. D1 (Data Pin 1)
9. D0 (Data Pin 0)
10. Read (LED / Beeper)
11. +2.8V thru +5.0V

3. Device Operational Characteristics

Parameter	ID-2LA, ID-12LA, ID-20LA
Frequency	125 kHz nominal
Card Format	EM 4001 or compatible
Read Range ID3	Up to 30 using suitable antenna using ID-Innovations clamshell card @5v
Read Range ID13	Up to 12cm using ISO card, up to 18cm using ID-Innovations clamshell card @5v
Read Range ID23	Up to 18cm using ISO card, up to 25cm using ID-Innovations clamshell card @5v
Encoding	Manchester 64-bit, modulus 64
Power Requirement	+2.8 VDC thru +5 VDC @ 35mA ID-12LA, 45mA ID-20LA
RF I/O Output Current	+/- 200mA PKPK

4. Data Formats

Output Data Structure - ASCII - 9600 Baud, No Parity, 1 stop bit.

Output = CMOS (Push Pull) 0-Vdd

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)

Example for Calculation of Checksum for ASCII

Suppose the output Data is 0C000621A58E

Here the actual data is 0C,00,00,06,21,A5 and the checksum is 6E

Using binary we Exclusive OR the bit columns

0C	=	00001100
00	=	00000000
06	=	00000110
21	=	00100001
A5	=	10100101
CHECKSUM		10001110 (8E)

Output Data Structure - Wiegand26 – 1mS repeat, 50uS pulse. Open Drain

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
P	E	E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	P
Even parity (E)													Odd parity (O)												

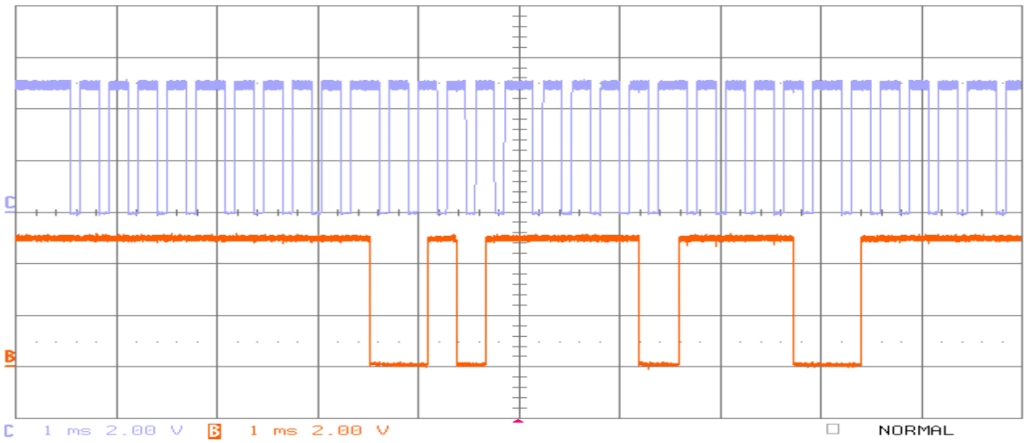
P = Parity start bit and stop bit

Output Data Magnetic ABA Track2 – At Approx. 80cm/sec. Open Drain

10 Leading Zeros	SS	Data	ES	LCR	10 Ending Zeros

[SS is the Start Character of 11010, ES is the end character of 11111, and LRC is the Longitudinal Redundancy Check.]

5. Magnetic Emulation Waveforms



Blue = Clock, Brown = Data

Fig. 3

Start and End Sequences for Magnetic Timing

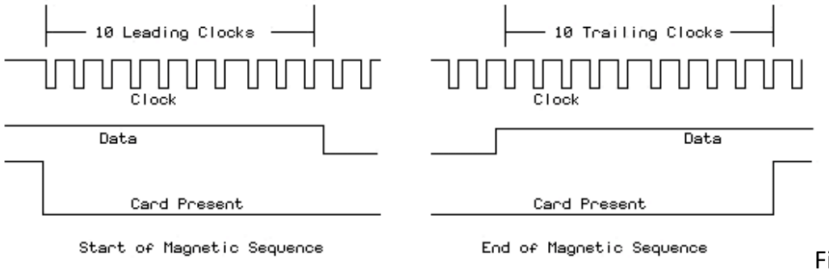


Fig. 4

Data Timings for Magnetic Emulation

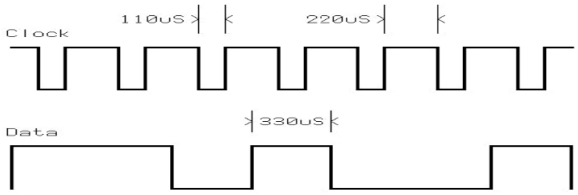


Fig. 5

The magnetic Emulation Sequence starts with the Card Present Line going active (down). There next follows 10 clocks with Zero '0' data. At the end of the 10 leading clocks the start character (11010) is sent and this is followed by the data. At the end of the data the end character is sent followed by the LCR. Finally 10 trailing clocks are sent and the card present line is raised.
The data bit duration is approximately 330uS. The approximate clock duration is 110uS. Because of the symmetry data can be clocked off either the rising or falling edge of the clock.

6. Pin Description & Output Data Formats

Pin #	Description	ASCII	Magnet Emulation	Wiegand26
Pin 1	Zero Volts	GND 0V	GND 0V	GND 0V
Pin 2	Strap to Pin11	Reset Bar	Reset Bar	Reset Bar
Pin 3	To External Antenna ID-2LA only	Antenna	Antenna	Antenna
Pin 4	To External Antenna ID-2LA only	Antenna	Antenna	Antenna
Pin 5	Card Present	No function	Card Present*	No function
Pin 6	Tag in Range (Future)	Tag in Range	Tag in Range	Tag in Range
Pin 7	Format Selector (+/-)	Strap to GND	Strap to Pin 10	Strap to +5V
Pin 8	Data 1	CMOS	Clock*	One Output*
Pin 9	Data 0	TTL Data (inverted)	Data*	Zero Output*
Pin 10	3.1 kHz Logic	Beeper / LED	Beeper / LED	Beeper / LED
Pin 11	DC Voltage Supply	+2.8 thru 5V	+2.8V thru 5V	+2.8V thru 5V

* Requires 4K7 Pull-up resistor to +5V

Pin1 is the zero volts supply pin and the communications common ground.

Pin2 is used in manufacture and should be strapped to pin11 at all times.

Pin3 is only connected on the ID-2LA and it should be connected to an external antenna of 1.337mH. On the ID-12LA and ID-20LA modules pin3 should be left blank.

Pin4 is only connected in the ID-2LA. It should be connected to an external antenna. It should be left blank on the ID12LA and ID20LA modules.

Pin5 is only used as a 'Card Present' output when the output format is set to Magnetic Emulation. For timings see the 'Magnetic Emulation Waveforms in section 5.

Pin6 is used as a 'Tag in Range' indicator. When a tag is in range the pin is set to VDD voltage else it is at 0v. Pin6 output has an internal 3K3 resistor and may be used to drive an LED directly.

Pin7 is the format selector. The format selector selects the format depending where it is connected. See table above. Note that the output format is decided at switch on and cannot be changed later.

Pin8 has alternate functions. When the output format is set to Magnetic Emulation pin8 is used as the 'Clock' output. For timings see the 'Magnetic Emulation Waveforms in section 5. The alternate mode is active when the output format is set to ASCII and pin8 then outputs the ASCII data. Pin 8 may be used to connect to a computer RS232 input. See section on connecting to a computer.

Pin9 is active when the output format is set to ASCII and it outputs complementary (inverted) ASCII data output. Pin 9 is also suitable for connection to a UART.

Pin10 is the beeper out pin. When the beeper is active pin10 delivers an output square wave of approximately 3.3KHz. The pin is not buffered and cannot be used to drive a beeper directly.

Pin11 is the DC supply pin. The supply voltage must be free from noise and preferably from a linear regulator with less than 3mV PKPK noise. Many modern regulators have noise below 100uV RMS and these are ideal. See the section on choice of power supply.

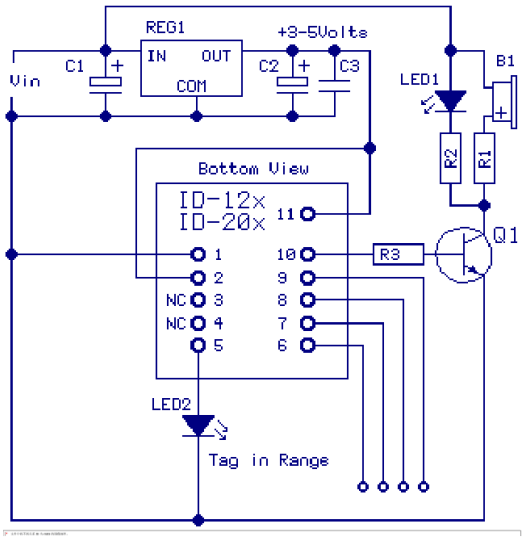
7. Absolute Maximum Ratings

Maximum voltage applied to Pin 2	(Vcc)	5.5volt
Maximum voltage applied to Pin 2	(Reset)	Vcc + 0.7v, -0.7v
Maximum current drawn from Pin 3 (	Antenna)	+/- 75mA
Maximum 125 KHz RF Voltage at Pin 4	(Antenna)	+/- 80volt Peak
Maximum current drawn from Pin 5	(Card Present)	+/- 5mA
Maximum current drawn from Pin 6	(Tag in Range)	+/- 5mA
Maximum Voltage at Pin 7	(Format Selector)	Vcc + 0.7v, -0.7v
Maximum current drawn from Pin 8	(Data1)	+/- 5mA
Maximum current drawn from Pin 9	(Data0)	+/- 5mA
Maximum current drawn from Pin 10	(Beeper)	+/- 10mA
Additionally, Pins 5, 6, 7, 8, 9 & 10 may not have a voltage exceeding		Vcc + 0.7v, -0.7v

These ratings are absolute maximums and operation at or near the maximums may cause stress and eventual damage or unpredictable behaviour.

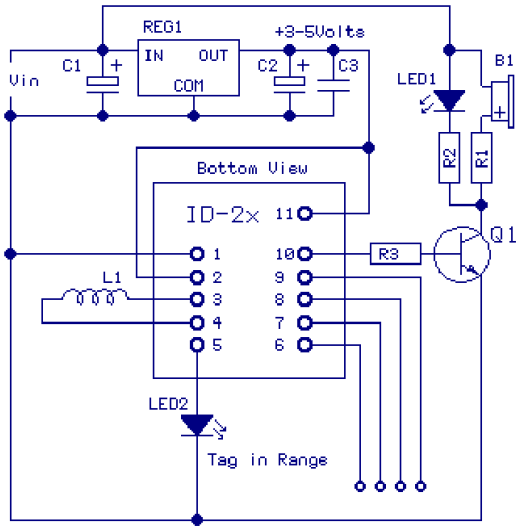
8. Circuit Diagram

8.1 Circuit Diagram for ID-12LA, ID-20LA



Parts List	
Part #	**Value**
R1	100R
R2	4K7
R3	2K2
C1	10uF 25v electrolytic
C2	1000uF 10v electrolytic
C3	100nF
Q1	BC457 or similar
LED1	Read LED
LED2	Tag In Range LED
B1	2.7khz – 3kHz 5v PKPK AC

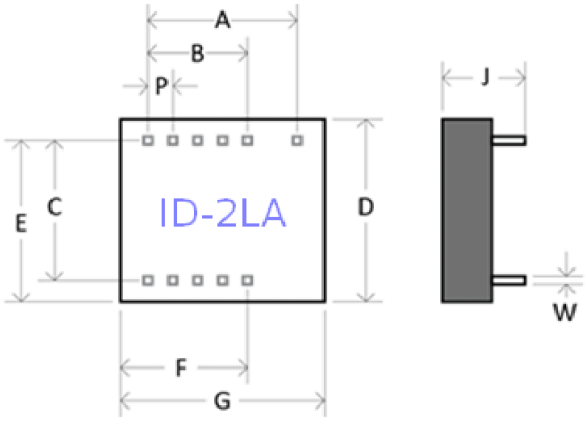
8.2 Circuit Diagram for ID-2LA



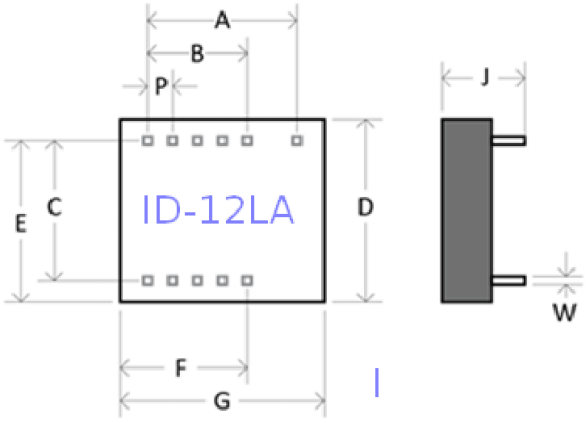
Parts List	
Part #	**Value**
R1	100R
R2	4K7
R3	2K2
C1	10uF 25v electrolytic
C2	1000uF 10v electrolytic
C3	100nF
L1	1.337mH
Q1	BC457 or similar
LED1	Read LED
LED2	Tag In Range LED
B1	2.7khz – 3kHz 5v PKPK AC

9. Dimensions (mm)

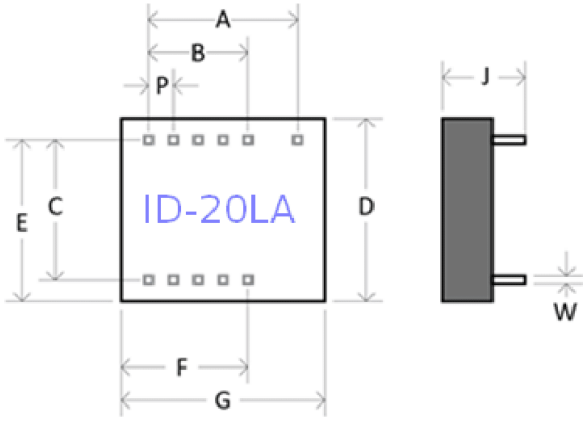
	ID-2LA		
	Nom.	Min.	Max.
A	12.0	11.6	12.4
B	8.0	7.6	8.4
C	15.0	14.6	15.4
D	20.5	20.0	21.5
E	18.5	18.0	19.2
F	14.0	13.0	14.8
G	22.0	21.6	22.4
P	2.0	1.8	2.2
H	5.92	5.85	6.6
J	9.85	9.0	10.5
W	0.66	0.62	0.67



	ID-12LA		
	Nom.	Min.	Max.
A	12.0	11.6	12.4
B	8.0	7.6	8.4
C	15.0	14.6	15.4
D	25.3	24.9	25.9
E	20.3	19.8	20.9
F	16.3	15.8	16.9
G	26.4	26.1	27.1
P	2.0	1.8	2.2
H	6.0	5.8	6.6
J	9.9	9.40	10.5
W	0.66	0.62	0.67



	ID-20LA		
	Nom.	Min.	Max.
A	12.0	11.6	12.4
B	8.0	7.6	8.4
C	15.0	14.6	15.4
D	40.3	40.0	41.0
E	27.8	27.5	28.5
F	22.2	21.9	23.1
G	38.5	38.2	39.2
P	2.0	1.8	2.2
H	6.8	6.7	7.0
J	9.85	9.4	10.6
W	0.66	0.62	0.67



10. Connection direct to a computer

Direct connection to a computer RS232 can be made by connecting Pin8 to a 1k series resistor and connecting the other end of the resistor to the computer RS232 input. The mode is called pseudo RS232. On a standard D9 socket, connect module Pin8 via the series 1k to pin2 of the D-type. Connect the ground to Pin5 on the D-type. Leave the TX pin3 open. See "Useful Information" below for free terminal download information.

Note that a +2.8v rail will result in the data outputs having a lower swing and may not be suitable for all computers.

10.1 Connection to a Processor UART

Direct connection to UART is made by connecting Pin9 to the UART Rx in pin

10.2 Connecting a Read LED

Sometimes the user may not want to drive a beeper but may still need to drive an LED. In this case a driver transistor may not be necessary because the Beeper Output Pin can supply 5mA continuously. Connect a 1k5 resistor to the Beeper Pin. This will limit the current. Connect the other end of the resistor to the LED anode and connect the cathode to ground.

11. Useful information

For general testing we suggest the user downloads a terminal program free from the internet. Here is one particularly good one to consider:

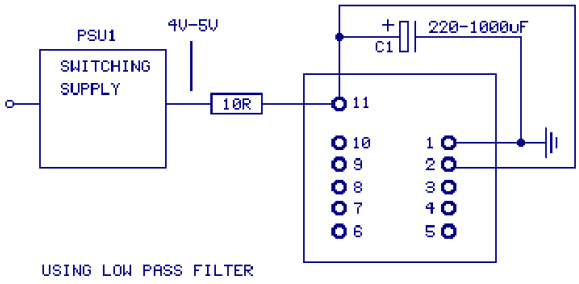
http://braypp.googlepages.com/terminal - Truly an excellent piece of software, the best terminal we have ever seen.

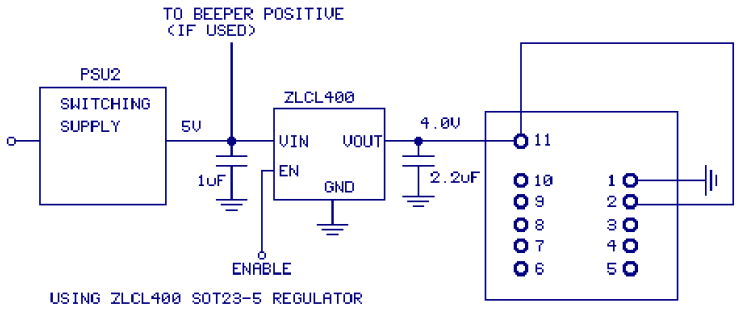
If you have any technical queries please contact your local distributor, they have all the technical resources to help you and support you. Where no local distributor exists, our technical helpline may be contacted by writing to **help@ID-Innovations.com**

12. Choice of Power Supply

The choice of power supply is very important. The ideal power supply will be a linear type such as an LM7805 or a 3.3 volt equivalent. Batteries may also be used without a regulator, a suitable arrangement can consist of a 3volt lithium cell or 3x 1.5v cells to give 4.5 volts. Note that ID-Innovations will be introducing a low-range series of micro-power reader modules (current <1mA) with the same pin out.

Switching power supplies are gaining increasing popularity, however switching supplies can present several problems for 125 KHz readers. The switching frequency must not be close to a multiple of 125 KHz or severe interference can result which will reduce the read range. Another issue with switching supplies is output voltage ripple. For good range the output ripple should be less than 3mV PKPK. This is very demanding on switching power supplies. Some recommended solutions are shown below for use with working with switching power supplies.





The upper diagram shows a low pass filter which may help increase range if the output ripple from the switching power supply is less than 10mV PKPK. The best solution, and the one which takes up less space, uses an SOT23 linear LDO regulator. The circuit employs a Zetex device which may also be powered down. Alternatively a Microchip MCP1702 could be considered, however the maximum current is limited to 50mA and may not be suitable for the ID-2LA in some circumstances.

13. Designing Coils for ID-2LA

The recommended Inductance is 1.337mH to be used with the internal tuning capacitor of 1n2. In general the bigger the antenna the better, provided the reader is generating enough field strength to excite the tag. The ID-2LA is relatively low power so a maximum coil size of 15x15cm is recommended if it is intended to read ISO cards. If the reader is intended to read glass tags the maximum coil size should be smaller, say 10x10cm.
Long read ranges, up to 30cm and more, can be achieved with Innovations Long Range Clamshell Cards using 10x10cm Antennas. There is a science to determine the exact size of an antenna but there are so many variables that in general it is best to get a general idea after which a degree of 'Try it and see' is unavoidable. If the reader is located in a position where there is a lot of heavy interference then less range cannot be avoided. In this situation the coil should be made smaller to increase the field strength and coupling. It is difficult to give actual examples of coils for hand wounding because the closeness and tightness of the winding will significantly change the inductance. A professionally wound coil will have much more inductance than a similar hand wound coil.

For those who want a starting point into practical antenna winding it was found that 71 turns on a 120mm diameter former gave an inductance of 1.33mH.
Sometimes the antenna coil is necessarily surrounded by a metallic enclosure or has an adjacent copper PCB plane. Both these can behave like a shorted turn. A shorted turn has the effect of setting up a current in opposition to the ID-2LA antenna coil current and is analogous to adding a negative inductance. Some range can be clawed back by either increasing the inductance or increasing the capacitance. If the experimenter has an inductance meter that works at 100 KHz then he can adjust the inductance back to 1.337mH by winding extra turns. Failing this an external capacitance, C3, can be increased to offset the lower inductance and bring the antenna back to tune.

Remember, that normally there is no need for external capacitance, because the ID2-LA has an internal capacitor of 1N2, which perfectly tunes the recommended inductance of 1.337mH.

Warning. Do not exceed the stated maximum voltages and currents for the ID2-LA RF output.
Anybody who wishes to be more theoretical we recommend a trip to the Microchip Website where we found an application sheet for Loop Antennas. All the same, once the user has got some theoretical knowledge, the advice of most experts is to suck it and see. http://ww1.microchip.com/downloads/en/AppNotes/00831b.pdf

14. Fine Tuning the ID-2LA

We recommend using an oscilloscope for fine-tuning. Connect the oscilloscope to observe the 125kHz AC voltage across the coil. Get a sizeable piece of ferrite and bring it up to the antenna loop. If the voltage increases then more inductance is required. If the voltage decreases as the ferrite is brought up to the antenna then the inductance is too great. If no ferrite is available then a piece of aluminium sheet may be used for testing in a slightly different way. Opposing currents will flow in the aluminium and it will act as a negative inductance. If the 125kH AC voltage increases as the aluminium sheet approaches the antenna then the inductance is too high. Note it may be possible that the voltage will first maximize then decrease. This means that the module is near optimum tuning. When using ferrite for test then it implies that the coil is a little under value and when using aluminium sheet it implies the coil is a little over value.

15. ID-2LA Compatibility Issues with the ID-2

With the exception of pin6, which should be left unconnected in the ID-2, and which now serves as a 'Tag in range Pin', the ID2-LA is 100% pin compatible and supply voltage compatible with the ID-2 and its read and output data functionality is also 100% compatible, the only difference is the RF drive power and tuning. The original ID-2 employed an internal 1n5 (0.0015uF) capacitor and an external 1.07mH search coil, the ID2-LA employs an internal 1n2(0.0012uF) capacitor and an external 1.337mH search coil.

The ID2-LA has a greater RF drive than the ID-2 and the internal tuning capacitor (1n2) requires an external antenna inductance of 1.337mH. If this value inductance is used an external tuning capacitor is not required.

Note that the original ID-2 search coil (value 1.07mH) may still be used if a 180pF or a 220pF external COG capacitor is added across pins 1 and 4. As with the ID-2, the ID2-LA internal tuning capacitor can be accessed on pins 1 and 4. If an external capacitor is added it should be a 100v 180pF COG or a 100v 220pF COG. The ID2-LA RF output drive is greater than the ID-2 RF drive. Users wishing to migrate to the ID2-LA should ensure that the RF voltage and current limits are not exceeded.

16. ID-12LA, ID-20LA Compatibility Issues with ID-12, ID-20

With the exception of pin6, which should be left unconnected in the ID-12 and ID-20, and which now serves as a 'Tag in range Pin', the ID12-LA and ID20-LA is 100% pin compatible and supply voltage compatible with the ID-12 and ID-20 and its read and output data functionality is also 100% compatible.

17. Layout Precautions for ID-12LA and ID-20LA

The layout of the mother board is important. Below is a short list of steps to ensure good read range.

1) Never put a copper ground plane under an ID-12LA or an ID-20LA. The antenna in the module will couple with the copper and the copper will act as a shorted turn. A shorted turn acts much like a negative inductance and will detune the antenna in the module.
2) Never run a noisy supply track under the module. The track will couple with the antenna and the noise will reduce the read range.
3) Never run the tracking of decoupling capacitors for other circuitry, processor etc., under the module.
4) Track the module supply decoupling capacitor as close to pin1 and pin11 as possible.

5) Use a linear regulator for the supply if possible. If a switching power must be used then see the section on choice of power supply.
6) If a switching supply is present on the same mother board, and uses an inductor, consider rotating the inductor to obtain the least coupling with the module antenna.
7) Preferably power the beeper from another supply. Beepers can draw large current that can cause the module supply voltage to dip momentarily causing a module brown out reset or other erratic malfunction.

Note. Low power regulators are less expensive, easier to configure, take up less room and noise is easier to control. If long read-range is not critical to your application, consider the ID-Innovations micro power range of pin compatible modules; these typically draw less than 100 micro amperes at 3v.

