# CALIFORNIA STATE UNIVERSITY SACRAMENTO

DARK INERTIAL MAPPING: END OF PROJECT DOCUMENTATION

Authors: Andre JULIEN Yan LIN Alexander MEADOWS Joseph YOUNGBLOOD

> *Instructor:* Professor TATRO

> > May 03, 2015

# CONTENTS

i

Abstr	act	1	
Index	Terms	1	
Ι	Introduc	ion 1	-
Π	Design I II-A II-B II-C	ea and Revised Societal Problem1Design Idea Contract2I-A1Background2I-A2Inertial Navigation2I-A3Feature Set2I-A4Feature: Self-Referential2I-A5Feature: Wearable3I-A6Feature: Local Data Storage3I-A7Feature: Visualization Software3I-A8Current Implementations4I-A9Dead-Reckoning Mitigation4Arket Review5I-C1Public Sector5I-C2Private Sector6I-C3Market Competition7	
III	Financia	<b>Responsibility</b> 8	}
IV	Work Br IV-A IV-B IV-C	Pakdown Structure8Self-Referential8V-A1Read Sensor Data8V-A2Calculating Position from Data9V-A3Test Accuracy10Vearable11V-B1Build Components on breadboard11V-B2Battery Implementation11V-B3Create Housing12V-B4Field Testing12Vorses Data Locally13V-C1Select file format13V-C2Connect SD shield to microcontroller13V-C4Make a button to enable/disable data recording13V-D1Data Manipulation13V-D2Drawing13V-D3Graphical User Interface13V-D4Debugging14V-D5Deployment14	
V	Target D	padlines 14	Ļ
VI	Risk Ass VI-A	Assessment14Vevere14VI-A1Lose Accuracy Due to Drift14VI-A2Design Unable to Meet Expectations15VI-A3Positional Measurements Unavailable15VI-A4Overworked/Exhaustion15VI-A5Lack of In-house Skill15	

	VI-B	Major VI-B1 Fe VI-B2 Se	eature Exceeds Time Quota	15 15 15
		VI-B3 C	ommunication Breakdown	15
		VI-B4 In	sufficient Hardware	15
		VI-B5 D	elay do to non-project priorities	15
	VI-C	Medium		16
		VI-C1 H	ive Down	16
		VI-C2 Fi	ile Structure Exceeds Filestructure Limitations	16
		VI-C3 In	Iterface Unintuitive	16
		VI-C4 Fi		16
	MD	VI-C5 Id	leal Sampling Rate Unattainable	16 17
	VI-D	$\begin{array}{ccc} \text{Minor} & \cdot & \cdot \\ \text{VID1} & \text{T} \end{array}$	aam Mambar Absance	10 16
		VI-D1 IC	ower Consumption Concerns/Expectations	10 16
		VI-D2 IV		16
		VI-D4 Pe	ersonal Computer Failure	16
		VI-D5 D	Pata Hosting Downtime	17
VII	Tasks (	Completed in	Fall 2014	17
VIII	Tacke I	Indertaken in	Spring 2015	17
V 111		Problem Stat	tement Revision	17 17
	VIII-B	Device Test	Plan	17
	VIII-C	Market Revie	ew	17
	VIII-D	Mid-Term Pi	rogress Review	18
	VIII-E	Feature Press	entations and Reports	18
	VIII-F	Deployable I	Prototype Review	18
	VIII-G	End of Proje	cct Documentation	18
	VIII-H	Self-Referent	tial	18
		VIII-HI Se	ensor Selection	18
		VIII-H2 E	stablish Kinematic Model	18 10
		VIII-H3 III VIII-H4 D	lenace Devices	19 19
	VIII-I	Local Data S	Storage	19
	,	VIII-I1 In	nplement Data Storage Protocol	19
		VIII-I2 N	ew Search Session	19
	VIII-J	Software		19
		VIII-J1 C	amera Rotation	19
		VIII-J2 M	ſenu	19
		VIII-J3 E	xit	19
		VIII-J4 L	0ad	19 10
		VIII-J5 D	Isplay	19 10
		VIII-J0 D	ines	19 19
		VIII-J8 C	ompile Code	19
	VIII-K	Wearability .		19
		VIII-K1 3I	D print housing	19
		VIII-K2 B	attery selection	20
		VIII-K3 In	Itegrate Device into Housing	20
		VIII-K4 Pe	erformance Testing	20
IV	Hear M	อทบอไ	,	<u>2</u> 0
іл		Device User	e Overview	20 2∩
	IX-R	Limitations	<u>, , , , , , , , , , , , , , , , , , , </u>	20
		IX-B1 A	ccuracy	20 20
		IX-B2 U	sability Conditions	20
		IX-B3 R	un-time	20