18. Contact Information

Head Office—Australia
ID Innovations
21 Sedges Grove
Canning Vale, W.A. 6155
Telephone: +61 8 94554615 Fax: +61 8 94553615

Distributor—P.R. China
Shanghai Jishi Identification Co. Ltd.
C/4F South building
No 829 Yi Shan Road
Shanghai 200233, P.R.China
上海宜山路829号南4楼C 邮编200233

Telephone: +86 21 64955114 Fax: +86 21 64950750

Disclaimer

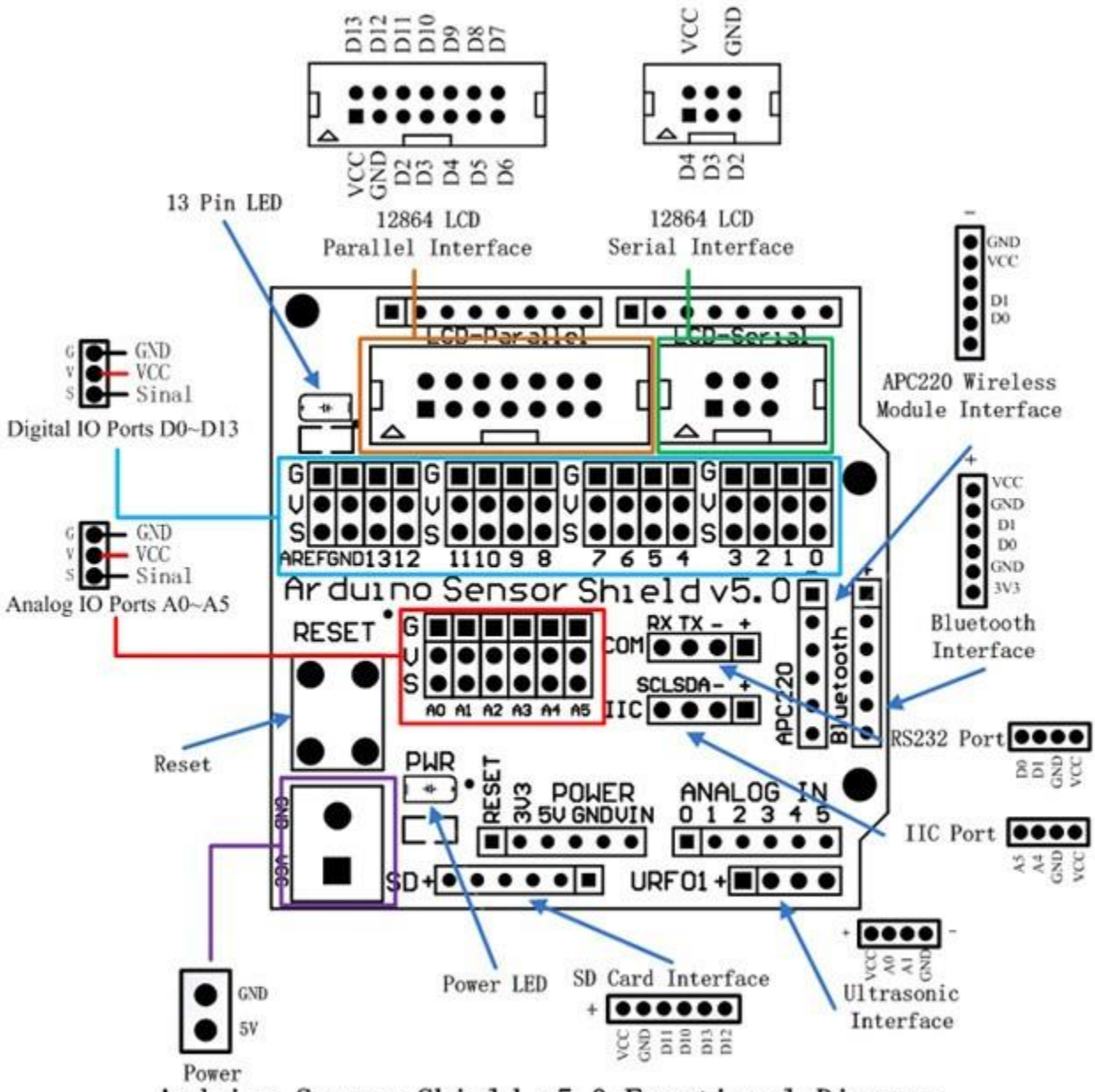
The information in this document is provided solely for the use of ID Innovations' products. There are no express or implied copyright licenses or intellectual property rights granted to design or fabricate any of ID Innovations' products based on the information provided in this document.

ID Innovations reserves the right to make changes to specifications and product descriptions at any time without notice. The specifications provided in ID Innovations' datasheets can and do vary in different applications and actual performance may vary over time.

ID Innovations' products are not intended for and will not be used in life support systems, human implantation, nuclear facilities or systems or any other application where product failure could lead to loss of life or catastrophic property damage.

ID Innovations makes no warranty, representation or guarantee regarding the suitability of its products for any purpose, and it assumes no liability, whether claim, loss, damage or expense of any kind caused in whole or in part, directly or indirectly by the inadequacy of any products for any purpose, any infringement, deficiency or defect in any products whether or not covered by any manufacturer's warranty, the use or performance of any products, any interruption or loss of service, or any special, direct, indirect, incidental or consequential damage, including, without limitation, personal injury or loss of business or profit, or other damage, whether or not ID Innovations has been informed of the possibility or likelihood of such or any other damages, arising out of the application or use of any product and information provided.

3) SainSmart Sensor Shield v5.0

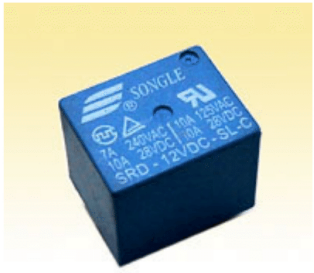


Arduino Sensor Shield v5.0 Functional Diagram

4) Songle Relay

SONGLE RELAY

松乐继电器 ® SONGLE RELAY	RELAY ISO9002	SRD



1. MAIN FEATURES

- Switching capacity available by 10A in spite of small size design for highdensity P.C. board mounting technique.
- UL,CUL,TUV recognized.
- Selection of plastic material for high temperature and better chemical solution performance.
- Sealed types available.
- Simple relay magnetic circuit to meet low cost of mass production.

2. APPLICATIONS

- Domestic appliance, office machine, audio, equipment, automobile, etc.

  (Remote control TV receiver, monitor display, audio equipment high rushing current use application.)

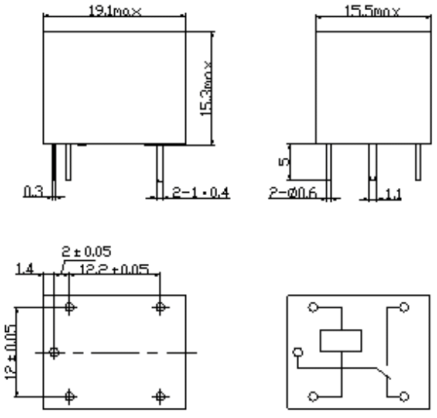
3. ORDERING INFORMATION

SRD	XX VDC	S	L	C
Model of relay	Nominal coil voltage	Structure	Coil sensitivity	Contact form
SRD	03、05、06、09、12、24、48VDC	S:Sealed type	L:0.36W	A:1 form A
				B:1 form B
		F:Flux free type	D:0.45W	C:1 form C

4. RATING

CCC	FILE NUMBER:CH0052885-2000	7A/240VDC
CCC	FILE NUMBER:CH0036746-99	10A/250VDC
UL /CUL	FILE NUMBER: E167996	10A/125VAC 28VDC
TUV	FILE NUMBER: R9933789	10A/240VAC 28VDC

5. DIMENSION(unit:mm) DRILLING(unit:mm) WIRING DIAGRAM



6. COIL DATA CHART (AT20°C)

Coil Sensitivity	Coil Voltage Code	Nominal Voltage (VDC)	Nominal Current (mA)	Coil Resistance (Ω) ±10%	Power Consumption (W)	Pull-In Voltage (VDC)	Drop-Out Voltage (VDC)	Max-Allowable Voltage (VDC)
SRD (High Sensitivity)	03	03	120	25	abt. 0.36W	75%Max.	10% Min.	120%
	05	05	71.4	70				
	06	06	60	100				
	09	09	40	225				
	12	12	30	400				
	24	24	15	1600				
	48	48	7.5	6400				
SRD (Standard)	03	03	150	20	abt. 0.45W	75% Max.	10% Min.	110%
	05	05	89.3	55				
	06	06	75	80				
	09	09	50	180				
	12	12	37.5	320				
	24	24	18.7	1280				
	48	48	10	4500	abt. 0.51W			

7. CONTACT RATING

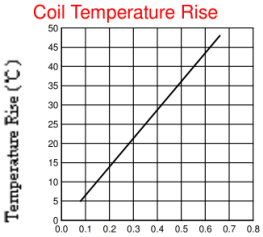
Item \ Type	SRD	
	FORM C	FORM A
Contact Capacity Resistive Load (cosΦ=1)	7A 28VDC 10A 125VAC 7A 240VAC	10A 28VDC 10A 240VAC
Inductive Load (cosΦ=0.4 L/R=7msec)	3A 120VAC 3A 28VDC	5A 120VAC 5A 28VDC
Max. Allowable Voltage	250VAC/110VDC	250VAC/110VDC
Max. Allowable Power Force	800VAC/240W	1200VA/300W
Contact Material	AgCdO	AgCdO

8. PERFORMANCE (at initial value)

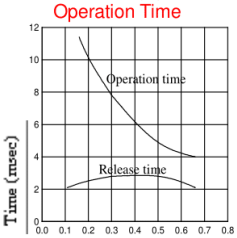
Item \ Type	SRD
Contact Resistance	100mΩ Max.
Operation Time	10msec Max.
Release Time	5msec Max.
Dielectric Strength Between coil & contact Between contacts	 1500VAC 50/60HZ (1 minute) 1000VAC 50/60HZ (1 minute)
Insulation Resistance	100 MΩ Min. (500VDC)
Max. ON/OFF Switching Mechanically Electrically	 300 operation/min 30 operation/min
Ambient Temperature	-25°C to +70°C
Operating Humidity	45 to 85% RH
Vibration Endurance Error Operation	 10 to 55Hz Double Amplitude 1.5mm 10 to 55Hz Double Amplitude 1.5mm
Shock Endurance Error Operation	 100G Min. 10G Min.
Life Expectancy Mechanically Electrically	 10^7 operations. Min. (no load) 10^5 operations. Min. (at rated coil voltage)
Weight	abt. 10grs.

9.REFERENCE DATA

Coil Temperature Rise

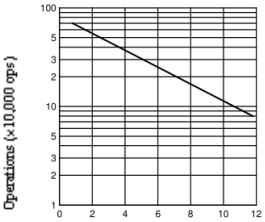


Operation Time

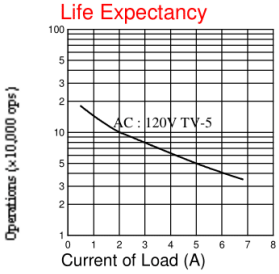


Life Expectancy

AC120V/DC24V cosΦ=1



Life Expectancy



W5100 Datasheet

Version 1.2.2

For more information, visit our website at http://www.wiznet.co.kr

Document History Information

Version	Date	Descriptions
Ver. 1.0.0	Dec. 21, 2006	Released with W5100 Launching
Ver. 1.0.1	Jan. 10, 2006	LB bit in Mode register is not used . W5100 is used only in Big-endian ordering.
Ver. 1.1.1	Jun. 19, 2007	Modified the OPMODE2-0 signals descriptions (P. 10) Modified the TEST_MODE3-0 signals description (P.11) Modified the Clock signals description (P.12) Modified the LINKLED signal description (P.12) Modified the explanation of RECV_INT in Sn_IR register (P. 27) Replaced the reset value of Sn_DHAR register (0x00 to 0xFF, P. 30) Modifted the explanation of Sn_DIPR, Sn_DPORT register(P. 31) Replaced the reset value of Sn_MSS register (0xFFFF to 0x0000, P. 31)
Ver. 1.1.2	Sep. 28, 2007	Modified the Operating temperature (P. 63)
Ver. 1.1.3	Oct. 8, 2007	Changed the typing error “MISO signal” (P. 10) Modified the SPI Timing diagram and description (P. 66)
Ver. 1.1.4	Oct. 18, 2007	Modified the diagram (P. 40)
Ver. 1.1.5	Nov. 1, 2007	Modified the Crystal Characteristics value (P. 67)
Ver. 1.1.6	Jan. 30, 2008	Modified the SEN signals description (P.10) Changed the typing error “SCLK” (P. 66)
Ver. 1.1.7	Feb. 12, 2009	Changed the typing error “memory test mode” (P. 19) Changed the description & type of clock signals (P. 12) Modified DC characteristic value(p. 63)
Ver. 1.2	Feb. 11, 2010	Add the power supply signal schematic (P. 12)
Ver. 1.2.1	Jun. 10, 2010	Add Sn_TX_WR value changing condition
Ver. 1.2.2	Jun. 28, 2010	Modify RD/WR timing diagram (P. 65, 66)

WIZnet's online Technical Support

If you have something to ask about WIZnet Products, write down your question on Q&A Board of 'Support' menu in WIZnet website (www.wiznet.co.kr). WIZnet Engineer will give an answer as soon as possible.

W5100 Datasheet

The W5100 is a full-featured, single-chip Internet-enabled 10/100 Ethernet controller designed for embedded applications where ease of integration, stability, performance, area and system cost control are required. The W5100 has been designed to facilitate easy implementation of Internet connectivity without OS. The W5100 is IEEE 802.3 10BASE-T and 802.3u 100BASE-TX compliant.

The W5100 includes fully hardwired, market-proven TCP/IP stack and integrated Ethernet MAC & PHY. Hardwired TCP/IP stack supports TCP, UDP, IPv4, ICMP, ARP, IGMP and PPPoE which has been proven in various applications for several years. 16Kbytes internal buffer is included for data transmission. No need of consideration for handling Ethernet Controller, but simple socket programming is required.

For easy integration, three different interfaces like memory access way, called direct, indirect bus and SPI, are supported on the MCU side.

Target Applications

The W5100 is well suited for many embedded applications, including:

- Home Network Devices: Set-Top Boxes, PVRs, Digital Media Adapters
- Serial-to-Ethernet: Access Controls, LED displays, Wireless AP relays, etc.
- Parallel-to-Ethernet: POS / Mini Printers, Copiers
- USB-to-Ethernet: Storage Devices, Network Printers
- GPIO-to-Ethernet: Home Network Sensors
- Security Systems: DVRs, Network Cameras, Kiosks
- Factory and Building Automations
- Medical Monitoring Equipments
- Embedded Servers

Features

- Support Hardwired TCP/IP Protocols : TCP, UDP, ICMP, IPv4 ARP, IGMP, PPPoE, Ethernet
- 10BaseT/100BaseTX Ethernet PHY embedded
- Support Auto Negotiation (Full-duplex and half duplex)
- Support Auto MDI/MDIX
- Support ADSL connection (with support PPPoE Protocol with PAP/CHAP Authentication mode)
- Supports 4 independent sockets simultaneously
- Not support IP Fragmentation
- Internal 16Kbytes Memory for Tx/Rx Buffers
- 0.18 μm CMOS technology
- 3.3V operation with 5V I/O signal tolerance
- Small 80 Pin LQFP Package
- Lead-Free Package
- Support Serial Peripheral Interface(SPI MODE 0, 3)
- Multi-function LED outputs (TX, RX, Full/Half duplex, Collision, Link, Speed)

Block Diagram

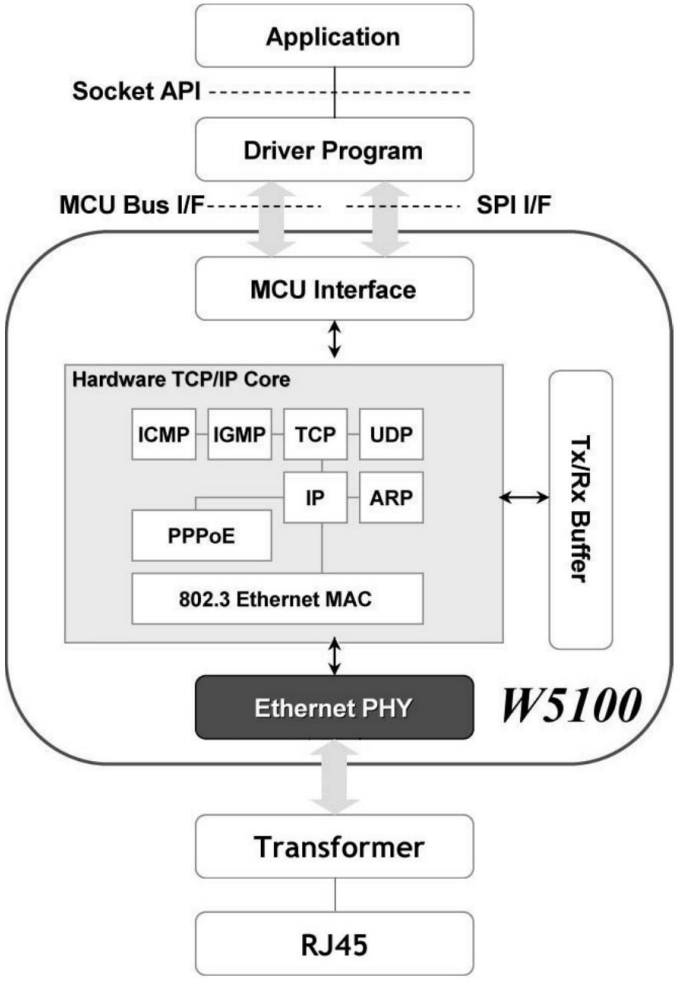
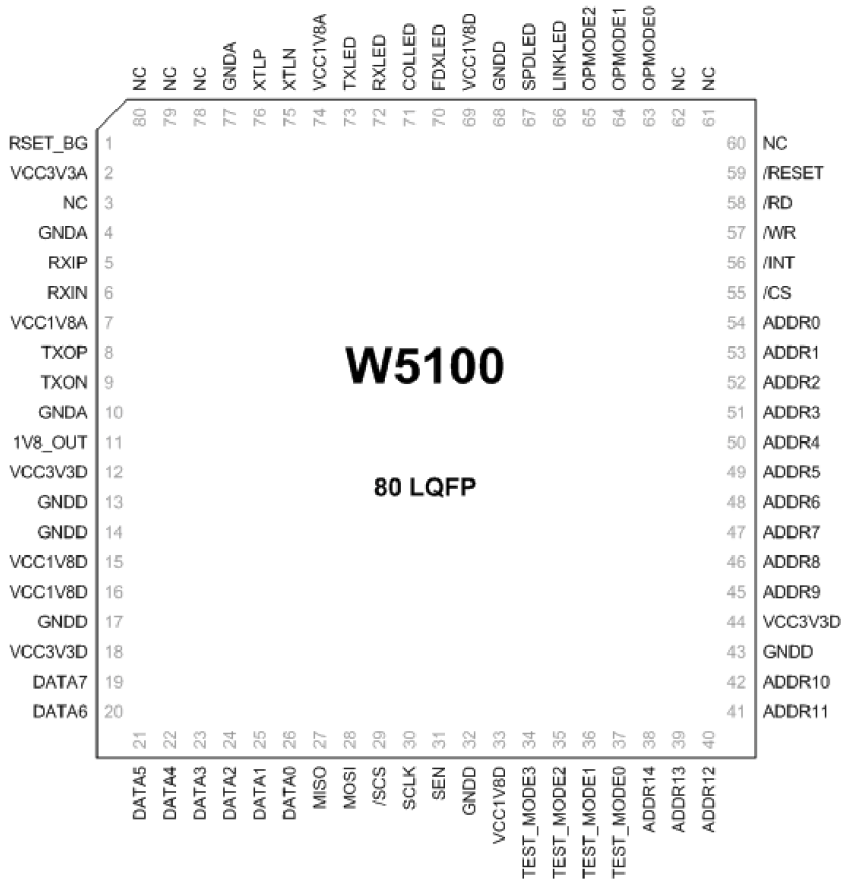


Table of Contents

1. Pin Assignment 8
1.1 MCU Interface Signals 9
1.2 PHY Signals 10
1.3 Miscellaneous Signals 11
1.4 Power Supply Signals 12
1.5 Clock Signals 12
1.6 LED Signals 13
2. Memory Map 14
3. W5100 Registers 15
3.1 common registers 15
3.2 Socket registers 16
4. Register Descriptions 20
4.1 Common Registers 20
4.2 Socket Registers 26
5. Functional Descriptions 38
5.1 Initialization 38
5.2.1 TCP 41
5.2.1.1 SERVER mode 42
5.2.1.2 CLIENT mode 49
5.2.2 UDP 51
5.2.3 IP raw 57
5.2.4 MAC raw 58
6. Application Information 60
6.1 Direct Bus Interface mode 60
6.2 Indirect Bus Interface mode 60
6.3 SPI (Serial Peripheral Interface) mode 61
6.3.1 Device Operations 62
6.3.2 Commands 62
6.3.3 Process of using general SPI Master device (According to SPI protocol) 62
7. Electrical Specifications 64
8. IR Reflow Temperature Profile (Lead-Free) 69
9. Package Descriptions 70

1. Pin Assignment



Pinout W5100

1.1 MCU Interface Signals

Symbol	Type	Pin No	Description
/RESET	I	59	RESET This pin is active Low input to initialize or re-initialize W5100. By asserting this pin low for at least 2us, all internal registers will be re-initialized to their default states.
ADDR14-0	I	38,39,40 ,41,42,45 ,46,47,48 ,49,50,51 ,52, 53,54	ADDRESS These pins are used to select a register or memory. Address pins are internally pulled down.
DATA7-0	I/O	19, 20, 21, 22, 23, 24, 25, 26	DATA These pins are used to read and write register or memory data.
/CS	I	55	CHIP SELECT Chip Select is for MCU to access to internal registers or memory. /WR and /RD select direction of data transfer. This pin is active low.
/INT	O	56	INTERRUPT This pin Indicates that W5100 requires MCU attention after socket connecting, disconnecting, data receiving or timeout. The interrupt is cleared by writing IR(Interrupt Register) or Sn_IR (Socket nth Interrupt Register). All interrupts are maskable. This pin is active low.
/WR	I	57	WRITE ENABLE Strobe from MCU to write an internal register/memory selected by ADDR[14:0]. Data is latched into the W5100 on the rising edge of this input. This signal is active low.
/RD	I	58	READ ENABLE Strobe from MCU to read an internal register/memory selected by ADDR[14:0]. This signal is active low.
SEN	I	31	SPI ENABLE This pin selects Enable/disable of the SPI Mode. Low = SPI Mode Disable

			High = SPI Mode Enable If you don't use SPI mode, then you tied this signal to '0'.
SCLK	I	30	SPI CLOCK This pin is used to SPI Clock signal Pin.
/SCS	I	29	SPI SLAVE SELECT This pin is used to SPI Slave Select signal Pin. This pin is active low
MOSI	I	28	SPI MASTER OUT SLAVE IN This pin is used to SPI MOSI signal pin.
MISO	O	27	SPI MASTER IN SLAVE OUT This pin is used to SPI MISO signal pin.

1.2 PHY Signals

Symbol	Type	Pin No	Description
RXIP	I	5	RXIP/RXIN Signal Pair The differential data from the media is received on the RXIP/RXIN signal pair.
RXIN	I	6	
TXOP	O	8	TXOP/TXON Signal Pair The differential data is transmitted to the media on the TXOP/TXIN signal pair.
TXON	O	9	
RSET_BG	O	1	PHY Off-chip resistor Connect a resistor of 12.3 kΩ±1% to the ground. Refer to the "Reference schematic".
OPMODE2-0	I	65, 64, 63	OPERATION CONTROL MODE

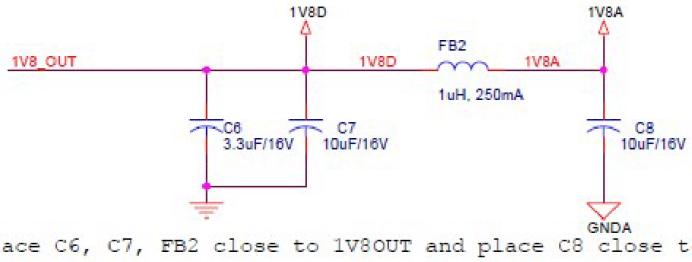
[2:0]	Description
000	Auto-negotiation enable with all capabilities
001	Auto-negotiation with 100 BASE-TX FDX/HDX ability
010	Auto-negotiation with 10 BASE-T FDX/HDX ability
011	Reserved
100	Manual selection of 100 BASE-TX FDX
101	Manual selection of 100 BASE-TX HDX
110	Manual selection of 10 BASE-T FDX
111	Manual selection of 10 BASE-T HDX

1.3 Miscellaneous Signals

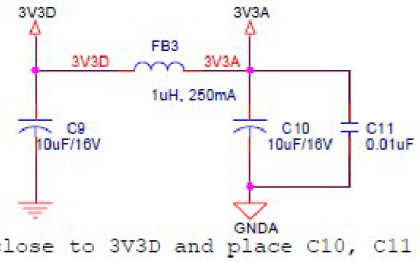
Symbol	Type	Pin No	Description
TEST_MODE3-0	I	34, 35, 36, 37	W5100 MODE SELECT Normal mode : 0000 Other test modes are internal test mode.
NC	I/O	3, 60, 61, 62, 78, 79, 80	NC TEST PIN for W5100 (for factory use only)

1.4 Power Supply Signals

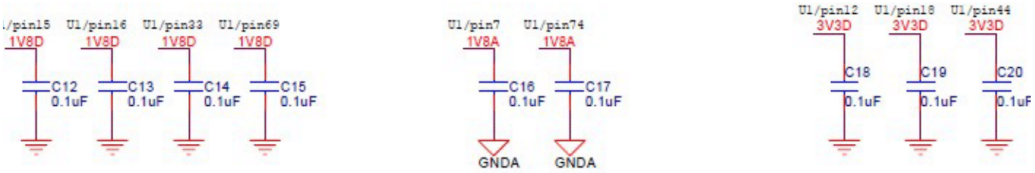
Symbol	Type	Pin No	Description
VCC3V3A	Power	2	3.3V power supply for Analog part
VCC3V3D	Power	12, 18, 44	3.3V power supply for Digital part
VCC1V8A	Power	7, 74	1.8V power supply for Analog part
VCC1V8D	Power	15, 16, 33, 69	1.8V power supply for Digital part
GNDA	Ground	4, 10, 77	Analog ground
GNDD	Ground	13, 14, 17, 32, 43, 68,	Digital ground
V18	O	11	1.8V regulator output voltage



Place C6, C7, FB2 close to 1V8OUT and place C8 close to 1V8A



Place C9, FB3 close to 3V3D and place C10, C11 close to 3V3A



ace C12, C13, C14, C15, C16, C17, C18, C19, C20 as close to each power pin as possibl

1.5 Clock Signals

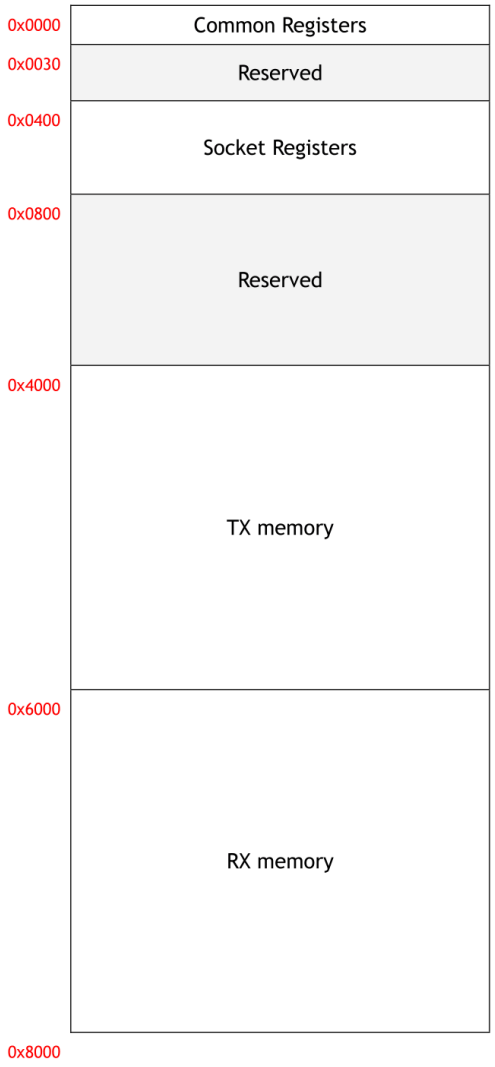
Symbol	Type	Pin No	Description
XTLP	O	76	25MHz crystal input/output A 25MHz parallel-resonant crystal is used to connect these pins to stabilize the internal oscillator If you want to use oscillator, 25MHz clock to connect XTLN signals and XTLP is open and MUST use 1.8V level oscillator.
XTLN	I	75	

1.6 LED Signals

Symbol	Type	Pin No	Description
LINKLED	O	66	Link LED Active low in link state indicates a good status for 10/100M. It is always ON when the link is OK and it flashes while in a TX or RX state.
SPDLED	O	67	Link speed LED Active low indicates the link speed is 100Mbps.
FDXLED	O	70	Full duplex LED Active low indicates the status of full-duplex mode.
COLLED	O	71	Collision LED Active low indicates the presence of collision activity.
RXLED	O	72	Receive activity LED Active low indicates the presence of receiving activity.
TXLED	O	73	Transmit activity LED Active low indicates the presence of transmitting activity.

2. Memory Map

W5100 is composed of Common Register, Socket Register, TX Memory, and RX Memory as shown below.



Memory Map

3. W5100 Registers

3.1 common registers

Address	Register
0x0000	Mode (MR)
	Gateway Address
0x0001	(GAR0)
0x0002	(GAR1)
0x0003	(GAR2)
0x0004	(GAR3)
	Subnet mask Address
0x0005	(SUBR0)
0x0006	(SUBR1)
0x0007	(SUBR2)
0x0008	(SUBR3)
	Source Hardware Address
0x0009	(SHAR0)
0x000A	(SHAR1)
0x000B	(SHAR2)
0x000C	(SHAR3)
0x000D	(SHAR4)
0x000E	(SHAR5)
	Source IP Address
0x000F	(SIPR0)
0x0010	(SIPR1)
0x0011	(SIPR2)
0x0012	(SIPR3)
0x0013	Reserved
0x0014	
0x0015	Interrupt (IR)
0x0016	Interrupt Mask (IMR)
	Retry Time
0x0017	(RTR0)
0x0018	(RTR1)
0x0019	Retry Count (RCR)

Address	Register
0x001A	RX Memory Size (RMSR)
0x001B	TX Memory Size (TMSR)
	Authentication Type in PPPoE
0x001C	(PATR0)
0x001D	(PATR1)
0x001E	
~	Reserved
0x0027	
0x0028	PPP LCP Request Timer
	(PTIMER)
0x0029	PPP LCP Magic number
	(PMAGIC)
	Unreachable IP Address
0x002A	(UIPR0)
0x002B	(UIPR1)
0x002C	(UIPR2)
0x002D	(UIPR3)
	Unreachable Port
0x002E	(UPORT0)
0x002F	(UPORT1)
0x0030	
~	Reserved
0x03FF	

3.2 Socket registers

Address	Register
0x0400	Socket 0 Mode (S0_MR)
0x0401	Socket 0 Command (S0_CR)
0x0402	Socket 0 Interrupt (S0_IR)
0x0403	Socket 0 Status (S0_SR)
0x0404	Socket 0 Source Port (S0_PORT0)
0x0405	(S0_PORT1)
0x0406	Socket 0 Destination Hardware Address (S0_DHAR0)
0x0407	(S0_DHAR1)
0x0408	(S0_DHAR2)
0x0409	(S0_DHAR3)
0x040A	(S0_DHAR4)
0x040B	(S0_DHAR5)
0x040C	Socket 0 Destination IP Address (S0_DIPR0)
0x040D	(S0_DIPR1)
0x040E	(S0_DIPR2)
0x040F	(S0_DIPR3)
0x0410	Socket 0 Destination Port (S0_DPORT0)
0x0411	(S0_DPORT1)
0x0412	Socket 0 Maximum Segment Size (S0_MSSR0)
0x0413	(S0_MSSR1)
0x0414	Socket 0 Protocol in IP Raw mode (S0_PROTO)

Address	Register
0x0415	Socket 0 IP TOS (S0_TOS)
0x0416	Socket 0 IP TTL (S0_TTL)
0x0417 ~ 0x041F	Reserved
0x0420	Socket 0 TX Free Size (S0_TX_FSR0)
0x0421	(S0_TX_FSR1)
0x0422	Socket 0 TX Read Pointer (S0_TX_RD0)
0x0423	(S0_TX_RD1)
0x0424	Socket 0 TX Write Pointer (S0_TX_WR0)
0x0425	(S0_TX_WR1)
0x0426	Socket 0 RX Received Size (S0_RX_RSR0)
0x0427	(S0_RX_RSR1)
0x0428	Socket 0 RX Read Pointer (S0_RX_RD0)
0x0429	(S0_RX_RD1)
0x042A	Reserved
0x042B	
0x042C ~ 0x04FF	Reserved

Address	Register
0x0500	Socket 1 Mode (S1_MR)
0x0501	Socket 1 Command (S1_CR)
0x0502	Socket 1 Interrupt (S1_IR)
0x0503	Socket 1 Status (S1_SR)
	Socket 1 Source Port
0x0504	(S1_PORT0)
0x0505	(S1_PORT1)
	Socket 1 Destination Hardware Address
0x0506	(S1_DHAR0)
0x0507	(S1_DHAR1)
0x0508	(S1_DHAR2)
0x0509	(S1_DHAR3)
0x050A	(S1_DHAR4)
0x050B	(S1_DHAR5)
	Socket 1 Destination IP Address
0x050C	(S1_DIPR0)
0x050D	(S1_DIPR1)
0x050E	(S1_DIPR2)
0x050F	(S1_DIPR3)
	Socket 1 Destination Port
0x0510	(S1_DPORT0)
0x0511	(S1_DPORT1)
	Socket 1 Maximum Segment Size
0x0512	(S1_MSSR0)
0x0513	(S1_MSSR1)
	Socket 1 Protocol in IP Raw mode
0x0514	(S1_PROTO)

Address	Register
0x0515	Socket 1 IP TOS (S1_TOS)
0x0516	Socket 1 IP TTL (S1_TTL)
0x0517	
~	Reserved
0x051F	
	Socket 1 TX Free Size
0x0520	(S1_TX_FSR0)
0x0521	(S1_TX_FSR1)
	Socket 1 TX Read Pointer
0x0522	(S1_TX_RD0)
0x0523	(S1_TX_RD1)
	Socket 1 TX Write Pointer
0x0524	(S1_TX_WR0)
0x0525	(S1_TX_WR1)
	Socket 1 RX Received Size
0x0526	(S1_RX_RSR0)
0x0527	(S1_RX_RSR1)
	Socket 1 RX Read Pointer
0x0528	(S1_RX_RD0)
0x0529	(S1_RX_RD1)
0x052A	Reserved
0x052B	
0x052C	
~	Reserved
0x05FF	

Address	Register
0x0600	Socket 2 Mode (S2_MR)
0x0601	Socket 2 Command (S2_CR)
0x0602	Socket 2 Interrupt (S2_IR)
0x0603	Socket 2 Status (S2_SR)
	Socket 2 Source Port
0x0604	(S2_PORT0)
0x0605	(S2_PORT1)
	Socket 2 Destination Hardware Address
0x0606	(S2_DHAR0)
0x0607	(S2_DHAR1)
0x0608	(S2_DHAR2)
0x0609	(S2_DHAR3)
0x060A	(S2_DHAR4)
0x060B	(S2_DHAR5)
	Socket 2 Destination IP Address
0x060C	(S2_DIPR0)
0x060D	(S2_DIPR1)
0x060E	(S2_DIPR2)
0x060F	(S2_DIPR3)
	Socket 2 Destination Port
0x0610	(S2_DPORT0)
0x0611	(S2_DPORT1)
	Socket 2 Maximum Segment Size
0x0612	(S2_MSSR0)
0x0613	(S2_MSSR1)
	Socket 2 Protocol in IP Raw mode
0x0614	(S2_PROTO)

Address	Register
0x0615	Socket 2 IP TOS (S2_TOS)
0x0616	Socket 2 IP TTL (S2_TTL)
0x0617	
~	Reserved
0x061F	
	Socket 2 TX Free Size
0x0620	(S2_TX_FSR0)
0x0621	(S2_TX_FSR1)
	Socket 2 TX Read Pointer
0x0622	(S2_TX_RD0)
0x0623	(S2_TX_RD1)
	Socket 2 TX Write Pointer
0x0624	(S2_TX_WR0)
0x0625	(S2_TX_WR1)
	Socket 2 RX Received Size
0x0626	(S2_RX_RSR0)
0x0627	(S2_RX_RSR1)
	Socket 2 RX Read Pointer
0x0628	(S2_RX_RD0)
0x0629	(S2_RX_RD1)
0x062A	Reserved
0x062B	
0x062C	
~	Reserved
0x06FF	

Address	Register
0x0700	Socket 3 Mode (S3_MR)
0x0701	Socket 3 Command (S3_CR)
0x0702	Socket 3 Interrupt (S3_IR)
0x0703	Socket 3 Status (S3_SR)
	Socket 3 Source Port
0x0704	(S3_PORT0)
0x0705	(S3_PORT1)
	Socket 3 Destination Hardware Address
0x0706	(S3_DHAR0)
0x0707	(S3_DHAR1)
0x0708	(S3_DHAR2)
0x0709	(S3_DHAR3)
0x070A	(S3_DHAR4)
0x070B	(S3_DHAR5)
	Socket 3 Destination IP Address
0x070C	(S3_DIPR0)
0x070D	(S3_DIPR1)
0x070E	(S3_DIPR2)
0x070F	(S3_DIPR3)
	Socket 3 Destination Port
0x0710	(S3_DPORT0)
0x0711	(S3_DPORT1)
	Socket 3 Maximum Segment Size
0x0712	(S3_MSSR0)
0x0713	(S3_MSSR1)
	Socket 3 Protocol in IP Raw mode
0x0714	(S3_PROTO)

Address	Register
0x0715	Socket 3 IP TOS (S3_TOS)
0x0716	Socket 3 IP TTL (S3_TTL)
0x0717	
~	Reserved
0x071F	
	Socket 3 TX Free Size
0x0720	(S3_TX_FSR0)
0x0721	(S3_TX_FSR1)
	Socket 3 TX Read Pointer
0x0722	(S3_TX_RD0)
0x0723	(S3_TX_RD1)
	Socket 3 TX Write Pointer
0x0724	(S3_TX_WR0)
0x0725	(S3_TX_WR1)
	Socket 3 RX Received Size
0x0726	(S3_RX_RSR0)
0x0727	(S3_RX_RSR1)
	Socket 3 RX Read Pointer
0x0728	(S3_RX_RD0)
0x0729	(S3_RX_RD1)
0x072A	Reserved
0x072B	
0x072C	
~	Reserved
0x07FF	

4. Register Descriptions

4.1 Common Registers

MR (Mode Register) [R/W] [0x0000] [0x00]

This register is used for S/W reset, ping block mode, PPPoE mode and Indirect bus I/F.

7	6	5	4	3	2	1	0
RST			PB	PPPoE		AI	IND

Bit	Symbol	Description
7	RST	**S/W Reset** If this bit is '1', internal register will be initialized. It will be automatically cleared after reset.
6	Reserved	Reserved
5	Reserved	Reserved
4	PB	**Ping Block Mode** 0 : Disable Ping block 1 : Enable Ping block If the bit is set as '1', there is no response to the ping request.
3	PPPoE	**PPPoE Mode** 0 : Disable PPPoE mode 1 : Enable PPPoE mode If you use ADSL without router or etc, you should set the bit as '1' to connect to ADSL Server. For more detail, refer to the application note, *"How to connect ADSL"*.
2	Not Used	Not Used
1	AI	**Address Auto-Increment in Indirect Bus I/F** 0 : Disable auto-increment 1 : Enable auto-increment At the Indirect Bus I/F mode, if this bit is set as '1', the address will be automatically increased by 1 whenever read and write are performed. For more detail, refer to "6.2 Indirect Bus IF Mode".
0	IND	**Indirect Bus I/F mode**

		0 : Disable Indirect bus I/F mode 1 : Enable Indirect bus I/F mode If this bit is set as '1', Indirect BUS I/F mode is set. For more detail, refer to "6. Application Information", "6.2 Indirect Bus IF Mode".

GWR (Gateway IP Address Register) [R/W] [0x0001 – 0x0004] [0x00]

This Register sets up the default gateway address.

Ex) in case of "192.168.0.1"

0x0001	0x0002	0x0003	0x0004
192 (0xC0)	168 (0xA8)	0 (0x00)	1 (0x01)

SUBR (Subnet Mask Register) [R/W] [0x0005 – 0x0008] [0x00]

This register sets up the subnet mask address.

Ex) in case of "255.255.255.0"

0x0005	0x0006	0x0007	0x0008
255 (0xFF)	255 (0xFF)	255 (0xFF)	0 (0x00)

SHAR (Source Hardware Address Register) [R/W] [0x0009 – 0x000E] [0x00]

This register sets up the Source Hardware address.

Ex) In case of "00.08.DC.01.02.03"

0x0009	0x000A	0x000B	0x000C	0x000D	0x000E
0x00	0x08	0xDC	0x01	0x02	0x03

SIPR (Source IP Address Register) [R/W] [0x000F – 0x0012] [0x00]

This register sets up the Source IP address.

Ex) in case of "192.168.0.3"

0x000F	0x0010	0x0011	0x0012
192 (0xC0)	168 (0xA8)	0 (0x00)	3 (0x03)

IR (Interrupt Register) [R] [0x0015] [0x00]

This register is accessed by the host processor to know the cause of an interrupt.

Any interrupt can be masked in the Interrupt Mask Register (IMR). The /INT signal retain low as long as any masked signal is set, and will not go high until all masked bits in this Register have been cleared.

7	6	5	4	3	2	1	0
CONFLICT	UNREACH	PPPoE	Reserved	S3_INT	S2_INT	S1_INT	S0_INT

Bit	Symbol	Description
7	CONFLICT	**IP Conflict** It is set as '1', when there is ARP request with same IP address as Source IP address. This bit is cleared to '0' by writing '1' to this bit.
6	UNREACH	**Destination unreachable** W5100 will receive ICMP(Destination Unreachable) packet if non-existing destination IP address is transmitted during UDP data transmission. (Refer to "5.2.2 UDP"). In this case, the IP address and the port number will be saved in Unreachable IP Address (UIPR) and Unreachable Port Register (UPORT), and the bit will be set as '1'. This bit will be cleared to '0' by writing '1' to this bit.
5	PPPoE	**PPPoE Connection Close** In the PPPoE Mode, if the PPPoE connection is closed, '1' is set. This bit will be cleared to '0' by writing '1' to this bit.
4	Reserved	Reserved
3	S3_INT	**Occurrence of Socket 3 Socket Interrupt** It is set in case that interrupt occurs at the socket 3. For more detailed information of socket interrupt, refer to "Socket 3 Interrupt Register (S3_IR)". This bit will be automatically cleared when S3_IR is cleared to 0x00.
2	S2_INT	**Occurrence of Socket 2 Socket Interrupt** It is set in case that interrupt occurs at the socket 2. For more detailed information of socket interrupt, refer to "Socket 2 Interrupt Register(S2_IR)". This bit will be automatically cleared when S2_IR is cleared to 0x00.
1	S1_INT	**Occurrence of Socket 1 Socket Interrupt** It is set in case that interrupt occurs at the socket 1. For more detailed information of socket interrupt, refer to "Socket 1 Interrupt Register

		(S1_IR)". This bit will be automatically cleared when S1_IR is cleared to 0x00.
0	S0_INT	**Occurrence of Socket 0 Socket Interrupt** It is set in case that interrupt occurs at the socket 0. For more detailed information of socket interrupt, refer to "Socket 0 Interrupt Register (S0_IR)". This bit will be automatically cleared when S0_IR is cleared to 0x00.

IMR (Interrupt Mask Register) [R/W] [0x0016] [0x00]

The Interrupt Mask Register is used to mask interrupts. Each interrupt mask bit corresponds to a bit in the Interrupt Register (IR). If an interrupt mask bit is set, an interrupt will be issued whenever the corresponding bit in the IR is set. If any bit in the IMR is set as '0', an interrupt will not occur though the bit in the IR is set.

7	6	5	4	3	2	1	0
IM_IR7	IM_IR6	IM_IR5	Reserved	IM_IR3	IM_IR2	IM_IR1	IM_IR0

Bit	Symbol	Description
7	IM_IR7	**IP Conflict Enable**
6	IM_IR6	**Destination unreachable Enable**
5	IM_IR5	**PPPoE Close Enable**
4	Reserved	**It should be set as '0'**
3	IM_IR3	**Occurrence of Socket 3 Socket Interrupt Enable**
2	IM_IR2	**Occurrence of Socket 2 Socket Interrupt Enable**
1	IM_IR1	**Occurrence of Socket 1 Socket Interrupt Enable**
0	IM_IR0	**Occurrence of Socket 0 Socket Interrupt Enable**

RTR (Retry Time-value Register) [R/W] [0x0017 – 0x0018] [0x07D0]

This register sets the period of timeout. Value 1 means 100us. The initial value is 2000(0x07D0). That will be set as 200ms.

Ex) For 400ms configuration, set as 4000(0x0FA0)

0x0017	0x0018
0x0F	0xA0

Re-transmission will occur if there is no response from the remote peer to the commands of CONNECT, DISCON, CLOSE, SEND, SEND_MAC and SEND_KEEP, or the response is delayed.

RCR (Retry Count Register) [R/W] [0x0019] [0x08]

This register sets the number of re-transmission. If retransmission occurs more than the number recorded in RCR, Timeout Interrupt (TIMEOUT bit of Socket *n* Interrupt Register (Sn_IR) is set as '1') will occur.

RMSR(RX Memory Size Register) [R/W] [0x001A] [0x55]

This register assigns total 8K RX Memory to each socket.

7	6	5	4	3	2	1	0
Socket 3		Socket 2		Socket 1		Socket 0	
S1	S0	S1	S0	S1	S0	S1	S0

The memory size according to the configuration of S1, S0, is as below.

S1	S0	Memory size
0	0	1KB
0	1	2KB
1	0	4KB
1	1	8KB

According to the value of S1 and S0, the memory is assigned to the sockets from socket 0 within the range of 8KB. If there is not enough memory to be assigned, the socket should not be used. The initial value is 0x55 and the 2K memory is assigned to each 4 sockets respectively.

Ex) When setting as 0xAA, the 4KB memory should be assigned to each socket.
However, the total memory size is 8KB. The memory is normally assigned to the socket 0 and 1, but not to the socket 2 and 3. Therefore, socket 2 and 3 can not be absolutely used.

Socket 3	Socket 2	Socket 1	Socket 0
0KB	0KB	4KB	4KB

TMSR(TX Memory Size Register) [R/W] [0x001B] [0x55]

This register is used in assigning total 8K TX Memory to sockets. Configuration can be done in the same way of RX Memory Size Register (RMSR). The initial value is 0x55 and it is to assign 2K memory to 4 sockets respectively.

PATR (Authentication Type in PPPoE mode) [R] [0x001C-0x001D] [0x0000]

This register notifies authentication method that has been agreed at the connection with PPPoE Server. W5100 supports two types of Authentication method - PAP and CHAP.

Value	Authentication Type
0xC023	PAP
0xC223	CHAP

PTIMER (PPP Link Control Protocol Request Timer Register) [R/W] [0x0028] [0x28]

This register indicates the duration for sending LCP Echo Request. Value 1 is about 25ms.

Ex) in case that PTIMER is 200,

200 * 25(ms) = 5000(ms) = 5 seconds

PMAGIC (PPP Link Control Protocol Magic number Register) [R/W] [0x0029] [0x00]

This register is used in Magic number option during LCP negotiation. Refer to the application note, *"How to connect ADSL"*.

UIPR (Unreachable IP Address Register) [R] [0x002A – 0x002D] [0x00]

In case of data transmission using UDP (refer to "5.2.2. UDP"), if transmitting to non-existing IP address, ICMP (Destination Unreachable) packet will be received. In this case, that IP address and port number will be saved in the Unreachable IP Address Register(UIPR) and Unreachable Port Register(UPORT) respectively.

Ex) in case of "192.168.0.11",

0x002A	0x002B	0x002C	0x002D
192 (0xC0)	168 (0xA8)	0 (0x00)	11 (0x0B)

UPORT (Unreachable Port Register) [R] [0x002E – 0x002F] [0x0000]

Refer to Unreachable IP Address Register (UIPR)

Ex) In case of 5000(0x1388),

0x002E	0x002F
0x13	0x88

4.2 Socket Registers

S*n*[1]_MR (Socket *n* Mode Register) [R/W] [0x0400, 0x0500, 0x0600, 0x0700] [0x00][2]

This register sets up socket option or protocol type for each socket.