	IX-C IX-D IX-E IX-F	IX-B4Memory CardHardwareUsageUsage	20 20 20 20 20 21 21 21 22 22
Χ	Design	Documentation	22
XI	<b>Hardwa</b> XI-A XI-B	Bre Block Diagram and Document at Block Level	23 23 25
XII	<b>Softwar</b> XII-A XII-B	e Block Diagram and Document at Block Level	27 27 27
XIII	Mechan XIII-A XIII-B XIII-C	ical Device Housing Main Body	28 28 29 29
XIV	<b>Test Pla</b> XIV-A	an and Test Results for Hardware Static Timed Test	29 29
	XIV-B	XIV-A1       Test results       Speed Test         XIV-B1       Test Results       Speed Test	29 29 29
	XIV-C	Drop Test	30 30
	XIV-D	Crush Test	30 30
	XIV-E	Shake Test	30
	XIV-F	Dissipation Test	30
	XIV-G XIV-H	Moisture Test       Shatter Test	30 30
	XIV-I	Battery Runtime Test	30 31
XV	Test Pla	an and Test Results for Software	31
11,	XV-A	Short Distance Test	31
	XV-B	XV-A1       Test Results         Medium Distance Test	31 31
<b>V V T</b>	I D	XV-B1 Test Results	31
XVI	Long D	Istance lest         XVI-2       Test Results	32 32
	XVI-A	Unit Test	32 32
	XVI-B	Sample Data Test	32
хvп	Record	Data File Test	32
	XVII-A	Test Results	33
XVII	l Overloa	d Test	33

XVIII Overload Test XVIII-A Test Results . • • . . . . . . . . . . . . . . . . • . . . .

33

XIX	onclusion	33
Glossa		33
Refere	es	33
Appen	ces	35
1	Data sheets	35
]	Resume	40

# LIST OF FIGURES

1	HMC-6343 Three-Axis Compass (By: Team D.I.M.)	15
2	Device Interior (By: Team D.I.M.)	20
3	Device Strap Back (By: Team D.I.M.)	21
4	Device Active Front (By: Team D.I.M.)	21
5	UI Functionality Navigation taken by team D.I.M.	21
6	UI Functionality Run Color (By: Team D.I.M.)	21
7	Basic Data flow (By: Team D.I.M.)	27
8	Basic Step Calculation (By: Team D.I.M.)	27
9	Basic Step Calculation (By: Team D.I.M.)	27
10	Main case body (By: Team D.I.M.)	29
11	Main case, the other parts (By: Team D.I.M.)	29
12	Static Time Test	29
13	Drop Test resulted in a fracture in the enclosure (By: Team D.I.M.)	30
14	Short Distance Test around center tables in 3001 Riverside Hall (By: Team D.I.M.)	31
15	Line Test (By: Team D.I.M.)	31
16	Medium Distance Test on 3rd Floor Hallway, Riverside Hall (By: Team D.I.M.)	31
17	Long Distance Test between Riverside Hall and Sequoia Hall (By: Team D.I.M.)	32

# LIST OF TABLES

Ι	Product pricing	8
II	Feature: Documentation Deadlines and Completion Dates	14
III	Feature: Self-Referential Deadlines and Completion Dates	14
IV	Feature: Local Data Storage Deadlines and Completion Dates	14
V	Feature: Software Deadlines and Completion Dates	14
VI	Feature: Wearability - Deadlines and Completion Dates	14
VII	Feature: Self-Referential Fall 2014 Completed Tasks with Hour Allocation	17
VIII	Feature: Local Data Storage Fall 2014 Completed Tasks with Hour Allocation	17
IX	Feature: Wearable Fall 2014 Completed Tasks with Hour Allocation	17
Х	Feature: Software Fall 2014 Completed Tasks with Hour Allocation	17
XI	Completed Course Assignments Fall 2014 Individual Member Contributions	17
XII	Feature: Documentation Spring 2015 Completed Tasks with Hour Allocation	18
XIII	Feature: Software - Spring 2015 - Completed Tasks with Hour Allocation	18
XIV	Feature: Local Data Storage - Spring 2015 - Completed Tasks with Hour Allocation	18
XV	Feature: Wearability - Spring 2015 - Completed Tasks with Hour Allocation	18