7	6	5	4	3	2	1	0
MULTI		ND / MC		P3	P2	P1	P0

Bit	Symbol	Description
7	MULTI	**Multicasting** 0 : disable Multicasting 1 : enable Multicasting It is applied only in case of UDP. For using multicasting, write multicast group address to Socket *n* Destination IP and multicast group port number to Socket *n* Destination Port Register, before OPEN command.
6	Reserved	Reserved
5	ND/MC	**Use No Delayed ACK** 0 : Disable No Delayed ACK option 1 : Enable No Delayed ACK option, It is applied only in case of TCP. If this bit is set as '1', ACK packet is transmitted whenever receiving data packet from the peer. If this bit is cleared to '0', ACK packet is transmitted according to internal Timeout mechanism. **Multicast** 0 : using IGMP version 2 1 : using IGMP version 1 It is applied only in case of MULTI bit is '1'
4	Reserved	Reserved
3 2	P3 P2	**Protocol** Sets up corresponding socket as TCP, UDP, or IP RAW mode

P3	P2	P1	P0	Meaning
0	0	0	0	Closed
0	0	0	1	TCP

[1] *n* is socket number (0, 1, 2, 3).
[2] [Read/Write] [address of socket 0, address of socket 1, address of socket 2, address of socket 3] [Reset value]

1	P1	0	0	1	0	UDP
		0	0	1	1	IPRAW
0	P0					

*** In case of socket 0, MACRAW and PPPoE mode exist.**

P3	P2	P1	P0	Meaning
0	1	0	0	MACRAW
0	1	0	1	PPPoE

S*n*_CR (Socket *n* Command Register) [R/W] [0x0401, 0x0501, 0x0601, 0x0701] [0x00]

This register is utilized for socket *n* initialization, close, connection establishment, termination, data transmission and command receipt. After performing the commands, the register value will be automatically cleared to 0x00.

Value	Symbol	Description
0x01	OPEN	It is used to initialize the socket. According to the value of Socket *n* Mode Register (S*n*_MR), Socket *n* Status Register(S*n*_SR) value is changed to SOCK_INIT, SOCK_UDP, SOCK_IPRAW, or SOCK_MACRAW. For more detail, refer to 5. Functional Description.
0x02	LISTEN	It is only used in TCP mode. It changes the value of Socket *n* Status Register (Sn_SR) to SOCK_LISTEN in order to wait for a connection request from any remote peer (TCP Client). For more detail, refer to 5.2.1.1 SERVER mode.
0x04	CONNECT	It is only used in TCP mode. It sends a connection request to remote peer(TCP SERVER). If the connection is failed, Timeout interrupt will occur. For more detail, refer to 5.2.1.2 CLIENT mode.
0x08	DISCON	It is only used in TCP mode. It sends a connection termination request. If connection termination is failed, Timeout interrupt will occur. For more detail, refer to 5.2.1.1 SERVER mode. ** In case of using CLOSE command instead of DISCON, only the value of Socket n Status Register(Sn_SR) is changed to SOCK_CLOSED without the connection termination process.*
0x10	CLOSE	It is used to close the socket. It changes the value of Socket *n* Status Register(S*n*_SR) to SOCK_CLOSED.

0x20	SEND	It transmits the data as much as the increased size of Socket *n* TX Write Pointer. For more detail, refer to Socket *n* TX Free Size Register (S*n*_TX_FSR), Socket *n* TX Write Pointer Register(S*n*_TX_WR), and Socket *n* TX Read Pointer Register(S*n*_TX_RR) or 5.2.1.1. SERVER mode.
0x21	SEND_MAC	It is used in UDP mode. The basic operation is same as SEND. Normally SEND operation needs Destination Hardware Address that is received in ARP(Address Resolution Protocol) process. SEND_MAC uses Socket *n* Destination Hardware Address(S*n*_DHAR) that is written by users without ARP process.
0x22	SEND_KEEP	It is only used in TCP mode. It checks the connection status by sending 1byte data. If the connection is already terminated or peer has no response, Timeout interrupt will occur.
0x40	RECV	Receiving is processed with the value of Socket *n* RX Read Pointer Register(S*n*_RX_RD). For more detail, refer to 5.2.1.1 SERVER mode Receiving Process with Socket *n* RX Received Size Register (S*n*_RX_RSR), Socket *n* RX Write Pointer Register(S*n*_RX_WR), and Socket *n* RX Read Pointer Register(S*n*_RX_RD)

S*n*_IR (Socket *n* Interrupt Register) [R] [0x0402, 0x0502, 0x0602, 0x0702] [0x00]

This register is used for notifying connection establishment and termination, receiving data and Timeout. The Socket *n* Interrupt Register must be cleared by writing '1'.

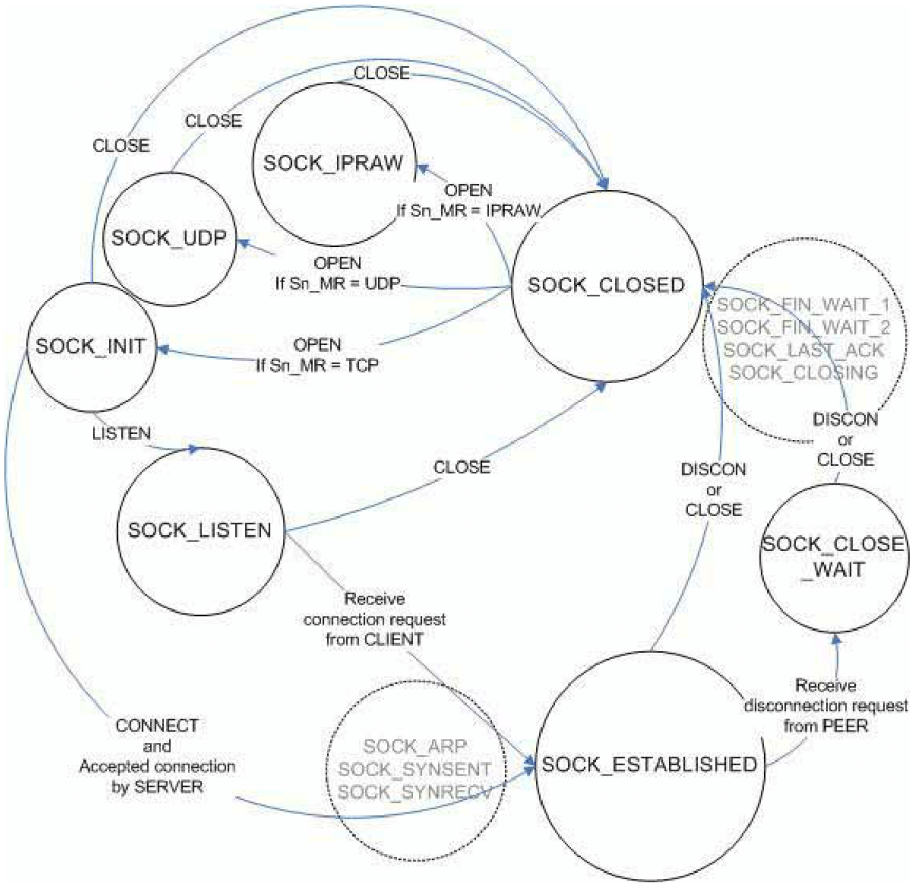
7	6	5	4	3	2	1	0
Reserved	Reserved	Reserved	SEND_OK	TIMEOUT	RECV	DISCON	CON

Bit	Symbol	Description
7	Reserved	Reserved
6	Reserved	Reserved
5	Reserved	Reserved
4	SEND_OK	It is set as '1' if send operation is completed.
3	TIMEOUT	It is set as '1' if Timeout occurs during connection establishment or termination and data transmission.
2	RECV	It is set as '1' whenever W5100 receives data. And it is also set as '1'

		if received data remains after execute CMD_RECV command.
1	DISCON	It is set as '1' if connection termination is requested or finished.
0	CON	It is set as '1' if connection is established.

S*n*_SR (Socket *n* Status Register) [R] [0x0403, 0x0503, 0x0603, 0x0703] [0x00]

This register has the status value of socket *n*. The main status is shown in the below diagram.



Value	Symbol	Description
0x00	SOCK_CLOSED	It is shown in case that CLOSE commands are given to S*n*_CR, and Timeout interrupt is asserted or connection is terminated. In this SOCK_CLOSED status, no operation occurs and all resources for the connection is released.

0x13	SOCK_INIT	It is shown in case that S*n*_MR is set as TCP and OPEN commands are given to S*n*_CR. This is the initial step for TCP connection establishment of a socket. In this SOCK_INIT status, the command type (LISTEN or CONNECT) of Sn_CR will decide the operation type – TCP server mode or Client mode.
0x14	SOCK_LISTEN	It is shown in case that LISTEN commands are given to S*n*_CR at the SOCK_INIT status. The related socket will operate as TCP Server mode, and become ESTBLISHED status if connection request is normally received.
0x17	SOCK_ESTABLISHED	It is shown in case that connection is established. In this status, TCP data is transmitted and received.
0x1C	SOCK_CLOSE_WAIT	It is shown in case that connection termination request is received from peer host. At this status, the Acknowledge message has been received from the peer, but not disconnected. The connection can be closed by receiving the DICON or CLOSE commands.
0x22	SOCK_UDP	It is shown in case that OPEN commands are given to S*n*_CR when S*n*_MR is set as UDP. As this status does not need the connection process with peer, the data can be directly transmitted and received.
0x32	SOCK_IPRAW	It is shown in case that OPEN commands are given to S*n*_CR when S*n*_MR is set as IPRAW. At the IPRAW status, the following protocols of IP Header are not processed. Refer to “IP RAW” for more information.
0x42	SOCK_MACRAW	It is shown in case that OPEN commands are given to S0_CR when S0_MR is set as MACRAW. At the MAC RAW status, there is no protocol process for a packet. For more information, refer to “MAC RAW”.
0x5F	SOCK_PPPOE	It is shown in case that OPEN commands are given to S0_CR when S0_MR is set as PPPoE.

Below is shown during changing the status.

Value	Symbol	Description
0x15	SOCK_SYNSENT	It is shown in case that CONNECT commands are given to Socket *n* Command Register(S*n*_CR) at the SOCK_INIT status. It is automatically changed to SOCK_ESTABLISH when the

		connection is established.
0x16	SOCK_SYNRECV	It is shown in case that connection request is received from remote peer(CLIENT). It normally responds to the requests and changes to SOCK_ESTABLISH.
0x18	SOCK_FIN_WAIT	It is shown in the process of connection termination. If the termination is normally processed or Timeout interrupt is asserted, it will be automatically changed to SOCK_CLOSED.
0x1A	SOCK_CLOSING	
0X1B	SOCK_TIME_WAIT	
0X1D	SOCK_LAST_ACK	
0x11 0x21 0x31	SOCK_ARP	It is shown when ARP Request is sent in order to acquire hardware address of remote peer when it sends connection request in TCP mode or sends data in UDP mode. If ARP Reply is received, it changes to the status, SOCK_SYNSENT, SOCK_UDP or SOCK_ICMP, for the next operation.

S*n*_PORT (Socket *n* Source Port Register) [R/W] [0x0404-0x0405, 0x0504-0x0505, 0x0604-0x0605, 0x0704-0x0705] [0x00]

This register sets the Source Port number for each Socket when using TCP or UDP mode, and the set-up needs to be made before executing the OPEN Command.

Ex) In case of Socket 0 Port = 5000(0x1388), configure as below,

0x0404	0x0405
0x13	0x88

S*n*_DHAR (Socket *n* Destination Hardware Address Register) [R/W] [0x0406-0x040B, 0x0506-0x050B, 0x0606-0x060B, 0x0706-0x070B] [0xFF]

This register sets the Destination Hardware address of each Socket.

Ex) In case of Socket 0 Destination Hardware address = 08.DC.00.01.02.10, configuration is as below,

0x0406	0x0407	0x0408	0x0409	0x040A	0x040B
0x08	0xDC	0x00	0x01	0x02	0x0A

S*n*_DIPR (Socket *n* Destination IP Address Register) [R/W] [0x040C-0x040F, 0x050C-0x050F, 0x060C-0x060F, 0x070C-0x070F] [0x00]

This register sets the Destination IP Address of each Socket to be used in setting the TCP connection. In active mode, IP address needs to be set before executing the Connect command. In passive mode, W5100 sets up the connection and then is internally updated with peer IP.

In UDP mode, this register value decided to user's written value after receiving peer's ARP response. Before receiving peer's ARP response, this register has reset value.

Ex) In case of Socket 0 Destination IP address = 192.168.0.11, configure as below.

0x040C	0x040D	0x040E	0x040F
192 (0xC0)	168 (0xA8)	0 (0x00)	11 (0x0B)

S*n*_DPORT (Socket *n* Destination Port Register) [R/W] [0x0410-0x0411, 0x0510-0x0511, 0x0610-0x0611, 0x0710-0x0711] [0x00]

This register sets the Destination Port number of each socket to be used in setting the TCP connection. In active mode, port number needs to be set before executing the Connect command. In passive mode, W5100 sets up the connection and then is internally updated with peer port number.

In UDP mode, this register value decided to user's written value after receiving peer's ARP response. Before receiving peer's ARP response, this register has reset value.

Ex) In case of Socket 0 Destination Port = 5000(0x1388), configure as below,

0x0410	0x0411
0x13	0x88

S*n*_MSS (Socket *n* Maximum Segment Size Register) [R/W] [0x0412-0x0413, 0x0512-0x0513, 0x0612-0x0613, 0x0712-0x0713] [0x 0000]

This register is used for MSS (Maximum Segment Size) of TCP, and the register displays MSS set by the other party when TCP is activated in Passive Mode.

Ex) In case of Socket 0 MSS = 1460(0x05B4), configure as below,

0x0412	0x0413
0x05	0xB4

S*n*_PROTO (Socket *n* IP Protocol Register) [R/W] [0x0414, 0x0514, 0x0614, 0x0714] [0x00]

This IP Protocol Register is used to set up the Protocol Field of IP Header at the IP Layer RAW Mode. There are several protocol numbers defined in advance by registering to IANA. For the overall list of upper level protocol identification number that IP is using, refer to online documents of IANA (http://www.iana.org/assignments/protocol-numbers).

Ex) Internet Control Message Protocol (ICMP) = 0x01, Internet Group Management Protocol = 0x02

S*n*_TOS (Socket *n* IP Type Of Service Register) [R/W] [0x0415,0x0515,0x0615,0x0715] [0x00]

This register sets up at the TOS(Type of Service) Field of IP Header.

S*n*_TTL (Socket *n* IP Time To Live Register) [R/W] [0x0416,0x0516,0x0616,0x0716] [0x80]

This register sets up at the TTL(Time To Live) Field of IP Header.

S*n*_TX_FSR (Socket *n* TX Free Size Register) [R] [0x0420-0x0421, 0x0520-0x0521, 0x0620-0x0621, 0x0720-0x0721] [0x0800]

This register notifies the information of data size that user can transmit. For data transmission, user should check this value first and control the size of transmitting data. *When checking this register, user should read upper byte(0x0420,0x0520,0x0620,0x0720) first and lower byte(0x0421,0x0521,0x0621,0x0721) later to get the correct value.*

Ex) In case of 2048(0x0800) in S0_TX_FSR,

0x0420	0x0421
0x08	0x00

Total size can be decided according to the value of TX Memory Size Register. In the process of transmission, it will be reduced by the size of transmitting data, and automatically increased after transmission finished.

S*n*_TX_RR (Socket *n* TX Read Pointer Register) [R] [0x0422-0x0423, 0x0522-0x0523, 0x0622-0x0623, 0x0722-0x0723] [0x0000]

This register shows the address that transmission is finished at the TX Memory. With the SEND command of Socket *n* Command Register, it transmits data from current S*n*_TX_RR to S*n*_TX_WR and automatically changes after transmission is finished. Therefore, after transmission is finished, S*n*_TX_RR and S*n*_TX_WR will have same value. When reading this register, user should read upper byte (0x0422, 0x0522, 0x0622, 0x0722) first and lower byte (0x0423, 0x0523, 0x0623, 0x0723) later to get the correct value.

S*n*_TX_WR (Socket *n* TX Write Pointer Register) [R/W] [0x0424-0x0425, 0x0524-0x0525, 0x0624-0x0625, 0x0724-0x0725] [0x0000]

This register offers the location information to write the transmission data. When reading this register, user should read upper byte (0x0424, 0x0524, 0x0624, 0x0724) first and lower byte (0x0425, 0x0525, 0x0625, 0x0725) later to get the correct value.

Caution: This register value is changed after the successfully executed the send command to Sn_CR.

Ex) In case of 2048(0x0800) in S0_TX_WR,

0x0424	0x0425
0x08	0x00

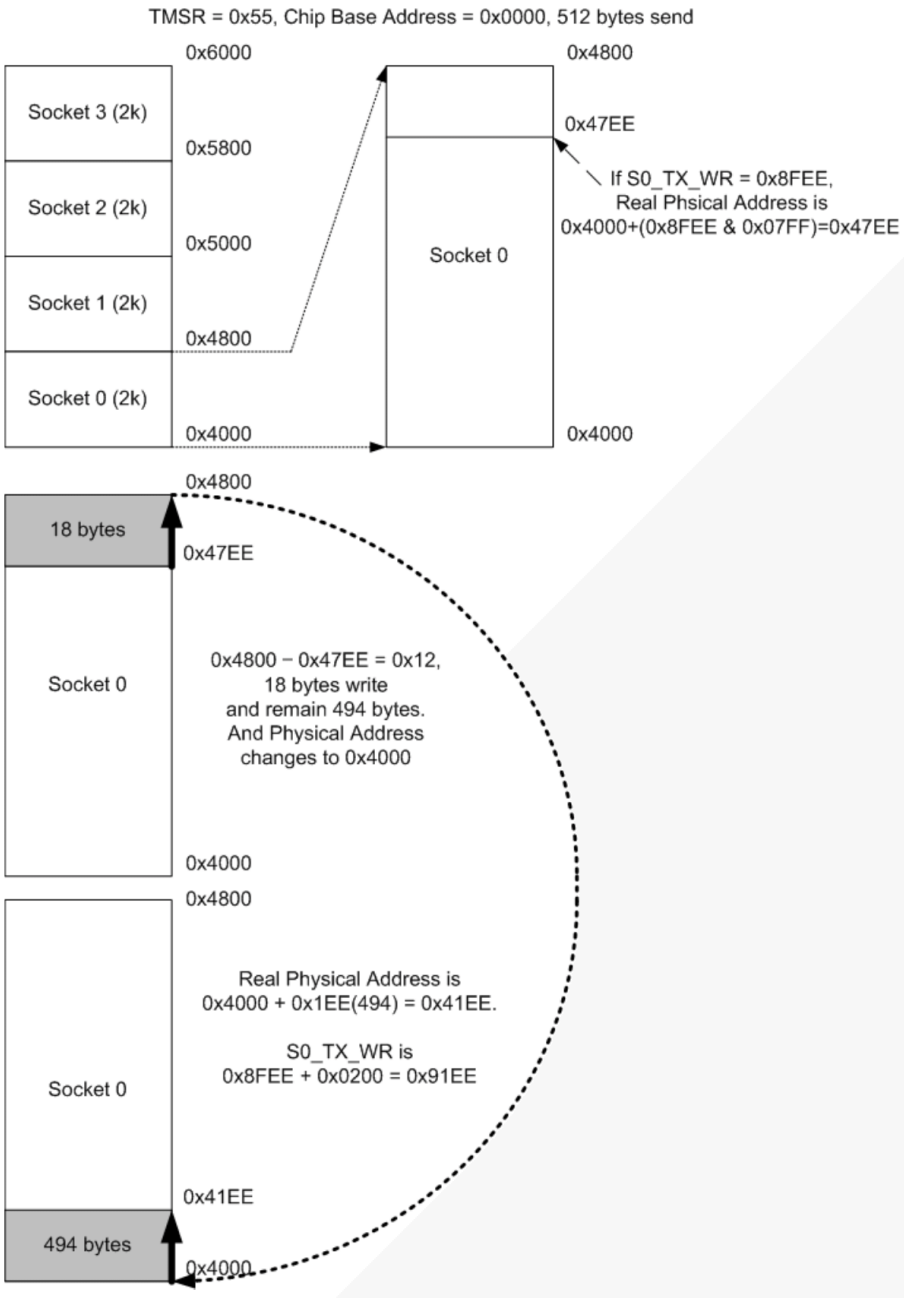
But this value itself is not the physical address to write. So, the physical address should be calculated as follow.

1. Socket *n* TX Base Address (hereafter we'll call *gSn_TX_BASE)* and Socket *n* TX Mask Address (hereafter we'll call *gSn_TX_MASK)* are calculated on TMSR value. *Refer to the psedo code of the Initialization if the detail is needed.*
2. The bitwise-AND operation of two values, Sn_TX_WR and *gSn_TX_MASK* give result the offset address(hereafter we'll call *get_offset*) in TX memory range of the socket.
3. Two values *get_offset* and *gSn_TX_BASE* are added together to give result the physical address(hereafter, we'll call *get_start_address*).

Now, write the transmission data to *get_start_address* as large as you want. (* There's a case that it exceeds the TX memory upper-bound of the socket while writing. In this case, write the transmission data to the upper-bound, and change the physical address to the *gSn_TX_BASE*. Next, write the rest of the transmission data.)

After that, be sure to increase the Sn_TX_WR value as much as the data size that indicates the size of writing data. Finally, give SEND command to Sn_CR(Socket *n* Command Register).

Refer to the psedo code of the transmission part on TCP Server mode if the detail is needed.



Calculate physical address

S*n*_RX_RSR (RX Received Size Register) [R] [0x0426-0x0427, 0x0526-0x0527, 0x0626-0x0627, 0x0726-0x0727] [0x0000]

This register notifies the data size received in RX Memory. As this value is internally calculated with the values of S*n*_RX_RD and S*n*_RX_WR, it is automatically changed by RECV command of Socket *n* Command Register(S*n*_CR) and receiving data for remote peer. *When reading this register, user should read upper byte(0x0426,0x0526,0x0626,0x0726) first and lower byte(0x0427,0x0527,0x0627,0x0727) later to get the correct value.*

Ex) In case of 2048(0x0800) in S0_RX_RSR,

0x0426	0x0427
0x08	0x00

The total size of this value can be decided according to the value of RX Memory Size Register.

S*n*_RX_RD (Socket *n* RX Read Pointer Register) [R/W] [0x0428-0x0429, 0x0528-0x0529, 0x0628-0x0629, 0x0728-0x0729] [0x0000]

This register offers the location information to read the receiving data. When reading this register, user should read upper byte (0x0428, 0x0528, 0x0628, 0x0728) first and lower byte (0x0429, 0x0529, 0x0629, 0x0729) later to get the correct value. It has a random value as its initial value.

Caution: This register value is changed after the successfully executed receive command to Sn_CR.

Ex) In case of 2048(0x0800) in S0_RX_RD,

0x0428	0x0429
0x08	0x00

But this value itself is not the physical address to read. So, the physical address should be calculated as follow.

1. Socket *n* RX Base Address (hereafter we'll call *gSn_RX_BASE)* and Socket *n* RX Mask Address (hereafter we'll call *gSn_RX_MASK)* are calculated on RMSR value. *Refer to the pseudo code of the 5.1 Initialization if the detail is needed.*
2. The bitwise-AND operation of two values, Sn_RX_RD and *gSn_RX_MASK* give result the offset address(hereafter we'll call *get_offset)*, in RX memory range of the socket.
3. Two values *get_offset* and *gSn_RX_BASE* are added together to give result the physical address(hereafter, we'll call *get_start_address).*

Now, read the receiving data from *get_start_address* as large as you want. (* There's a case that it exceeds the RX memory upper-bound of the socket while reading. In this case, read the receiving data to the upper-bound, and change the physical address to the *gSn_RX_BASE*.

Next, read the rest of the receiving data.)

After that, be sure to increase the Sn_RX_RD value as large as the data size that indicates the size of reading data. (* Must not increase more than the size of received data. So must check Sn_RX_RSR before receiving process.) Finally, give RECV command to Sn_CR(Socket *n* Command Register).

Refer to the pseudo code of the receiving part on TCP Server mode if the detail is needed.

5. Functional Descriptions

By setting some register and memory operation, W5100 provides internet connectivity. This chapter describes how it can be operated.

5.1 Initialization

- Basic Setting

For the W5100 operation, select and utilize appropriate registers shown below.

1. Mode Register (MR)
2. Interrupt Mask Register (IMR)
3. Retry Time-value Register (RTR)
4. Retry Count Register (RCR)

For more information of above registers, refer to the “Register Descriptions”.

- Setting network information

Below register is for basic network configuration information to be configured according to the network environment.

1. Gateway Address Register (GAR)
2. Source Hardware Address Register (SHAR)
3. Subnet Mask Register (SUBR)
4. Source IP Address Register (SIPR)

The Source Hardware Address Register (SHAR) is the H/W address to be used in MAC layer, and can be used with the address that manufacturer has been assigned. The MAC address can be assigned from IEEE. For more detail, refer to IEEE homepage.

- Set socket memory information

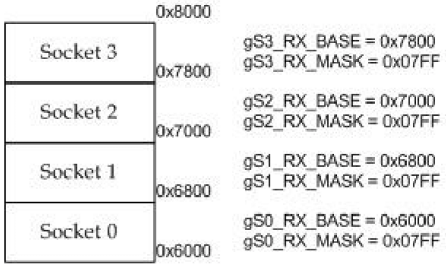
This stage sets the socket tx/rx memory information. The base address and mask address of each socket are fixed and saved in this stage.

```
In case of, assign 2K rx memory per socket.
{
    RMSR = 0x55; // assign 2K rx memory per socket.
    gS0_RX_BASE = chip_base_address + RX_memory_base_address(0x6000);
    gS0_RX_MASK = 2K – 1 ; // 0x07FF, for getting offset address within assigned socket 0 RX
memory.
    gS1_RX_BASE = gS0_BASE + (gS0_MASK + 1);
    gS1_RX_MASK = 2K – 1 ;
```

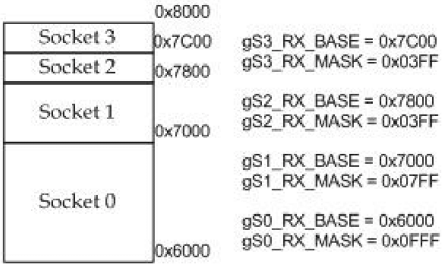
```
    gS2_RX_BASE = gS1_BASE + (gS1_MASK + 1);
    gS2_RX_MASK = 2K - 1 ;
    gS3_RX_BASE = gS2_BASE + (gS2_MASK + 1);
    gS3_RX_MASK = 2K - 1 ;
    TMSR = 0x55; // assign 2K tx memory per socket.
    Same method, set gS0_TX_BASE, gS0_TX_MASK, gS1_TX_BASE, gS1_TX_MASK,
    gS2_TX_BASE, gS2_TX_MASK, gS3_TX_BASE and gS3_TX_MASK.
}
```

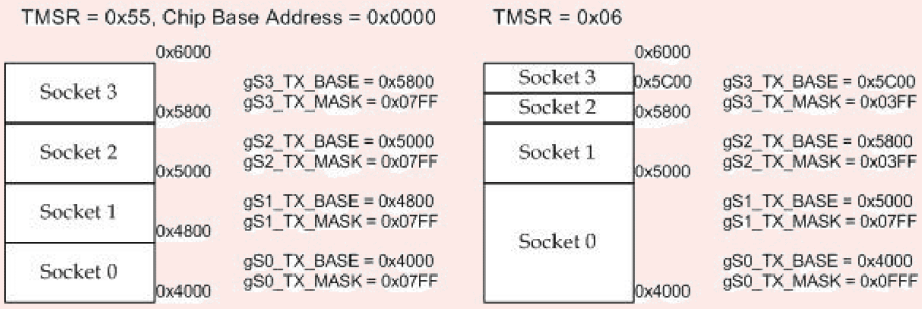
```
In case of, assign 4K,2K,1K,1K.
{
    RMSR = 0x06; // assign 4K,2K,1K,1K rx memory per socket.
    gS0_RX_BASE = chip_base_address + RX_memory_base_address(0x6000);
    gS0_RX_MASK = 4K - 1 ; // 0x0FFF, for getting offset address within assigned socket 0 RX
memory.
    gS1_RX_BASE = gS0_BASE + (gS0_MASK + 1);
    gS1_RX_MASK = 2K - 1 ; // 0x07FF
    gS2_RX_BASE = gS1_BASE + (gS1_MASK + 1);
    gS2_RX_MASK = 1K - 1 ; // 0x03FF
    gS3_RX_BASE = gS2_BASE + (gS2_MASK + 1);
    gS3_RX_MASK = 1K - 1 ; // 0x03FF
    TMSR = 0x06; // assign 4K,2K,1K,1K rx memory per socket.
    Same method, set gS0_TX_BASE, gS0_TX_MASK, gS1_TX_BASE, gS1_TX_MASK,
    gS2_TX_BASE, gS2_TX_MASK, gS3_TX_BASE and gS3_TX_MASK.
}
```

RMSR = 0x55, Chip Base Address = 0x0000



RMSR = 0x06



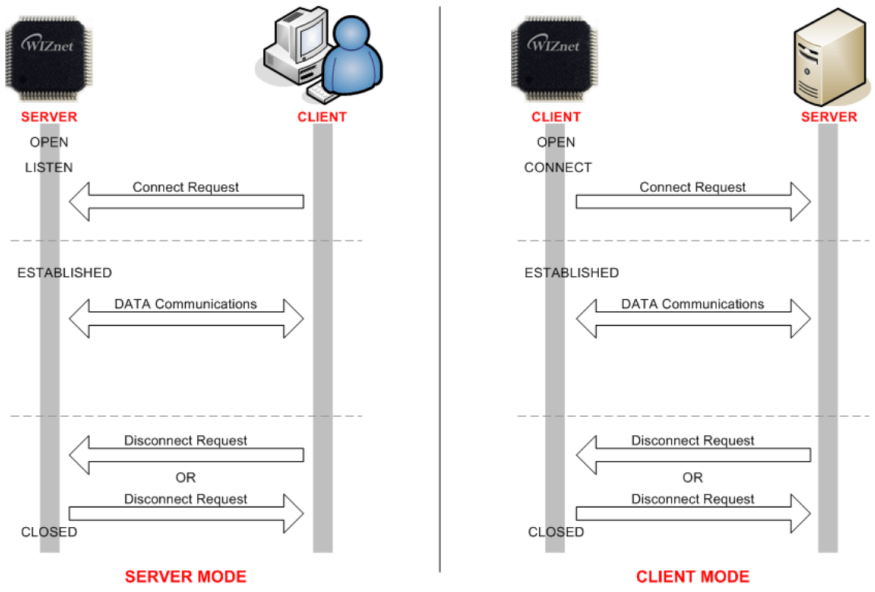


5.2 Data Communications

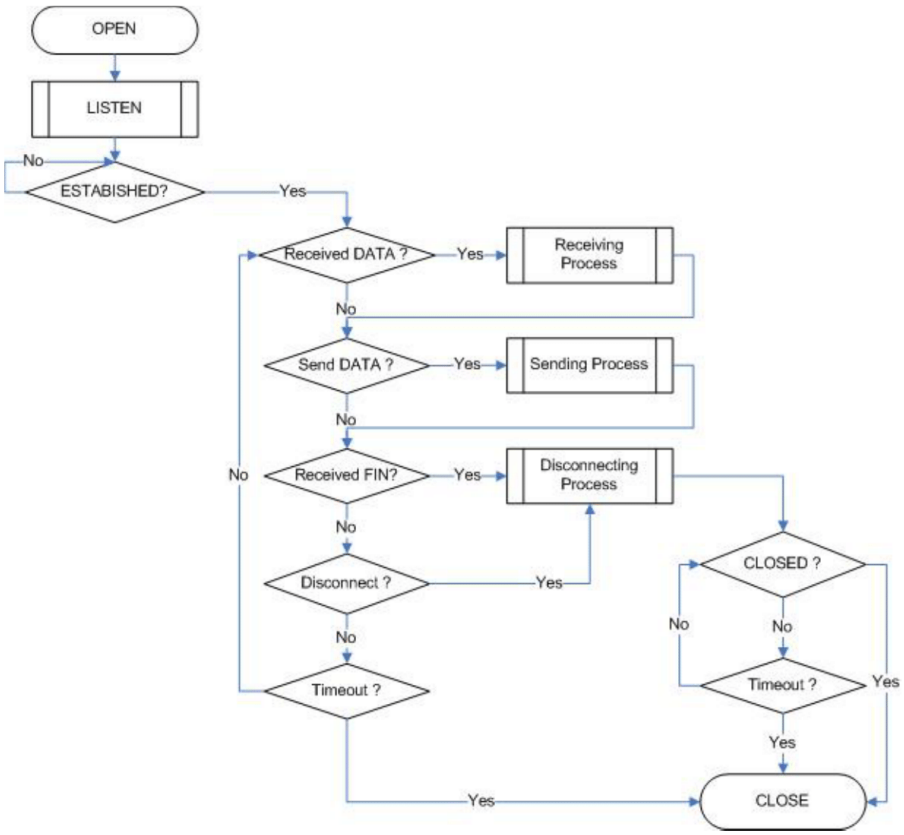
Data communication is available through TCP, UDP, IP-Raw and MAC-Raw . In order to select it, configure protocol field of Socket *n* Mode Register(S*n*_MR) of the communication sockets (W5100 supports total 4 sockets).

5.2.1 TCP

TCP is the connection based communication method that will establish connection in advance and deliver the data through the connection by using IP Address and Port number of the systems. There are two methods to establish the connection. One is SERVER mode(passive open) that is waiting for connection request. The other is CLIENT mode (active open) that sends connection request to a server.



5.2.1.1 SERVER mode



■ Socket Initialization

In order to initialize a socket, set the operation mode and port of the socket, and provide OPEN command to the command register of the socket. Below is the registers related.

Socket n Mode Register (Sn_MR)

Socket n Source Port Register (Sn_PORT)

Socket n Command Register (Sn_CR)

It initializes the socket *n* as TCP,

```
{
START:
    /* sets TCP mode */
    Sn_MR = 0x01;
    /* sets source port number */
    Sn_PORT = source_port;
```

```
    /* sets OPEN command */
    Sn_CR = OPEN;
    if (Sn_SR != SOCK_INIT) Sn_CR = CLOSE; goto START;
}
```

■ LISTEN

Set the LISTEN command to the command register. The related register is below.

Socket n Command Register (Sn_CR)

```
{
    /* listen socket */
    Sn_CR = LISTEN;
    if (Sn_SR != SOCK_LISTEN) Sn_CR = CLOSE; goto START; // check socket status
}
```

■ ESTABLISHED ?

If connection request is received from remote peer (the status of SOCK_SYNRECV), W5100 sends ACK packet and changes to SOCK_ESTABLISHED status. This status can be checked as below.

```
First method :
{
    If (Sn_IR(CON bit) == '1') goto ESTABLISHED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second method :
{
    If (Sn_SR == SOCK_ESTABLISHED) goto ESTABLISHED stage;
}
```

As connection is established, data transmission and receipt can be performed.

■ ESTABLISHED : Received Data ?

Check as below to know if data is received from remote peer or not.

```
First method :
{
    If (Sn_IR(RECV bit) == '1') goto Receiving Process stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second Method :
{
    if (Sn_RX_RSR != 0x0000) goto Receiving Process stage;
}
```

■ ESTABLISHED : Receiving Process

Received data can be processed as below.

```
{
    /* first, get the received size */
    get_size = Sn_RX_RSR;
    /* calculate offset address */
    get_offset = Sn_RX_RD & gSn_RX_MASK;
    /* calculate start address(physical address) */
    get_start_address = gSn_RX_BASE + get_offset;

    /* if overflow socket RX memory */
    if ( (get_offset + get_size) > (gSn_RX_MASK + 1) )
    {
        /* copy upper_size bytes of get_start_address to destination_addr */
        upper_size = (gSn_RX_MASK + 1) - get_offset;
        memcpy(get_start_address, destination_addr, upper_size);
        /* update destination_addr*/
        destination_addr += upper_size;
        /* copy left_size bytes of gSn_RX_BASE to destination_addr */
        left_size = get_size - upper_size;
```

```
        memcpy(gSn_RX_BASE, destination_addr, left_size);
    }
    else
    {
        /* copy get_size bytes of get_start_address to destination_addr */
        memcpy(get_start_address, destination_addr, get_size);
    }
    /* increase Sn_RX_RD as length of get_size */
    Sn_RX_RD += get_size;
    /* set RECV command */
    Sn_CR = RECV;
}
```

■ ESTABLISHED : Send DATA ? / Sending Process

The sending procedure is as below.

```
{
    /* first, get the free TX memory size */
FREESIZE:
    get_free_size = Sn_TX_FSR;
    if (get_free_size < send_size) goto FREESIZE;

    /* calculate offset address */
    get_offset = Sn_TX_WR & gSn_TX_MASK;
    /* calculate start address(physical address) */
    get_start_address = gSn_TX_BASE + get_offset;

    /* if overflow socket TX memory */
    if ( (get_offset + send_size) > (gSn_TX_MASK + 1) )
    {
        /* copy upper_size bytes of source_addr to get_start_address */
        upper_size = (gSn_TX_MASK + 1) - get_offset;
        memcpy(source_addr, get_start_address, upper_size);
        /* update source_addr*/
        source_addr += upper_size;
        /* copy left_size bytes of source_addr to gSn_TX_BASE */
        left_size = send_size - upper_size;
```

```
        memcpy(source_addr, gSn_TX_BASE, left_size);
    }
    else
    {
        /* copy send_size bytes of source_addr to get_start_address */
        memcpy(source_addr, get_start_address, send_size);
    }
    /* increase Sn_TX_WR as length of send_size */
    Sn_TX_WR += send_size;
    /* set SEND command */
    Sn_CR = SEND;
}
```

■ ESTABLISHED : Received FIN?

Waiting for a connection termination request from remote peer.

It can be checked as below if it received connection termination request of remote peer.

```
First method :
{
    If (Sn_IR(DISCON bit) == '1') goto CLOSED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second method :
{
    If (Sn_SR == SOCK_CLOSE_WAIT) goto CLOSED stage;
}
```

■ ESTABLISHED : Disconnect ? / Disconnecting Process

Check if user requests to terminate this connection.

To terminate the connection, proceed as below,

```
{
    /* set DISCON command */
    Sn_CR = DISCON;
}
```

■ ESTABLISHED : CLOSED ?

No connection state at all. It can be checked as below,

```
First method :
{
    If (Sn_IR(DISCON bit) == '1') goto CLOSED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second method :
{
    If (Sn_SR == SOCK_CLOSED) goto CLOSED stage;
}
```

■ ESTABLISHED : Timeout

In case that connection is closed due to the error of remote peer during data receiving or connection closing process, data transmission can not be normally processed. At this time Timeout occurs after some time.

```
First method :
{
    If (Sn_IR(TIMEOUT bit) == '1') goto CLOSED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second method :
{
    If (Sn_SR == SOCK_CLOSED) goto CLOSED stage;
}
```

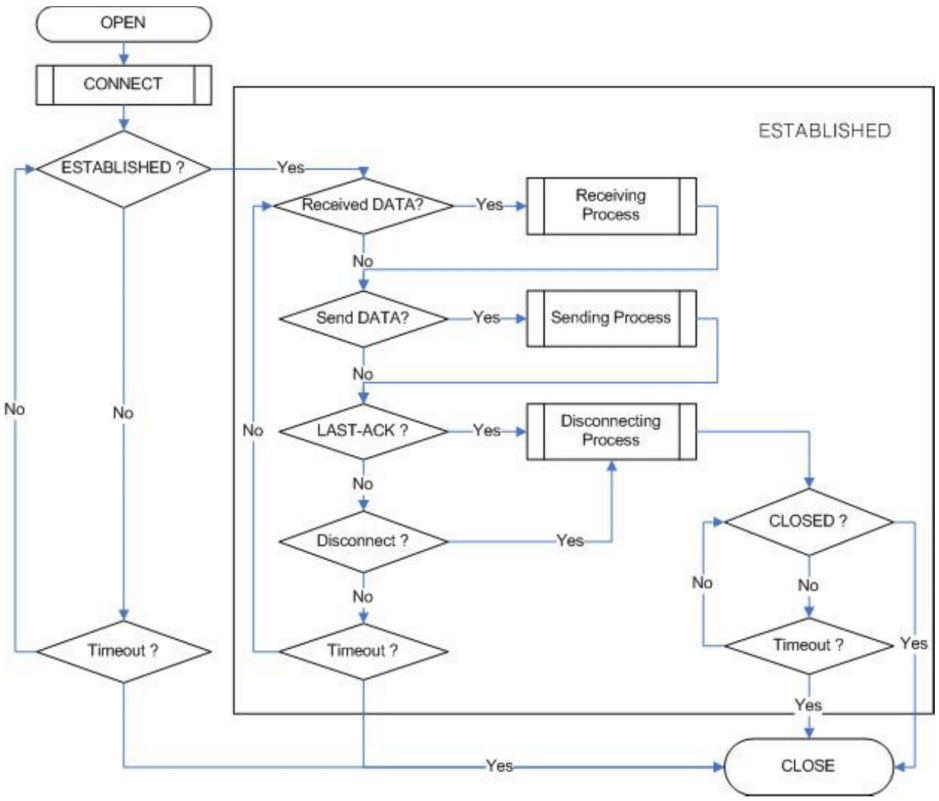
■ Socket Close

This process should be processed in case that connection is closed after data exchange, socket should be closed with Timeout occurrence, or forcible disconnection is necessary due to abnormal operation.

```
{
    /* set CLOSE command */
    Sn_CR = CLOSE;
}
```

5.2.1.2 CLIENT mode

Whole process is shown as below.



■ Socket Initialization

Refer to “5.2.1.1 SERVER mode” (The operation is same as SERVER).

■ CONNECT

Send connection request to remote HOST(SERVER) is as below.

```
{
    /* Write the value of server_ip, server_port to the Socket n Destination IP Address
       Register(Sn_DIPR), Socket n Destination Port Register(Sn_DPORT). */
    Sn_DIPR = server_ip;
    Sn_DPORT = server_port;
    /* set CONNECT command */
```

```
    Sn_CR = CONNECT;
}
```

■ ESTABLISHED ?

The connection is established. It can be checked as below,

```
First method :
{
    If (Sn_IR(CON bit) == '1') goto ESTABLISHED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

```
Second method :
{
    If (Sn_SR == SOCK_ESTABLISHED) goto ESTABLISHED stage;
}
```

■ Timeout

Socket is closed as Timeout occurs as there is not response from remote peer. It can be checked as below.

```
First method :
{
    If (Sn_IR(TIMEOUT bit) == '1') goto CLOSED stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

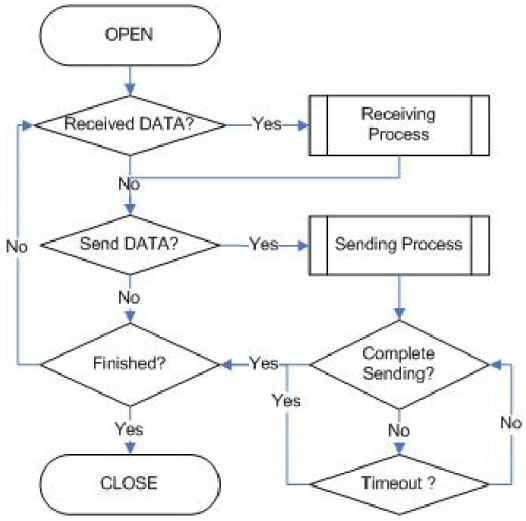
```
Second method :
{
    If (Sn_SR == SOCK_CLOSED) goto CLOSED stage;
}
```

■ ESTABLISHED

Refer to "5.2.1.1. SERVER mode" (The operation is same as SERVER mode)

5.2.2 UDP

UDP provides unreliable and connectionless datagram transmission structure. It processes data without connection establishment. Therefore, UDP message can be lost, overlapped or reversed. As packets can arrive faster, recipient can not process all of them. In this case, user application should guarantee the reliability of data transmission. UDP transmission can be processed as below,



■ Socket Initialization

Initialize the socket *n* as UDP.

```
{
START:
    /* sets UDP mode */
    Sn_MR = 0x02;
    /* sets source port number */
     /* ※ The value of Source Port can be appropriately delivered when remote HOST knows it. */
    Sn_PORT = source_port;
    /* sets OPEN command */
    Sn_CR = OPEN;
    /* Check if the value of Socket n Status Register(Sn_SR) is SOCK_UDP. */
    if (Sn_SR != SOCK_UDP) Sn_CR = CLOSE; goto START;
}
```

■ Received DATA?

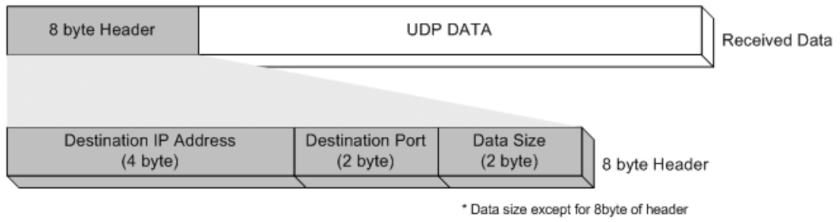
It can be checked as below if data is received from remote peer.

```
First method :
{
    if (Sn_RX_RSR != 0x0000) goto Receiving Process stage;
}
```

```
Second Method :
{
    If (Sn_IR(RECV bit) == '1') goto Receiving Process stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

■ Receiving Process

Received data can be processed as below. In case of UDP, 8byte header is attached to receiving data. The structure of the header is as below.




```
{
    /* first, get the received size */
    get_size = Sn_RX_RSR;
    /* calculate offset address */
    get_offset = Sn_RX_RD & gSn_RX_MASK;
    /* calculate start address(physical address) */
    get_start_address = gSn_RX_BASE + get_offset;

    /* read head information (8 bytes) */
    header_size = 8;
```

```
/* if overflow socket RX memory */
if ( (get_offset + header_size) > (gSn_RX_MASK + 1) )
{
    /* copy upper_size bytes of get_start_address to header_addr */
    upper_size = (gSn_RX_MASK + 1) - get_offset;
    memcpy(get_start_address, header_addr, upper_size);
    /* update header_addr*/
    header_addr += upper_size;
    /* copy left_size bytes of gSn_RX_BASE to header_addr */
    left_size = header_size - upper_size;
    memcpy(gSn_RX_BASE, header_addr, left_size);
    /* update get_offset */
    get_offset = left_size;
}
else
{
    /* copy header_size bytes of get_start_address to header_addr */
    memcpy(get_start_address, header_addr, header_size);
    /* update get_offset */
    get_offset += header_size;
}
/* update get_start_address */
get_start_address = gSn_RX_BASE + get_offset;

/* save remote peer information & received data size */
peer_ip = header[0 to 3];
peer_port = header[4 to 5];
get_size = header[6 to 7];

/* if overflow socket RX memory */
if ( (get_offset + get_size) > (gSn_RX_MASK + 1) )
{
    /* copy upper_size bytes of get_start_address to destination_addr */
    upper_size = (gSn_RX_MASK + 1) - get_offset;
    memcpy(get_start_address, destination_addr, upper_size);
    /* update destination_addr*/
    destination_addr += upper_size;
```

```
        /* copy left_size bytes of gSn_RX_BASE to destination_addr */
        left_size = get_size - upper_size;
        memcpy(gSn_RX_BASE, destination_addr, left_size);
    }
    else
    {
        /* copy get_size bytes of get_start_address to destination_addr */
        memcpy(get_start_address, destination_addr, get_size);
    }
    /* increase Sn_RX_RD as length of get_size+header_size */
    Sn_RX_RD = Sn_RX_RD + get_size + header_size;
    /* set RECV command */
    Sn_CR = RECV;
}
```

■ Send Data? / Sending Process

Data transmission process is as below.

```
{
    /* first, get the free TX memory size */
FREESIZE:
    get_free_size = Sn_TX_FSR;
    if (get_free_size < send_size) goto FREESIZE;

    /* Write the value of remote_ip, remote_port to the Socket n Destination IP Address
      Register(Sn_DIPR), Socket n Destination Port Register(Sn_DPORT). */
    Sn_DIPR = remote_ip;
    Sn_DPORT = remote_port;

    /* calculate offset address */
    get_offset = Sn_TX_WR & gSn_TX_MASK;
    /* calculate start address(physical address) */
    get_start_address = gSn_TX_BASE + get_offset;

    /* if overflow socket TX memory */
    if ( (get_offset + send_size) > (gSn_TX_MASK + 1) )
    {
```

```
        /* copy upper_size bytes of source_addr to get_start_address */
        upper_size = (gSn_TX_MASK + 1) - get_offset;
        memcpy(source_addr, get_start_address, upper_size);
        /* update source_addr*/
        source_addr += upper_size;
        /* copy left_size bytes of source_addr to gSn_TX_BASE */
        left_size = send_size - upper_size;
        memcpy(source_addr, gSn_TX_BASE, left_size);
    }
    else
    {
        /* copy send_size bytes of source_addr to get_start_address */
        memcpy(source_addr, get_start_address, send_size);
    }
    /* increase Sn_TX_WR as length of send_size */
    Sn_TX_WR += send_size;
    /* set SEND command */
    Sn_CR = SEND;
}
```

■ Complete Sending?

The sending completion should be checked after SEND command.

```
{
    If (Sn_CR == 0x00) transmission is completed.
}
```

■ Timeout

Timeout occurs if remote peer does not exist or data transmission is not normally processed. It can be checked as below.

```
{
    If (Sn_IR(TIMEOUT bit) == '1') goto next stage;
    /* In this case, if the interrupt of Socket n is activated, interrupt occurs. Refer to
       Interrupt Register(IR), Interrupt Mask Register (IMR) and Socket n Interrupt Register
       (Sn_IR). */
}
```

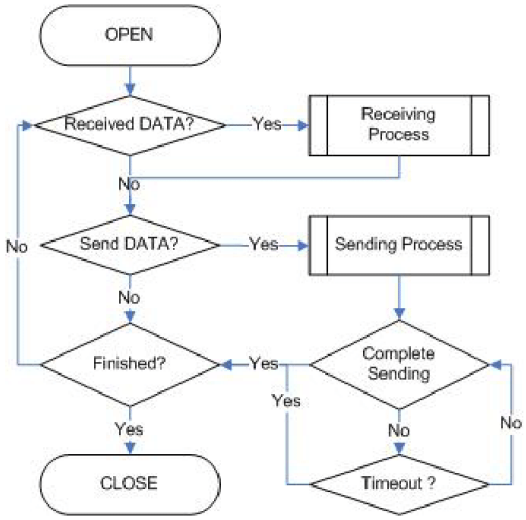
■ Finished? / Socket Close

If all the actions are finished, close the socket.