# Dark Inertial Mapping: End of Project Documentation

Andre Julien\*, Yan Lin<sup>†</sup>, Alexander Meadows<sup>‡</sup> and Joseph Youngblood<sup>§</sup>

Department of Computer and Electrical Engineering, California State University Sacramento

Sacramento, CA USA

Email: \*andrejulien@csus.edu, <sup>†</sup>yanlin@csus.edu, <sup>‡</sup>alexander.s.meadows@gmail.com, <sup>§</sup>jpy28@csus.edu

Abstract—Search and rescue personnel typically conduct operations within under two categories: field search, and coordination management. In urban environments where building geometry contributes to signal dampening and reflection, popular localization methodologies such as GPS cannot be employed effectively. The Dark Inertial Mapping system seeks to address technical deficiencies seen within the popular methods of search and rescue under these circumstances. This is accomplished using a hip-mounted wearable device that utilizes a PDR algorithm to generate displacement, allowing for the search coordinator to analyze and efficiently deploy consequent searches, to search efficiency. The wearable device generates and stores displacement data, and includes accompanying software dedicated to the visual interpretation of device data. The intended aim for such a system to provide a better level of search path information regarding field search personnel to the search coordination manager. By reducing path redundancy, the device aims to save more lives through enhancing search coordination efficiency.

*Index Terms*—test process, tests results, self referential, wearability, data storage, software testing, mapping.

#### I. INTRODUCTION

Urban Search and Rescue is the location, extraction, of and provision of aid to civilians trapped in confined spaces [1]. Organizations such as FEMA have implemented infrastructures to expedite these search efforts through division of labor based on the nature of work performed. The two primary roles associated with the execution of search activity are Task Forces and Incident Support teams [1]. With the assistance of technology, these teams are capable of coordinating to efficiently relay information regarding a catastrophic event. However in urban areas, satellite-based localization technology such as GPS have proven to be unreliable due to signal dampening and reflection caused by building geometry. In these situations, PDR-based localization algorithms are typically employed [2]. Realizing the difficulties of of generating search paths explicitly from memory and verbal exchange, Team DIM comprising of Andre Julien, Yan Lin, Alexander Meadows, and Joseph Youngblood set out to create a passive, wearable device capable of generating, and approximating search path data from Urban Search and Rescue task forces. To further enhance the results of search efforts, a complementary software visualization aspect was built intended for use by Incident Support teams to aggregate the search data generated by these devices into a visual model. Initially the team focused primarily on implementing kinematic model based on double integration of accelerometer data. Due to complications that

arose as the result of non-uniform rotational data, in conjunction with accumulated error over time, the kinematic model required drastic revisions in order to fall within the drafted featureset within the Design Idea Contract. Based on research conducted involving Pedestrian dead-reckoning systems, the system's kinematic model was changed to record and interpret human gait motion. Based off of the testing performed, the final product was capable of generating paths with an error of 5% over 90 meters. The aim such a system for Search and Rescue efforts in urban search environments would contribute toward the increase the number of lives saved by reducing search path redundancy in consequently scheduled searches.

#### II. DESIGN IDEA AND REVISED SOCIETAL PROBLEM

Initial research conducted during the Fall semester of 2014 showed that there was a compelling societal need for the improvement of Search and Rescue operator protocol. Key resources of interest included topical journal review spanning topics from coordinated robotic search and rescue to SWAT team coordination protocol. Based on the information found, a solution to meet the technical needs of Search and Rescue teams was proposed in form of the Design Idea Contract. For counsel regarding field operations and standard operating procedure, the team consulted Ray Stedronsky, a retired Captain from Truck 106 Metro Fire with more than three decades of field experience. Due to technical issues that occurred during the Fall Project Semester, the team required a change in kinematic model in order to meet the original contractual obligation detailed under the Design Idea Contract by the term deadline. The original kinematic model was based on translational motion, assuming a static orientation of the device itself. To calculate displacement, the device recorded acceleration data to external storage. Due to the low throughput of the AtMega328 platform, data was postprocessed in the visualization software in order to maximize sampling rate of sensors. However, this led to the requirement of rapid orientation analysis to determine the linear acceleration with respect to the frame of reference of the earth. Critical issues consisted of poor sampling rate, accumulated error in integration of accelerometer data, non-symmetrical heading, along with a large number of corner cases attributed to device orientation. To mitigate the risks associated with an undeliverable design, the project switched from the original intent of double-integration of acceleration data to a revised