```
{
    /* set CLOSE command */
    Sn_CR = CLOSE;
}
```

5.2.3 IP raw

IP Raw mode can be utilized if transport layer protocol of some ICMP or IGMP that W5100 does not support, needs to be processed.



- Socket Initialization

It initializes the socket as IP raw.

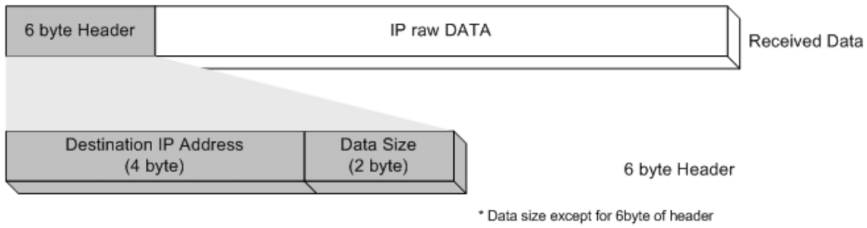
```
{
START:
    /* sets IP raw mode */
    Sn_MR = 0x03;
    /* sets Protocol value */
    /* The value of Protocol is used in Protocol Field of IP Header.
     For the list of protocol identification number of upper classification, refer to on line
     documents of IANA (http://www.iana.org/assignments/protocol-numbers). */
    Sn_PROTO = protocol_value;
    /* sets OPEN command */
    Sn_CR = OPEN;
    /* Check if the value of Socket n Status Register(Sn_SR) is SOCK_IPRAW. */
    if (Sn_SR != SOCK_IPRAW) Sn_CR = CLOSE; goto START;
}
```

■ Received DATA?

It is same as UDP. Refer to “5.2.2 UDP”.

■ Receiving Process

This is same as UDP. Refer to “5.2.2 UDP” except the header information and header size. In case of IP raw, 6byte header is attached to the data received. The header structure is as below.



■ Send DATA? / Sending Process

This is same as UDP. Refer to “5.2.2 UDP” except that remote_port information is not needed.

■ Complete Sending

■ Timeout

■ Finished? / Socket Closed

Next actions are same as UDP. Refer to “5.2.2 UDP”.

5.2.4 MAC raw

MAC Raw mode(only supported in socket 0) can be utilized.

■ Socket Initialization

It initializes the socket as MAC raw.

```
{
START:
    /* sets MAC raw mode */
```

```
    Sn_MR = 0x04;
    /* sets OPEN command */
    Sn_CR = OPEN;
    /* Check if the value of Socket n Status Register(Sn_SR) is SOCK_MACRAW. */
    if (Sn_SR != SOCK_MACRAW) Sn_CR = CLOSE; goto START;
}
```

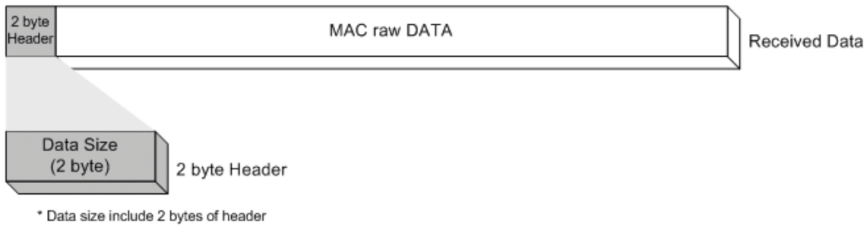
■ Received DATA?

This is same as UDP. Refer to “5.2.2 UDP”.

■ Receiving Process

MAC raw received Ethernet packet having packet size information.

In case of MAC raw, 2byte header is attached to the data received. The header structure is as below.



■ Send DATA? / Sending Process

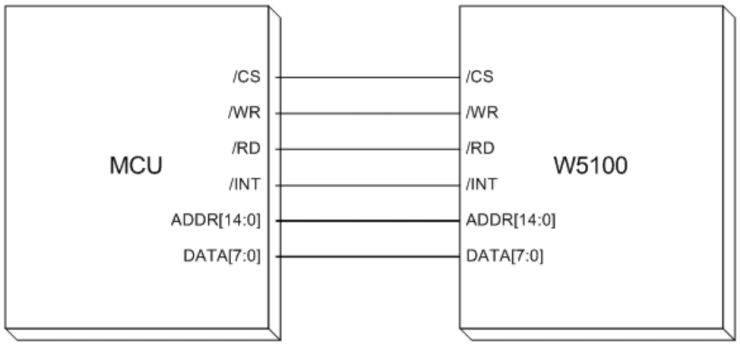
This is same as UDP. Refer to “5.2.2 UDP” except that remote_port information is not needed.

6. Application Information

For the communication with MCU, W5100 provides Direct, Indirect Bus I/F, and SPI I/F modes.
For the communication with Ethernet PHY, MII is used.

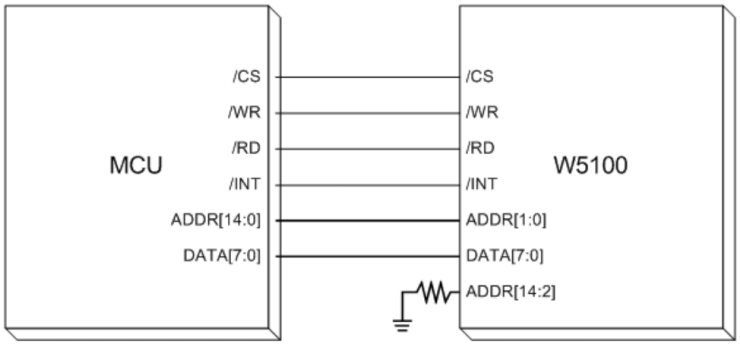
6.1 Direct Bus Interface mode

Direct Bus I/F mode uses 15bit address line and 8bit data line, /CS, /RD, /WR, /INT.



6.2 Indirect Bus Interface mode

Indirect Bus I/F mode uses 2bit address line and 8bit data line, /CS, /RD, /WR, /INT.
[14:2], other address lines should process Pull-down.



Indirect bus I/F mode related register is as below.

Value	Symbol	Description
0x00	MR	It performs the selection of Indirect bus I/F mode, address automatic increase. Refer to “4. Register Description” for more detail.
0x01 0x02	IDM_AR0 IDM_AR1	Indirect bus I/F mode address Register Big-endian use only · In case of Big-endian ordering 0x01 0x02 IDM_AR0 : MSB | IDM_AR1 : LSB Ex) In case of reading S0_CR(0x0401), 0x01(IDM_AR0) 0x02(IDM_AR1) 0x04 | 0x01
0x03	IDM_DR	Indirect bus I/F mode data Register

In order to read or write the internal register or internal TX/RX Memory,

1. Write the address to read or write on IDM_AR0, 1.
2. Read or Write IDM_DR.

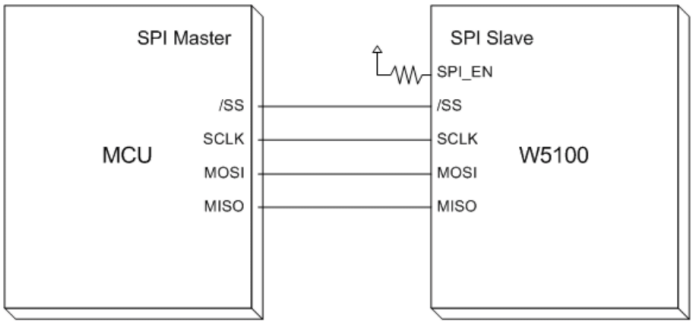
In order to read or write the data on the sequential address, set AI bit of MR(Mode Register). With this, user performs above 1 only one time. Whenever read or write IDM_DR, IDM_AR , the value is automatically increased by 1. Therefore, the value can be processed on the sequential address just by continuous reading or writing of IDM_DR.

6.3 SPI (Serial Peripheral Interface) mode

Serial Peripheral Interface Mode uses only four pins for data communication.

Four pins are SCLK, /SS, MOSI, MISO.

At the W5100, SPI_EN pin is used for SPI operation.



6.3.1 Device Operations

The W5100 is controlled by a set of instruction that is sent from a host controller, commonly referred to as the SPI Master. The SPI Master communicates with W5100 via the SPI bus which is composed of four signal lines: Slave Select(/SS), Serial Clock(SCLK), MOSI(Master Out Slave In), MISO(Master In Slave Out).

The SPI protocol defines four modes for its operation (Mode 0, 1, 2, 3). Each mode differs according to the SCLK polarity and phase - how the polarity and phase control the flow of data on the SPI bus.

The W5100 operates as SPI Slave device and supports the most common modes - SPI Mode 0 and 3.

The only difference between SPI Mode 0 and 3 is the polarity of the SCLK signal at the inactive state. With SPI Mode 0 and 3, data is always latched in on the rising edge of SCLK and always output on the falling edge of SCLK.

6.3.2 Commands

According to SPI protocol, there are only two data lines used between SPI devices. So, it is necessary to define OP-Code. W5100 uses two types of OP-Code - Read OP-Code and Write OP-Code. Except for those two OP-Codes, W5100 will be ignored and no operation will be started.

In SPI Mode, W5100 operates in “unit of 32-bit stream”.

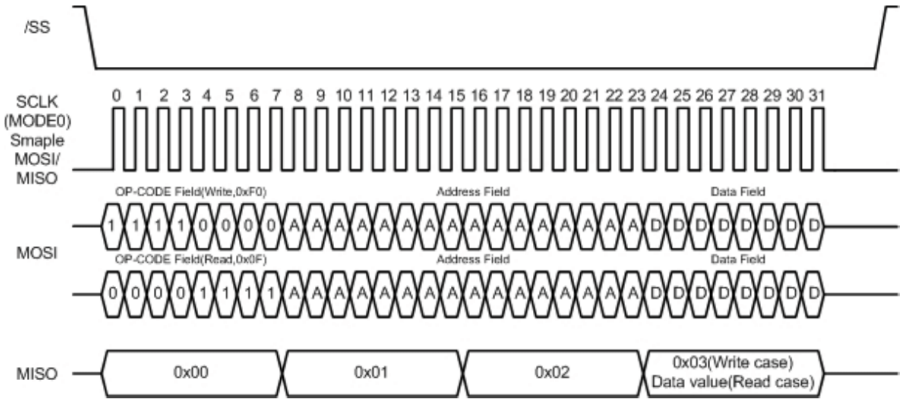
The unit of 32-bit stream is composed of 1 byte OP-Code Field, 2 bytes Address Field and 1 byte data Field.

OP-Code, Address and data bytes are transferred with the most significant bit(MSB) first and least significant bit(LSB) last. In other words, the first bit of SPI data is MSB of OP-Code Field and the last bit of SPI data is LSB of Data-Field. W5100 SPI data format is as below.

Command	OP-Code Field		Address Field	Data Field
Write operation	0xF0	1111 0000	2 bytes	1 byte
Read operation	0x0F	0000 1111	2 bytes	1 byte

6.3.3 Process of using general SPI Master device (According to SPI protocol)

1. Configure Input/Output direction on SPI Master device pins.
    * /SS (Slave Select) : Output pin
    * SCLK (Serial Clock) : Output pin
    * MOSI (Master Out Slave In) : Output pin
    * MISO (Master In Slave Out) : Input pin
2. Configure /SS as 'High'
3. Configure the registers on SPI Master device.
    * SPI Enable bit on SPCR register (SPI Control Register)
    * Master/Slave select bit on SPCR register
    * SPI Mode bit on SPCR register
    * SPI data rate bit on SPCR register and SPSR register (SPI State Register)
4. Write desired value for transmission on SPDR register (SPI Data Register).
5. Configure /SS as 'Low' (data transfer start)
6. Wait for reception complete
7. If all data transmission ends, configure /SS as 'High'



7. Electrical Specifications

Absolute Maximum Ratings

Symbol	Parameter	Rating	Unit
V_{DD}	DC Supply voltage	-0.5 to 3.6	V
V_{IN}	DC input voltage	-0.5 to 5.5 (5V tolerant)	V
I_{IN}	DC input current	±5	mA
T_{OP}	Operating temperature	-40 to 85	°C
T_{STG}	Storage temperature	-55 to 125	°C

***COMMENT**: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage.

DC Characteristics

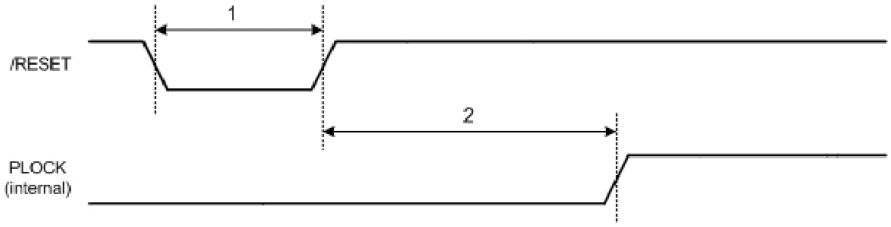
Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
V_{DD}	DC Supply voltage	Junction temperature is from -55°C to 125°C	3.0		3.6	V
V_{IH}	High level input voltage		2.0		5.5	V
V_{IL}	Low level input voltage		- 0.5		0.8	V
V_{OH}	High level output voltage	I_{OH} = 2 ~ 16 mA	2.4			V
V_{OL}	Low level output voltage	I_{OL} = -2 ~ -16 mA			0.4	V
I_I	Input Current	$V_{IN} = V_{DD}$			±5	μA

POWER DISSIPATION

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
P_{10Base}	Power consumption in 10BaseT	Vcc 3.3V Temperature 25°C	-	138	183	mA
$P_{100Base}$	Power consumption in 100BaseT	Vcc 3.3V Temperature 25°C	-	146	183	mA

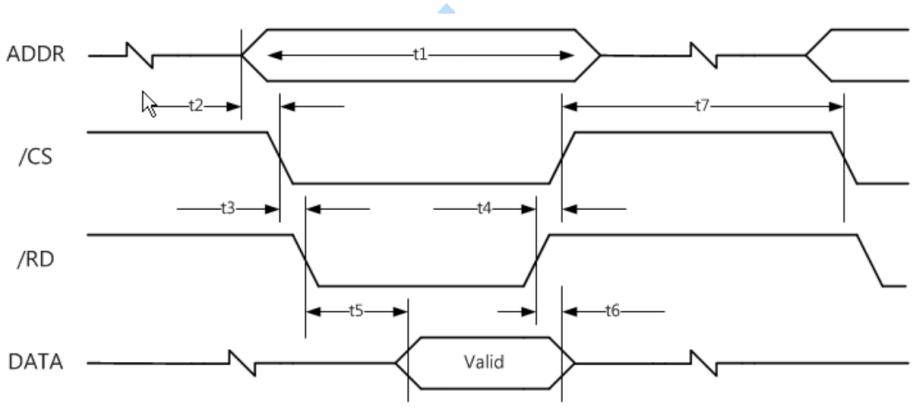
AC Characteristics

Reset Timing



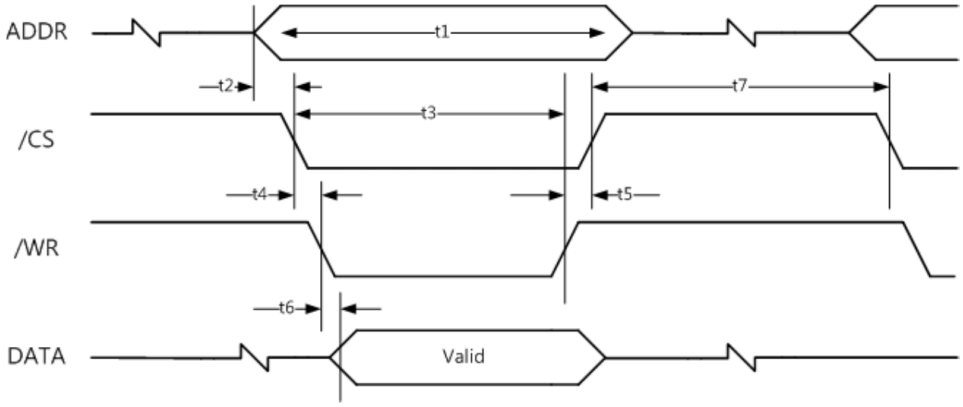
	Description	Min	Max
1	Reset Cycle Time	2 us	-
2	/RESET to internal PLOCK	-	10 ms

Register/Memory READ Timing



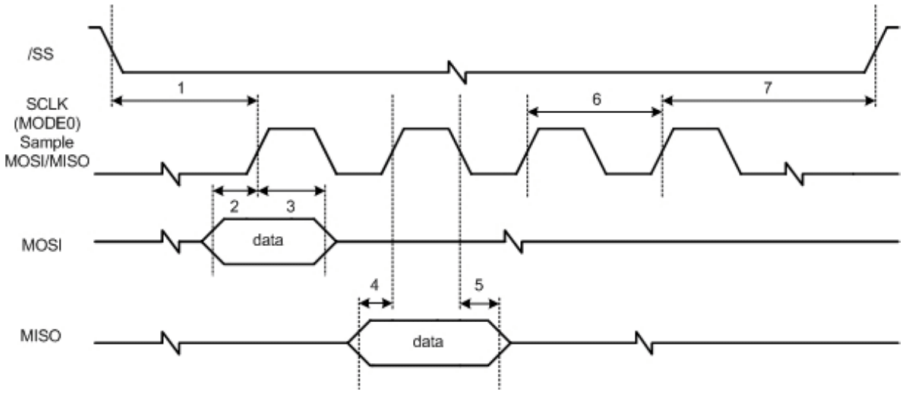
Symbol	Description	Min	Max
t1	Read Cycle Time	80 ns	-
t2	Valid Address to /CS low time	8 ns	-
t3	/CS low to /RD low time	0 ns	
t4	/RD high to /CS high time	0 ns	-
t5	/RD low to Valid data output time	48 ns	-
t6	/RD high to Data High-Z Output time	-	1 ns
t7	/CS high to next /CS low time	32 ns	-

Register/Memory WRITE Timing



Symbol	Description	Min	Max
t1	Write Cycle Time	70 ns	-
t2	Valid Address to /CS low time	7 ns	-
t3	/CS low to /WR high time	70 ns	
t4	/CS low to /WR low time	0 ns	-
t5	/WR high to /CS high time	0 ns	-
t6	/WR low to Valid data time	-	7 ns
t7	/CS high to next /CS low time	32 ns	-

SPI Timing



	Description	Mode	Min	Max
1	/SS low to SCLK high	Slave	21 ns	-
2	Input setup time	Slave	7 ns	-
3	Input hold time	Slave	28 ns	-
4	Output setup time	Slave	7 ns	14 ns
5	Output hold time	Slave	21 ns	-
6	SCLK time	Slave	70 ns	
7	SCLK high to /SS high	Slave	21ns	

Crystal Characteristics

Parameter	Range
Frequency	25 MHz
Frequency Tolerance (at 25°C)	±30 ppm
Shunt Capacitance	7pF Max
Drive Level	100uW
Load Capacitance	27pF
Aging (at 25°C)	±3ppm / year Max

Transformer Characteristics

Parameter	Transmit End	Receive End
Turn Ratio	1:1	1:1
Inductance	350 uH	350 uH

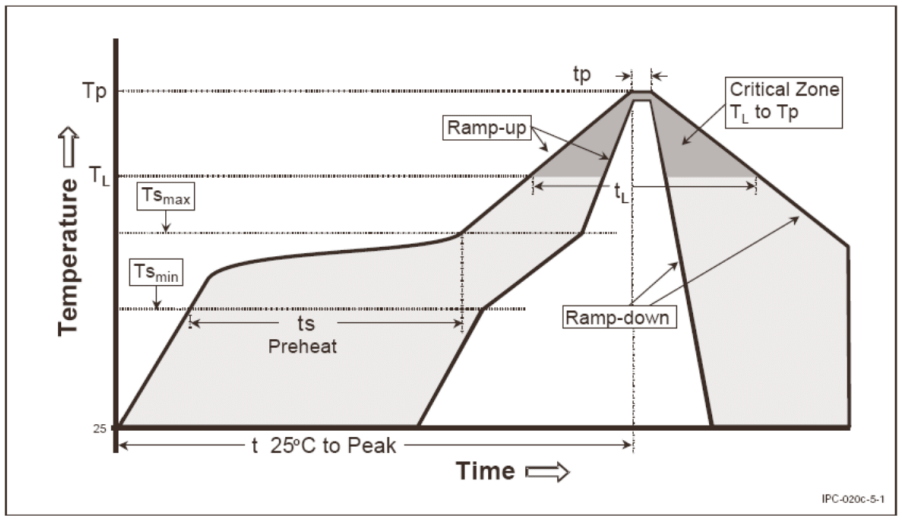
Symmetrical TX & RX channels for auto MDI/MDIX capability

8. IR Reflow Temperature Profile (Lead-Free)

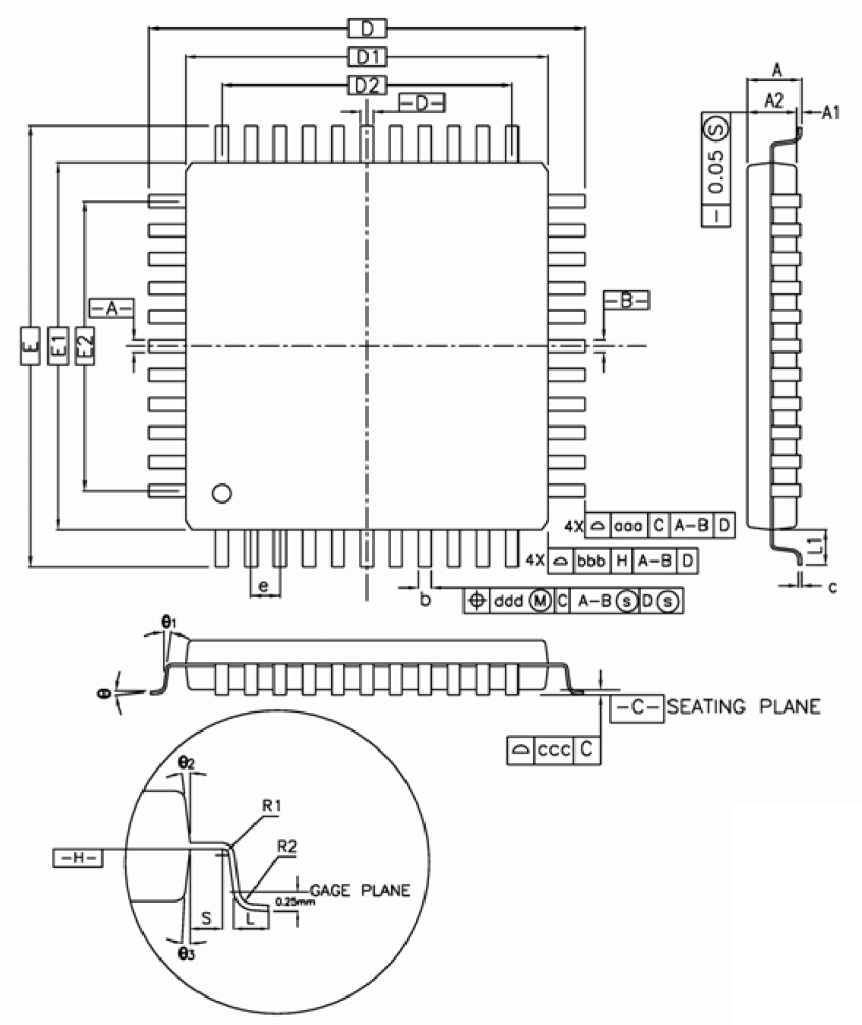
Moisture Sensitivity Level : 3

Dry Pack Required : Yes

Average Ramp-Up Rate (Ts_{max} to Tp)	3° C/second max.
Preheat	
– Temperature Min (Ts_{min})	150 °C
– Temperature Max (Ts_{max})	200 °C
– Time (ts_{min} to ts_{max})	60-180 seconds
Time maintained above:	
– Temperature (TL)	217 °C
– Time (tL)	60-150 seconds
Peak/Classification Temperature (Tp)	260 + 0 °C
Time within 5 °C of actual Peak Temperature (tp)	20-40 seconds
Ramp-Down Rate	6 °C/second max.
Time 25 °C to Peak Temperature	8 minutes max.



9. Package Descriptions



☞ Above diagram shows PIN dimension. All 80 pins are not displayed.

SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	-	-	1.60	-	-	0.063
A1	0.05	-	0.15	0.002	-	0.006

A2	1.35	1.40	1.45	0.053	0.055	0.057
D		12.00 BSC.			0.472 BSC.	
D1		10.00 BSC.			0.393 BSC.	
E		12.00 BSC.			0.472 BSC.	
E1		10.00 BSC.			0.393 BSC.	
R2	0.08	-	0.20	0.003	-	0.008
R1	0.08	-	-	0.003	-	-
θ	0°	3.5°	7°	0°	3.5°	7°
θ_1	0°	-	-	0°	-	-
θ_2	11°	12°	13°	11°	12°	13°
θ_3	11°	12°	13°	11°	12°	13°
c	0.09	-	0.20	0.004	-	0.008
L	0.45	0.60	0.75	0.018	0.024	0.030
L1		1.00 REF			0.039 REF	
S	0.20	-	-	0.008	-	-
b	0.13	0.16	0.23	0.005	0.006	0.009
e		0.40 BSC			0.016 BSC	
D2		7.60			0.299	
E2		7.60			0.299	
aaa		0.20			0.008	
bbb		0.20			0.008	
ccc		0.08			0.003	
ddd		0.07			0.003	

Note :

1. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION.
ALLOWABLE PROTRUSION IS 0.25mm PER SIDE. D1 AND E1 ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.

2. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION.
ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM b DIMENSION BY MORE THAN 0.08mm.
DAMBAR CAN NOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND AN ADJACENT LEAD IS 0.07mm FOR 0.4mm AND 0.5mm PITCH PACKAGES.

6) CC3000

TI SimpleLink™ CC3000 Module – Wi-Fi 802.11b/g Network Processor

FEATURES

- **Wireless network processor**
  - **IEEE 802.11 b/g**
  - **Embedded IPv4 TCP/IP stack**
- **Best-in-class radio performance**
  - **TX power: +18.0 dBm at 11 Mbps, CCK**
  - **RX sensitivity: –88 dBm, 8% PER, 11 Mbps**
- **Works with low MIPS and low-cost MCUs with compact memory footprint**
- **FCC, IC, and CE certified with a chip antenna**
- **HW design files and design guide available from TI**
- **Integrated crystal and power management**
- **Small form factor: 16.3 mm × 13.5 mm × 2 mm**
- **Operating temperature: –20°C to 70°C**
- **Based on TI's seventh generation of proven Wi-Fi solutions**
- **Complete platform solution including user and porting guides, API guide, sample applications, and support community**

APPLICATIONS

- **Home automation**
- **Home security**
- **Connected appliances**
- **Smart energy**
- **M2M communication**

DESCRIPTION

The TI CC3000 module is a self-contained wireless network processor that simplifies the implementation of Internet connectivity (see Figure 1). TI's SimpleLink™ Wi-Fi solution minimizes the software requirements of the host microcontroller (MCU) and is thus the ideal solution for embedded applications using any low-cost and low-power MCU.

The TI CC3000 module reduces development time, lowers manufacturing costs, saves board space, eases certification, and minimizes the RF expertise required. This complete platform solution includes software drivers, sample applications, API guide, user documentation, and a world-class support community.

For more information on TI's wireless platform solutions for Wi-Fi, go to TI's Wireless Connectivity wiki (www.ti.com/connectivitywiki).

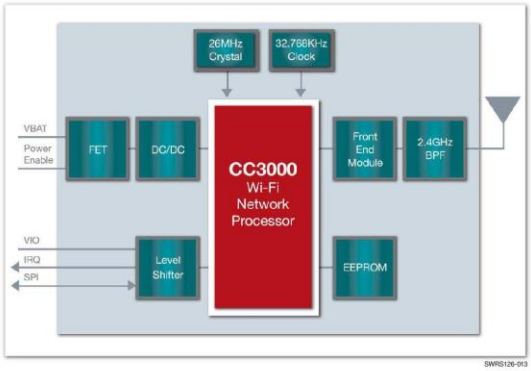


Figure 1. Wi-Fi Solution for TI SimpleLink CC3000 Module

This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

FEATURES

WLAN

- 802.11b/g integrated radio, modem, and MAC supporting WLAN communication as a BSS station with CCK and OFDM rates from 1 to 54 Mbps in the 2.4-GHz ISM band
- Auto-calibrated radio with a single-ended 50-Ω interface enables easy connection to the antenna without requiring expertise in radio circuit design.
- Advanced connection manager with seven user-configurable profiles stored in an NVMEM allows automatic fast connection to an access point without user or host intervention.
- Supports all Wi-Fi security modes for personal networks: WEP, WPA, and WPA2 with on-chip security accelerators
- Smart Config™ WLAN provisioning tools allow customers to connect a headless device to a WLAN network using a smart phone, tablet, or PC.

Network Stack

- Integrated IPv4 TCP/IP stack with BSD socket APIs enables simple internet connectivity with any microcontroller, microprocessor, or ASIC.
- Supports four simultaneous TCP or UDP sockets
- Built-in network protocols: ARP, ICMP, DHCP client, and DNS client enable easy connection to the local network and to the Internet.

Host Interface and Driver

- Interfaces over 4-wire serial peripheral interface (SPI) with any microcontroller, or processor at clock speed up to16 MHz
- Low footprint driver provided for TI MCUs and easily ported to any processor or ASIC
- Simple APIs enable easy integration with any single-threaded or multi-threaded application.

System

- Works from a single, preregulated power supply or connects directly to a battery
- Separated I/O voltage rail allows flexible integration with host processors
- Ultra-low leakage shut-down mode with current <5 µA
- Integrated clock sources

EEPROM

- Integrated EEPROM stores firmware patch, network configuration, and MAC address.
- Programmable through an I2C interface or over APIs from the host, allowing over-the-air firmware upgrades
- Can store 5 KB of user data accessible to the host application, enhancing the MCU NVM

PACKAGE INFORMATION

Module Outline

For the PCB layout of your applications, TI recommends the footprint shown in Figure 2.

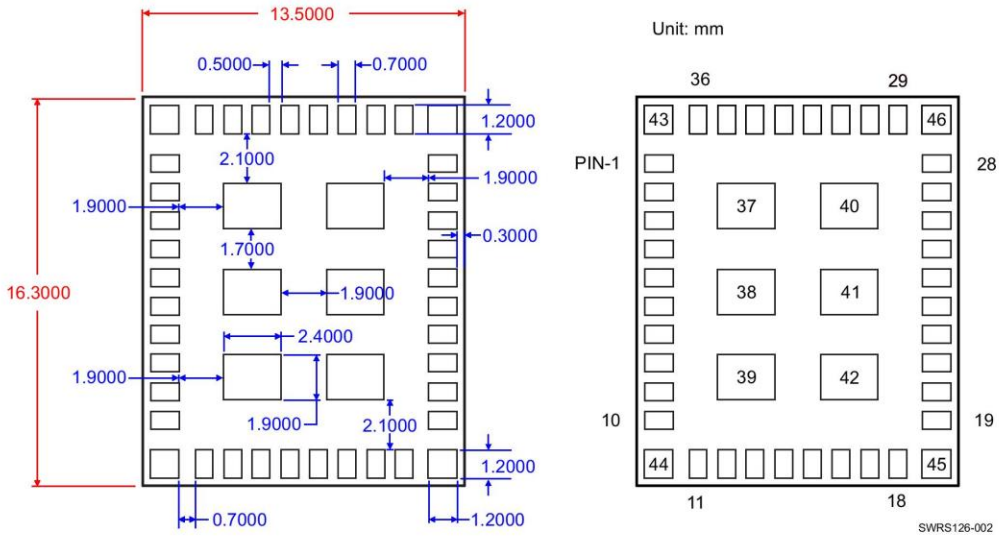


Figure 2. CC3000 Module Footprint and Pinouts

Pin Description

Table 1 describes the CC3000 module pins.

Table 1. CC3000 Module Pins Description

Pin	Signal Name	Type	State at Reset	State After Reset	Voltage Level	Description
1	GND	GND	—	—	—	Ground
2	Reserved_1	—	—	—	1.8 V	Reserved. Connect to test point.
3	NC	—	—	—	—	Not connected
4	Reserved_2	—	—	—	1.8 V	Reserved. Connect to test point.
5	WL_EN2(1)	I	Hi-Z	—	—	Mode setting
6	WL_RS232_TX(2)	O	Hi-Z	Force 1	1.8 V	RS232 test-mode signal (1.8-V logic). Connect to test point. Serial connection for CC3000 radio tool.
7	WL_EN1	I	Hi-Z	—	—	Mode setting
8	WL_RS232_RX(2)	I	Hi-Z	PU	1.8 V	RS232 test-mode signal (1.8-V logic). Connect to test point. Serial connection for CC3000 radio tool.
9	GND	GND	—	—	—	Ground
10	GND	GND	—	—	—	Ground
11	GND	GND	—	—	—	Ground
12	SPI_CS	I	Hi-Z	PU	VIO_HOST	Host interface SPI chip-select (CS)
13	SPI_DOUT	O	Hi-Z	PU	VIO_HOST	Host interface SPI data out
14	SPI_IRQ	O	Hi-Z	Force 1	VIO_HOST	Host interface SPI interrupt

(1) Connect WL_EN1 to WL_EL2 for proper operation of the module.
(2) Leave unconnected in function module.

ESD PERFORMANCE

Because electrostatic discharge (ESD) can damage this integrated circuit, TI recommends handling all integrated circuits (ICs) with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage. ESD damage can range from subtle performance degradation to complete device failure. Precision ICs can be more susceptible to damage because very small parametric changes can cause devices not to meet their published specifications.

Table 2 describes the ESD performance.

Table 2. ESD Performance

HDM(1)	CDM(2)
1000 V	500 V

(1) JEDEC ESD HBM spec JS-001-2012
(2) JEDEC ESD CDM spec 22C101E

MODULE SPECIFICATIONS

Absolute Maximum Ratings

Parameters	Pin	Min	Max	Unit
VBAT_IN	19	–0.5	+6.0	V
VIO_HOST	23	–0.5	+4.6	V
I2C and WL_RS232	27, 28, 29, 30, 6, 8	–0.5	+2.1	V
SPI interface	12, 13, 14, 15, 17	–0.5	+4.6	V
VBAT_SW_EN	26	–0.3	+6.0	V
Storage temperature range	–	–40	+85	°C

Recommended Operating Conditions

Rating	Condition		Sym	Min	Max	Unit
Operating ambient temperature				–20	+70	°C
VBAT_IN				2.7	4.8	V
VIO_HOST supply voltage				1.8	3.6	V
SPI interface						
High-level input voltage	VIO_HOST =	1.8 to 1.95 V	VIH	VIO_HOST x 0.65		V
		1.95 to 2.7 V		1.6		
		2.7 to 3.6 V		2		
Low-level input voltage	VIO_HOST =	1.8 to 1.95 V	VIL		VIO_HOST × 0.35	V
		1.95 to 2.7 V			0.7	
		2.7 to 3.6 V			0.8	
Input voltage			VI	0	3.6	V
Output voltage	Active state		VO	0	VIO_HOST	V
Input transition rise or fall rate			$\Delta t/\Delta v$		5	ns/V
VBAT SW EN						
High-level input voltage			VIH	1.1	5.5	V
Low-level input voltage			VIL	0	0.4	V

Power Consumption

Parameters	Test Conditions	Typ	Max	Unit
802.11b TX current	VBAT = 3.6 V Tamb = +25°C Po = 18 dBm, 11 Mbps L = 2048 bytes tdelay (idle) = 40 µs	260	275	mA
802.11g TX current	VBAT = 3.6 V Tamb = +25°C Po = 14 dBm, 54 Mbps L = 2048 tdelay (idle) = 40 µs	190	207	mA
802.11bg RX current	VBAT = 3.6 V	92	103	mA
Shut-down mode	VBAT = 3.6 V VBAT_SW_EN = 0 V		5	µA

WLAN Transmitter RF Characteristics

(TA = +25°C, VBAT = 3.6 V)

Characteristics	Condition (Mbps)	Min	Typ	Max	Unit
Maximum RMS output power	1		18.3		dBm
	2		18.2		
	11		18.1		
	6		17.0		
	9		17.0		
	18		17.0		
	36		15.5		
	54		14.0		
In-band power variation				±1	
Transmit center frequency accuracy				±20	ppm

Receiver RF Characteristics

(TA = +25°C, VBAT = 3.6 V)

Characteristics	Condition (Mbps)	Min	Typ	Max	Unit
Sensitivity	1 DSSS		–97.5		dBm
	2 DSSS		–95.0		
	11 CCK		–89.0		
	6 OFDM		–91.0		
	9 OFDM		–91.0		
	18 OFDM		–87.0		
	36 OFDM		–81.0		
	54 OFDM		–75.0		
Maximum input level	802.11b			–10	dBm
	802.11g			–20	

SPI HOST CONTROLLER INTERFACE

The SPI is the primary host interface to the CC3000 module.

The SPI interface contains the five-line, master and slave communication model shown in Figure 3.

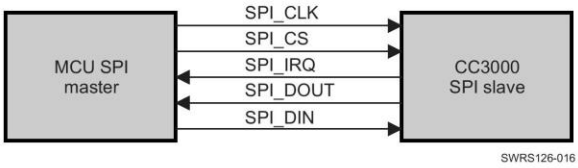


Figure 3. SPI Host Connectivity

Table 3 highlights the CC3000 SPI pin names and functions.

Table 3. SPI Line Description

Pin Name	Description
SPI_CLK	Clock (0 to 16 MHz) from host to slave
SPI_CS(1)	CS (active low) signal from host to slave
SPI_DIN	Data from host to slave
SPI_IRQ(2)	Interrupt from slave to host
SPI_DOUT	Data from slave to host

(1) SPI_CS selects a CC3000 module, indicating that a master wants to communicate to the device.
(2) SPI_IRQ is a dual-purpose slave to the master direction line: in SPI IDLE state while no data transfer is active, driving SPI_IRQ low indicates to the master that the CC3000 module has data to pass to it; driving SPI_IRQ low following SPI_CS deassertion indicates that the CC3000 module is ready to receive data.

SPI Timing

Figure 4 shows the SPI timing sequence.

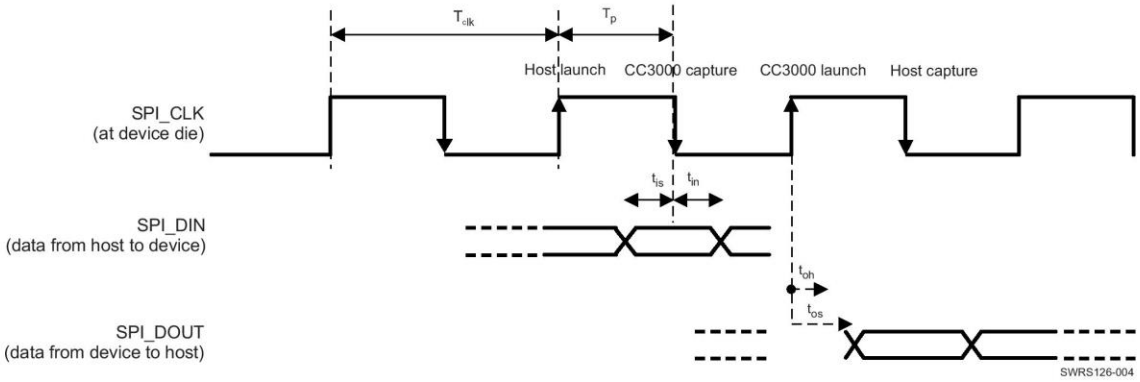


Figure 4. SPI Timing Sequence

Table 4 lists the SPI timing parameters.

Table 4. SPI Timing Parameters

Symbol	Parameter(1)(2)	Min	Max	Unit
T_{clk}	Clock period	62.5		ns
T_p	High pulse width (including jitter and duty cycle)	25(3)	37.5(3)	ns
t_{is}	RX setup time; minimum time in which data is stable before capture edge	5		ns
t_{ih}	RX hold time; minimum time in which data is stable after capture edge	5		
t_{os}	TX setup propagation time; maximum time from launch edge until data is stable		10.2	
t_{oh}	TX hold propagation time; minimum time of data stable after launch edge	3		
C_L	Capacitive load on interface		20	pF

(1) The SPI_CS signal is considered to be asynchronous.
(2) In this example, launch is on the rising edge, and capture is on the falling edge. The opposite scheme can be configured.
(3) 40% to 60% DC (valid for the minimum clock period)

POWER-UP SEQUENCE

Figure 5 demonstrates the wake-up sequence of the CC3000 module.

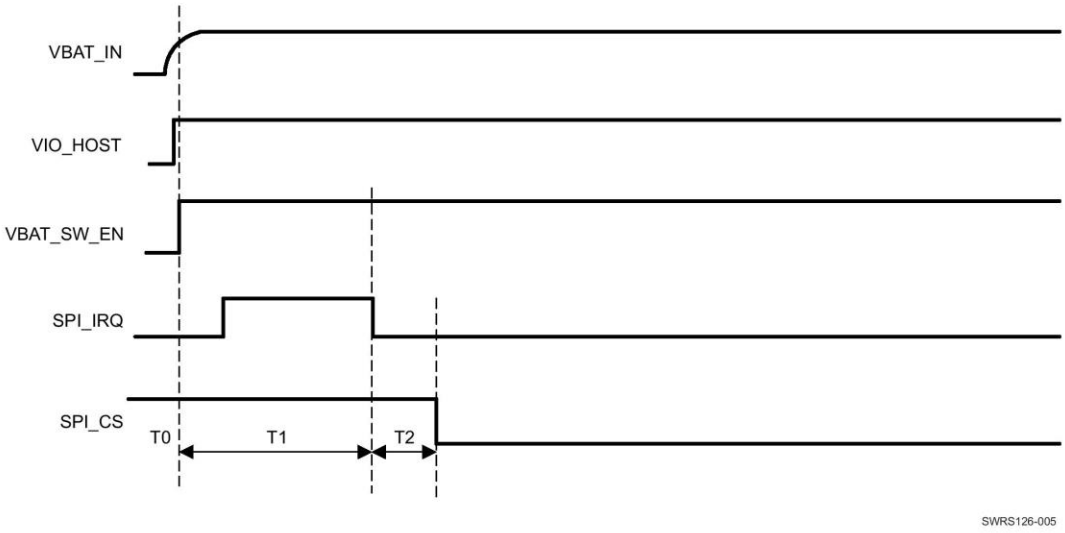


Figure 5. CC3000 Module Power-On Sequences

NOTE

- VBAT_IN and VIO_HOST must be available before VBAT_SW_EN is asserted.
- At wake-up time (T1): The CC3000 module powers up after SPI_IRQ changes state to LOW. T1 is approximately 53 ms.
- At T2: The normal master SPI write sequence is SPI_CS low, followed by SPI_IRQ low (CC3000 host), indicating that the CC3000 core module is ready to accept data. T2 duration is approximately 7 ms.

CC3000 Enable Pins Configuration

Table 5 describes the CC3000 mode of operation based on the enable (EN) pins setting.

Table 5. CC3000 EN Pins Configuration

Mode	State
Test mode (1)	WL_EN1: Leave disconnected. WL_EN2: Connect to ground.
Functional mode (2)	WL_EN1 and WL_EN2 are shorted together.

(1) For CC3000 radio tool operation
(2) For normal operation

cl

Test Mode Serial Interface

The CC3000 module contains a dedicated WLAN serial interface to connect to the CC3000 radio tests tool, an external PC-based software test utility, during development and evaluation phase (see Figure 6 and Table 6). The CC3000 radio test tool utility can be obtained from the CC3000 TI wiki (www.ti.com/connectivitywiki).

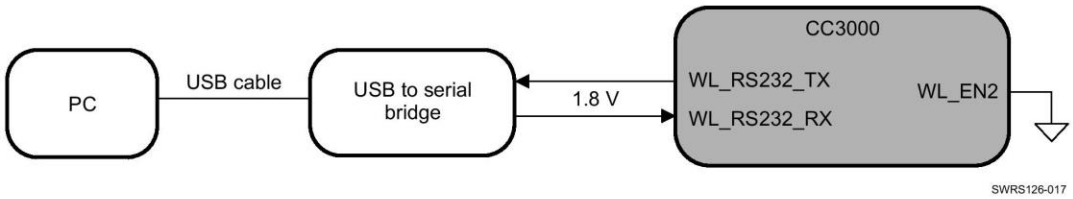


Figure 6. CC3000 Test Mode Serial Interface Connection

Table 6. CC3000 Test Mode Debug Interface Description

Signal Name	Function
WL_RS232_TX	Connection with CC3000 radio PC-based software (1)
WL_RS232_RX	

(1) WL_EN2 pins must be grounded while bringing up the CC3000 radio tool.

SURFACE MOUNT INFORMATION

The CC3000 module uses a flat shield cover designed for a fully automated assembly process. For baking and reflow recommendations, follow MSL level 4 found in the JEDEC/IPC Standard J-STD-20b. The classification temperature (T_C) for the module is 250°C.

MECHANICAL INFORMATION

Module Mechanical Outline

Figure 7 shows the mechanical outline for the CC3000 module.

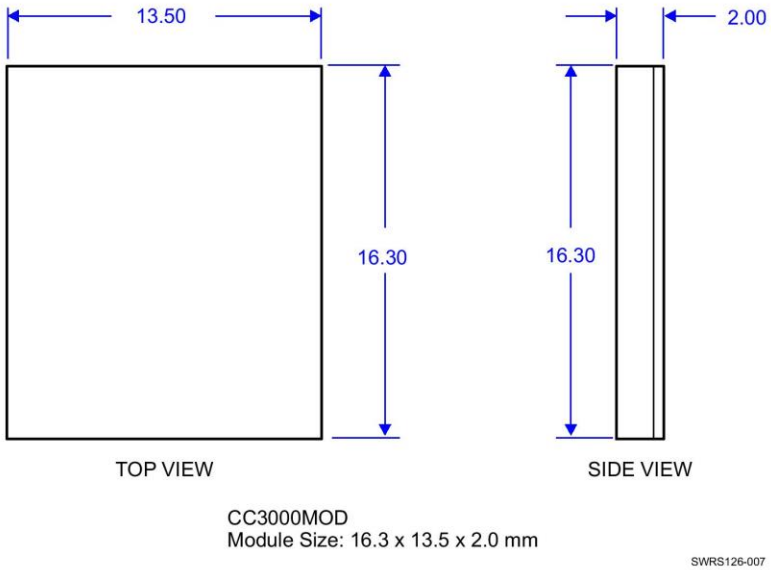


Figure 7. CC3000 Module Mechanical Outline

cli

Package Marking

Figure 8 shows the CC3000 module package marking.

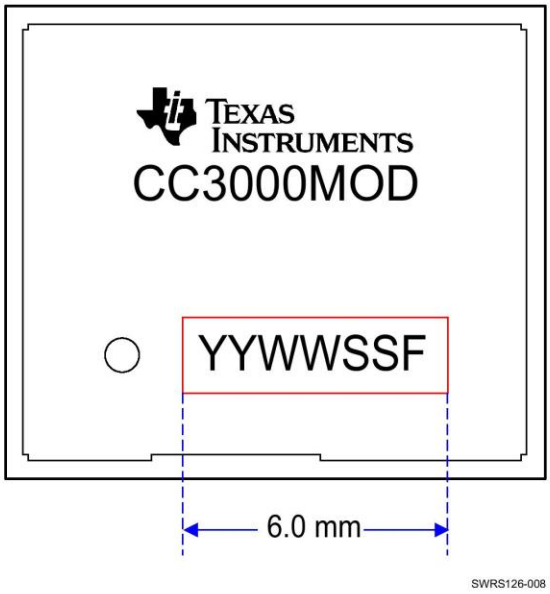


Figure 8. CC3000 Module Package Marking

Table 7 defines the marking code.

Table 7. Package Marking Definitions

Code	Definition
YYWWSSF	Date
YY	Year (for example, 2012 = 12)
WW	Week (01 through 53)
SS	Serial number from 01 to 99 to match manufacturer lot number
F	Reserved for internal use

Ordering Information

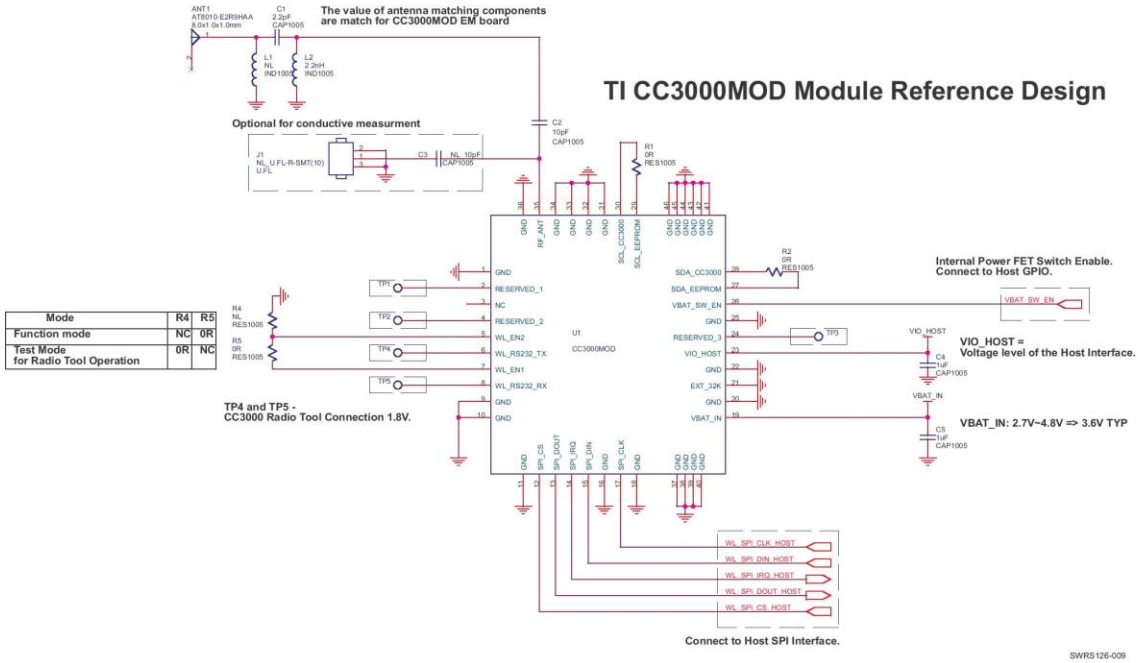
Table 8 lists the CC3000 module part numbers.

Table 8. CC3000 Module Part Numbers

Order Number	Description
CC3000MOD	CC3000 module, 44 modules per tray
CC3000MODR	CC3000 module reel, 1200 modules per reel

REFERENCE SCHEMATICS AND BILL OF MATERIALS

Figure 9 shows the schematics for the CC3000 to host reference design.



Mode	R4	R5
Function mode	NC	0R
Test Mode for Radio Tool Operation	0R	NC

Figure 9. CC3000 Module to Host Reference Design

NOTE

For flexibility, VIO_HOST supports both cases in which the VBAT and VIO voltages of the MCU can be the same or different.

Table 9 lists the bill of materials.

Table 9. Bill of Materials

Part Reference	Description	Manufacturer	Manufacturer PN
ANT1	2.4-GHz chip antenna, 8.0 × 1.0 mm	ACX	AT8010-E2R9HAA
C1	C0402, 2.2 pF	Walsin	0402N2R2C500LT
L2	L0402, 2.2 nH	ACX	HI1005-1C2N2SMT
C2(1)	C0402, 10 pF	Walsin	0402N100J500LT
C4, C5(1)	C0402, 1 µF	Murata	GRM155R60J105KE19D
R1, R2, R5(1)	R0402, 0R	Walsin	WR04X000PTL
J1	RF coaxial U.FL, SMD	Hirose	U.FL-R-SMT-1(10)

(1) Any component with similar values can be used.

DESIGN RECOMMENDATIONS

This section describes the layout recommendations for the CC3000 module, RF trace, and antenna.

Antenna

The ACX ceramic antenna is mounted on the CC3000 EVM board with a specific layout and matching circuit for the radiation test conducted in FCC, CE and IC certifications. Figure 10 shows the location of the antenna on the EVM board as well as the RF trace routing from the CC3000 module.

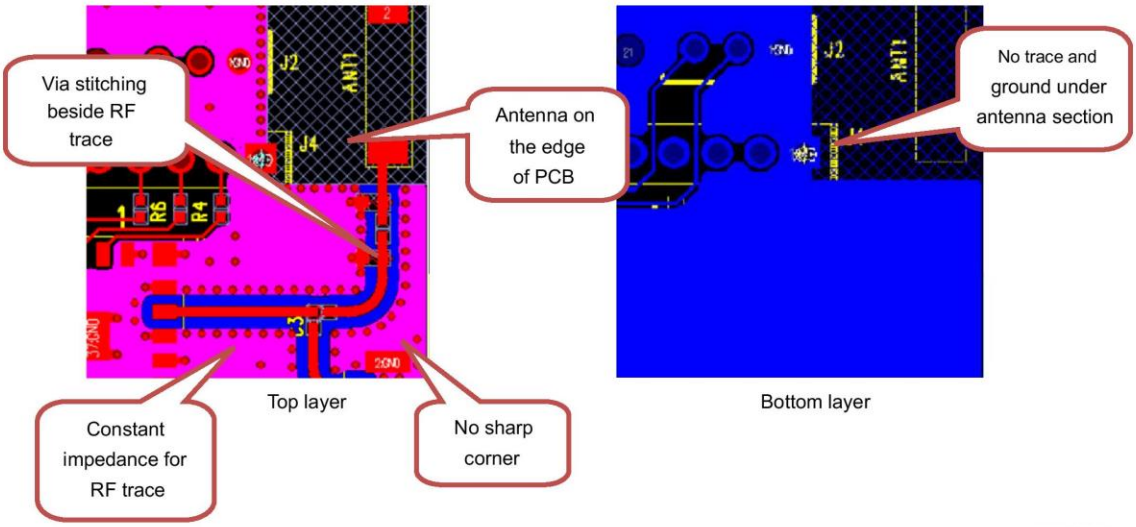


Figure 10. RF Trace and Antenna Design for PCB Layout

Module Layout Recommendations

Observe the following module layout recommendations (see also Figure 11):

- Have a solid ground plane and ground vias under the module for stable system and thermal dissipation.
- Do not run signal traces underneath the module on a layer where the module is mounted.
- Signal traces can be run on a third layer under the solid ground layer and beneath the module mounting layer.

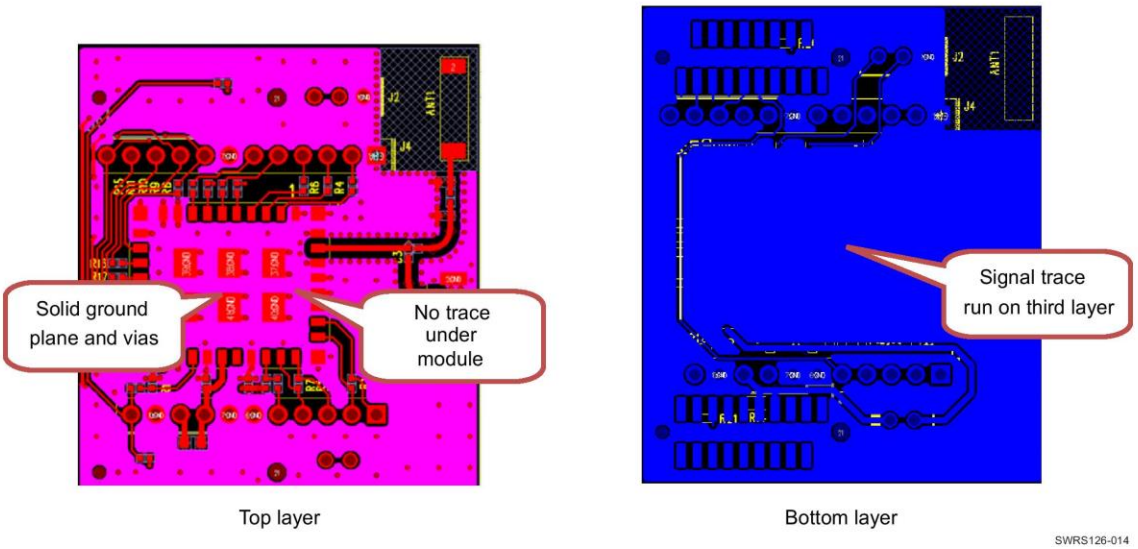


Figure 11. Module Layout

RF Trace and Antenna Layout Recommendations

Observe the following recommendations for RF trace and antenna layout (see also Figure 10):

- RF traces must have 50-Ω impedance (microstrip transmission line).
- RF trace bends must be gradual with a maximum bend of approximately 45 degrees and with trace mitered. RF traces must not have sharp corners.
- There must be no traces or ground under the antenna section.
- RF traces must have via stitching on the ground plane beside the RF trace on both sides.
- RF traces must be as short as possible. The antenna, RF traces, and the module must be on the edge of the PCB product in consideration of the product enclosure material and proximity.

clv

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead/Ball Finish (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
CC3000MOD	NRND			46	84	TBD	Call TI	Call TI	-20 to 70		
CC3000MODR	NRND			46	1200	TBD	Call TI	Call TI	-20 to 70		
CC3000MODT	NRND			46	250	TBD	Call TI	Call TI			
CC3000YFVR	OBSOLETE	DSBGA	YFV	126		Green (RoHS & no Sb/Br)	Call TI	Level-1-260C-UNLIM		CC3000	
CC3000YFVT	OBSOLETE	DSBGA	YFV	126		TBD	Call TI	Call TI		CC3000	
XCC3000MOD	OBSOLETE			46		TBD	Call TI	Call TI	-20 to 70		

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
TBD: The Pb-Free/Green conversion plan has not been defined.
Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead/Ball Finish - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead/Ball Finish values may wrap to two lines if the finish value exceeds the maximum column width.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

B. Resumes

1) Phillip Davis

Phillip Davis

[redacted]

Sacramento, Ca 95834

[redacted]

Pjdavis1988@hotmail.com

OBJECTIVE

An internship in the field of computer engineering. Available for immediate employment.

EDUCATION

In progress: **BS , Computer Engineering• CSU Sacramento•** To be completed May 2015• GPA 3.1

Related Courses

Advanced Logic Design	Network Analysis	Computer Interfacing
Data Structure+Algorithm Analysis	Advanced Computer Organization	Electronics I
Computer Hardware Design	Signals and Systems	CMOS and VLSI
Intro System Program Unix	Physics	Calculus
Computer Network+Inet*	Embedded Proc System Design*	Physical Electronics*
Operating Systems Principals*		*Spring, 2014

SKILLS AND KNOWLEDGE

Programming Languages: C•C++•Verilog• VHDL• x86 Assembly
Operating Systems: Windows 7• Windows XP• Unix• Linux
Software: PSpice• Multisim• Cadence• Microsoft Office• Xilinx ISE
Tools: Oscilloscope• Function Generator• Arduino• Parallax Propeller• Spartan E3 FPGA

ORGANIZATIONAL AND COMMUNICATION SKILLS

- Strong communication skills developed through lab reports and group presentations.
- Strong logic and analytical skills as well as excellent interpersonal skills.
- Loyal, dependable and diligent worker.

PROJECT EXPERIENCE

Microcontroller Projects:
Worked as a part of a 2 man team to use an Arduino microcontroller to power a windshield wiper, car alarm, and a distance monitor. Also worked individually with the Propeller and the Arudino microcontrollers on practice labs which included diodes and motors.

16-bit MIPS processor:
Worked as a two person team to design and implement a 16-bit MIPS processor with a four stage pipeline. We used behavior modeling in Verilog to implement branch prediction, hazard detection, and forwarding as well as load/store and ALU implementation.

Spartan E3 LCD:
Worked individually to create an animation using Xilinx ISE and the LCD of the Spartan E3 board in the Verilog language.

WORK EXPERIENCE

Sales Representative **Kentucky Fried Chicken, Sacramento, CA** 03/10-Present
Open and closing the store as well as interacting with customers, production of food, and training new employees. In control when there is no relief or manager present.

Team Leader **Child Abuse Prevention Council, Sacramento, CA** 11/06-11/07
Through AmeriCorps, assigned to the Child Abuse Prevention Council of Sacramento inc. Worked for the Bridges Afterschool Program at Howe Elementary School. Responsible for 40 4th and 5th graders while leading them through homework and recreational activities.

2) Adam Batakji

Adam Batakji

adambatakji@yahoo.com

OBJECTIVE:

A challenging internship in the field of Computer Engineering/Computer Science, or related engineering discipline

EDUCATION:

In progress: Bachelor of Science, Computer Engineering
CSUS Expected date of graduation: Spring 2015
California State University, Sacramento

RELATED COURSES:

- Advanced Logic Design
- Computer Interfacing
- Computer Hardware Design
- Operating System Principles
- Computer Software Engineering
- Data Structure and Algorithm Analysis
- Database Management
- Computer Network and Internet
- Network Analysis
- Signals and Systems

KNOWLEDGE AND SKILLS

Computer Languages:

C, C++, Python, Verilog, Assembly, Java, SQL, PHP, HTML

Hardware/Software:

Oscilloscope, Xilinx ISE, ModelSim, ORCAD PSpice, Linux, 170E Traffic Controller, Digi Rabbit Series

Communication/Organization:

Outstanding technical writing and presentation skills
Neat and orderly in organization

Leadership:

Involved numerous times in positions where relied upon
Personal trainer, helping people achieve their goals
Pressure only increases motivation

PROJECT EXPERIENCE:

Computer Interfacing

Collaborated in designing an entertainment system named Raspberry 720. Project consisted of Raspberry Pi hardware and other parts along with code and configuration. Project created an excellent learning experience in Linux Operating System.

ECS Problem Report System

Developed through collaborative effort an application that lets users submit a problem report online that gets stored into a database for servicing. Languages such as PHP, SQL and C were used as the tools for development. Received a good handle on database systems as a result.

Advanced Logic Design

Used Verilog program to connect Spartan 3E FPGA board with PS/2 Keyboard to record keystrokes on LCD screen. Participated in role that involved debugging and keyboard verification.

GUI Software Application

Developed address book application using GUI toolkit that allowed user to store contacts into a database and view them in a sorted order. Project enforced expansive knowledge in Java GUI toolkits.

Database Management

Developed banking application using MYSQL in which it stores customer names and other basic information. Project emphasized significance behind organization in applications.

ACTIVITIES AND ACCOMPLISHMENTS:

Coding Problems
Top Sales Associate
Personal Training
Dean's List

3) Kevin Nguyen

Phuc Nguyen

kevin_nguyen_12@hotmail.com

OBJECTIVE:

To obtain an Internship position in the Computer Engineering field.

EDUCATION:

Bachelor of Science, Computer Engineering, CSU Sacramento, GPA 3.46 (In Progress)
Expected Graduation Date: Spring 2015

RELATED COURSES:

Programming Concepts & Methodology I, II
Advanced Logic Design
Advanced Computer Organization
CMOS & VLSI Design
Computer Hardware Design
Embedded Processor System Design
Computer Interfacing
Cryptography *
Operating System Principals / Pragmatics
Signals and Systems
Network Analysis
Systems Programming & Assembly Lang.
Data Structure & Algorithm Development
Calculus I, II, III
Differential Equations & Linear Algebra
Computer System Attacks and Countermeasures *

** In progress as of Fall 2014*

KNOWLEDGE AND SKILLS

Computer Languages:

Java, C, Objective-C, Assembly, Verilog, HTML

Software Applications:

Xilinx ISE, Multisim, Xcode IDE

PROJECT EXPERIENCE:

Microcontroller Project

Built an Arduino-powered Smart Bathroom RFID Access System. A custom GFCI outlet is assembled and connect to a relay to power several appliances in accordance to specific room access levels.

WORK EXPERIENCE:

Shift Leader **Taco Bell** 8/06 to present

Managerial duties (Customer Service, Team Management, Cash Handling)
Technical support (PAR Registers, TACO System, Debit/Credit Terminals, LAN)

ACTIVITIES AND ACCOMPLISHMENTS:

Dean's Honor Roll, CSU Sacramento, Fall 2013
Dean's Honor Roll, Sierra College, Fall 2008

Working 24 hours per week, while carrying 14 units per semester and maintaining a 3.46 GPA

4) Joel Barrera

Joel Barrera

3663 Owens Way (916) 764-7153 jbjoel91@gmail.com

OBJECTIVE: An internship position in the field of Computer Engineering.

EDUCATION: In progress: BS, Computer Engineering, CSU Sacramento, expected date May 2015

RELATED COURSES:

Computer Network and Internet	Advanced Computer Organization *	CMOS and VLSI
Advanced Logic Design	Computer Interfacing	Computer Hardware Design *
Data Structures and Algorithm Analysis	Operating System Principles *	Discrete Structures
Network Analysis		

** In progress as of Spring 2014*

SKILLS:

Programming:
C, C++, Java, Verilog, VHDL, HTML, x86 Assembly

Systems:
Windows XP, Windows 7, Windows 8, Windows Vista, Macintosh, Unix, Linux

Software:
Visual Studio, Eclipse, ModelSim, Multisim, Matlab, Xilinx ISE, PSpice, L-Edit, Microsoft Office

Communication/Organization:
- Excellent verbal and written communication skills
- Strong analytical and problem solving skills
- Bilingual: English/Spanish
- Strong coordinating and planning skills

WORK EXPERIENCE:

Cashier/Crew Member **McDonalds Restaurant** 8/09 to present
- Providing customer service, handling large sums of money, inventory, working in a fast-paced environment.

PROJECT EXPERIENCE:

In progress: Undergraduate Team Space Transportation Design Competition
- Design and business plan for a commercial Earth-to-Orbit (ETO) passenger vehicle. Member of a six-person team that will implement traditional aerospace engineering disciplines such as structures, propulsion, flight mechanics, orbital mechanics, and optimization.

VOLUNTEER WORK:

- MEP's Mentoring Program (2012-Present)
- Volunteer at St. Rose of Lima Parish

PROFESSIONAL ACTIVITIES:

- Member, Society of Hispanic Professional Engineers (SHPE)
- Member, The American Institute of Aeronautics and Astronautics (AIAA)
- President, La Voz Latina, Highlands High School, 2009

C. References

[1] World Future Society, [Online] Available: http://www.wfs.org/blogs/thomas-frey/end-theft (Accessed: 6 September 2014).

[2] The Alantic City Lab, "These 8 Depressing Bike Theft Statistics Show Just How Bad the Problem Is" Available: http://www.citylab.com/commute/2014/04/these-8-depressing-bike-theft-statistics-show-just-how-bad-problem/8890/ (Accessed: 6 September 2014).

[3] Rohit Pathak, Satyadhar Joshi, "Internet, 2009 First Asian Himalays International Conference on,"(Recent Trends in RFID and a Java based Software Framework for its Integration in Mobile Phones) [online] 2009, Available: http://ieeexplore.ieee.org.proxy.lib.csus.edu/stamp/stamp.jsp?tp=&arnumber=5340296 (Accessed: 6 September 2014).

[4] RFID Journal, [Online] Available: http://www.rfidjournal.com/site/faqs#Anchor-How-7638 (Accessed: 6 September 2014).

[5] Luk, Kwai-Man, "Proceedings of the IEEE (Volume: 99, Issue: 12)," (The Importance of the New Developments in Antennas for Wireless Communications), [online] 2011, Available: http://ieeexplore.ieee.org.proxy.lib.csus.edu/stamp/stamp.jsp?tp=&arnumber=6077119 (Accessed: 7 September 2014).

[6] State of California Department of Justice, "Crime Data" Avalilable: http://oag.ca.gov/crime (Accessed: 7 September 2014).

[7] Ichino. M, Yamazaki. Y, "Biometrics and Kansei Engineering (ICBAKE), 2013 International Conference on," (Soft Biometrics and Its Application to Security and Business), [online] 2013, Available: http://ieeexplore.ieee.org.proxy.lib.csus.edu/xpls/icp.jsp?arnumber=6603525#note-ref_1a (Accessed: 8 September 2014).

[8] A. K. Jain, A. Ross, S. Pankanti, "IEEE Transactions on Information Forensics and Security, (Vol. 1, Issue. 2)," (Biometrics: A Tool for Information Security), [online] 2006, Available: http://ieeexplore.ieee.org.proxy.lib.csus.edu/stamp/stamp.jsp?tp=&arnumber=1634356 (Accessed: 8 September 2014).

[9] J. Dong, T. Tan, "Visual Information Engineering, 2008. VIE 2008. 5th International Conference on," (Security Enhancement of Biometrics, Cryptography and Data Hiding by Their Combinations) [online] 2008, Available: http://ieeexplore.ieee.org.proxy.lib.csus.edu/stamp/stamp.jsp?tp=&arnumber=4743423 (Accessed: 8 September 2014).

[10] Morrall, P, P Marshall, S Pattison, and G MacDonald. "Crime and Health: A Preliminary Study into the Effects of Crime on the Mental Health of UK University Students." Journal of Psychiatric & Mental Health Nursing, 17.9, [online] 2010, Available: http://xerxes.calstate.edu/sacramento/ebsco/record?id=rzh-2010809754 (Accessed: 5 September 2014).

[11] The Keyless Lock Store, "Proximity readers and cards for single doors" Available: http://www.nokey.com/proxread.html (Accessed: 15 September 2014).

[12] Earth LCD, "We make LCDs work" Available: http://store.earthlcd.com/ezLCD-301?sc=21&category=297 (Accessed: 13 September 2014).

[13] Panic, "Coda 2" Available: https://www.panic.com/coda/buy.html (Accessed 13 September 2014).

[14] Sequel Pro, "MySQL database management" Available: http://www.sequelpro.com/ (Accessed 13 September 2014).

[15] Arduino, "Arduino store" Available: http://store.arduino.cc/product/A000066 (Accessed 13 September 2014).

[16] "Breaking lock piece of cake, teen to magistrate", Nagpur, TimesContent Newspaper, January 18, 2010, http://timesofindia.indiatimes.com/city/nagpur/Breaking-lock-a-piece-of-cake-teen-to-magistrate/articleshow/5465756.cms (accessed February 1, 2015)

[17] "Digital Lock-Picking: This Simple Arduino Hack Opens Millions of Hotel Keycard Doors", Justin, WonderHowTo.com. Online. http://null-byte.wonderhowto.com/inspiration/digital-lock-picking-simple-arduino-hack-opens-millions-hotel-keycard-doors-0138312/ (accessed February 1, 2015)

[18] "The Evolution of RFID Security", Melanie R. Rieback, Bruno Crispo, and Andrew S. Tanenbaum. Online. http://scholar.googleusercontent.com/scholar?q=cache:lbiXfsPYWw4J:scholar.google.com/+RFID+and+security&hl=en&as_sdt=1,5 (accessed February 1, 2015)

[19] Ballezza, Richard A. "Health Club Credit Card Theft: A National Crime Problem". Online. November 2003. http://leb.fbi.gov/2003-pdfs/leb-november-2003 (accessed January 30, 2015).

[20] "File:Sacramento State Logo.svg." Wikipedia. Web 05 Oct. 2011. <http://en.wikipedia.org/wiki/File:Sacramento_State_Logo.svg>

[21] Schreier, Jason. The Man Who Promised Too Much. Kotaku, 2014. http://kotaku.com/the-man-who-promised-too-much-1537352493

[22] GoKeyless Online Store [Online]. Accessed: 2/20/15. http://www.gokeyless.com/product/1443/schlage-fe210f-computer-managed-locking-system

[23] The Keyless Lock Store [Online]. Accessed: 2/22/15. http://www.nokey.com/elbaprcacalo.html?utm_source=googlepepla&utm_medium=adwords&id=18283950120

[24] Noke-Fuz Designs [Online]. Accessed: 2/28/15. http://fuzdesigns.com/products/Noke

[25] Master Lock [Online]. Accessed: 3/2/15. http://www.masterlock.com/about-us/history

[26] Poverty and Crime [Online]. Accessed: 3/12/2015. http://economics.fundamentalfinance.com/povertycrime.php

[27] 10 Ways To Boost House Value [Online]. Accessed: 3/02/15. http://www.thisoldhouse.com/toh/photos/0,,20247552_20558019,00.html

[28] History of Locks [Online]. Accessed: 3/12/2015. http://www.historyofkeys.com/locks-history/history-of-locks/

[29] An Overview [Online]. Accessed: 3/12/2015. http://www.chubb.co.uk/utcfs/Templates/Pages/Template-54/0,8063,pageId%3D1583%26siteId%3D403,00.html

[30] Wood, David "What is a solenoid. Definitions, Uses & Examples" Study.com http://study.com/academy/lesson/what-is-solenoid-definition-uses-examples.html Accessed: 4/5/15

[31] John "Working of Relays" Circuitstoday.com. http://www.circuitstoday.com/wp-content/uploads/2010/02/Relay-Design.jpg. Accessed: 4/5/15

[32] "Need a Loud Electrical Component for Magnetic Lock" electronics.stackexchange.com. http://i.stack.imgur.com/K5zne.gif Accessed: 4/5/15

[33] "Shanghai Wafer Microelectronics" waferstar.com. http://www.waferstar.com/image/Lock-Solenoid-01.jpg. Accessed 4/5/15

[34] "File:Sacramento State Logo.svg." Wikipedia. Web 05 Oct. 2011. <http://en.wikipedia.org/wiki/File:Sacramento_State_Logo.svg

[35] "Linear Solenoid Actuator" Electronics Tutorial. Web 05 Oct. 2011 <http://www.electronics-tutorials.ws/io/io_6.html>

[36] Bonsor, Kevin, and Wesley Fenlon. "How RFID Works". HowStuffWorks.com. <http://electronics.howstuffworks.com/gadgets/high-tech-gadgets/rfid.htm>.

[37] "What can RFID be used for?"Technovelgy.com. <http://www.technovelgy.com/ct/Technology-Article.asp?ArtNum=4>.

[38] Floyd, Ray. "Passive vs. Semi-Passive vs. Active Tags in RFID". ThomasNet. <http://news.thomasnet.com/imt/2014/03/04/passive-vs-semi-passive-vs-active-tags-in-rfid>.

D. Glossary

Radio Frequency Identification (RFID) *– Method of identifying things using radio frequency communication.*

Societal Problem *– Problem in today's world that is effecting our society negatively.*

Design Idea *– A detailed vision on how to help effect the problem with Team 2's project.*

Work Breakdown Structure *– Decomposition of the work schedule of the project into a graphic view.*

Mobile Lock *– Portable lock that can be carried around and used in places similar to a padlock.*

Web Server *– Online server that keeps track of all data with support from a database.*