implementation utilizing a dead reckoning algorithm. During the spring semester, a market review was also conducted, assessing and reviewing previous attempts by other teams to address similar issues in coordination of Search and Rescue efforts.

### A. Design Idea Contract

1) Background: Based on research conducted by Team DIM, it was made evident that the condition of "Lost or Disoriented" was considered an impactful cause of loss of life for first-response emergency personnel. Publications released by FEMA between the years of 2003 and 2012 provided the statistics of fatalities. In these annual reports [3], the U.S. Fire Administration listed "Lost or Disoriented" as a category for nature of fatality, containing resulting cause of fatality as a result of these factors. Within the given years of the reports, 10 personnel fatalities were reported in structure fire incidents, where the nature of the deaths was primarily due to asphyxiation. With the population increase in urban areas and the age and condition of buildings, there is an increasing concern of death and injury in the case of a catastrophic event.

"The University of California has released to Los Angeles city officials a list of about 1,500 old concrete buildings that are potentially at risk of collapse during an earthquake [4]."

Compounded with predictions that catastrophic events will increase, more effective and reliable technologies are necessary to assist Search and Rescue personnel.

"We predict that the number of earthquakes with death tolls greater than 100,000 (50,000) will increase in the 21st century to 8.73.3 (20.5 4.3 from 4 (7) observed in the 20th century." [5]

These figures suggested an increase in Search and Rescue missions that will require planning and coordination among rescue teams searching through buildings.

"Collapsed structure rescue work is dangerous, even with optimal safety practices (being able to accurately determine when further risk becomes unwarranted, may minimize overall risk to rescuers)" [6] With inefficient or nonexistent monitoring of rescue personnel there is no information about what locations were searched, so it would not be uncommon for an area to be searched repeatedly putting the rescuers in undue risk. With accurate information, search and rescue teams can quickly find holes in their patterns to ensure that all locations are checked.

2) Inertial Navigation: As a result of the research conducted, Team DIM was opted to devise a solution that would assist rescuers in expediting the extraction process of victims, preventing unnecessary deaths due to unfamiliar environments, low vision, or disorientation. As an initial point of contact, the members of Team DIM contacted Captain Ray Stedronsky. The initial correspondence provided insight for the standard operating procedures that are typically employed for search and rescue operations in interior environments [7]. In these situations, the priority is placed on covering as much ground as fast as possible due to the rapidly diminishing survival conditions. Search and rescue responders are paired, and typically use a right-hand method to traverse hallways until they can find a room. When a room is located, one rescuer stays at the door, while the other checks the room, and acts as an exit beacon for their partner. For large areas, a long rope (typically 200 feet) is used with segmented knots every 50 feet. This rope is anchored to a sturdy point at the entry area. A 20 secondary foot-rope with a metal ring at the end is affixed to the rescuer, and the larger rope is inserted through the ring. The knot/ring sequence presents a rough estimation of distance every 50 feet, and ensures that the operative does not stray too far from the rope leading to the exit. These methods have been a long-standing navigation methods in the search and rescue field, as they have proved reliable given typical interior design methodologyIf a wall is followed continuously, the navigator should return to their starting point. Additionally, they require no dependence on external sources. Using these principles, Team DIM sought out to make a device was both capable of determining displacement, as well as providing a rudimentary map of the path traveled.

3) Feature Set: Upon establishing that the loss of life due to unfamiliar environments or lack of orientation was a valid concern, Team DIM was tasked with the creation of a device that would address the needs of rescue personnel and provide them a device that would expedite their efforts. The features highlighted in the Design Idea Contract consisted of the following:

-Self-Referential

-Wearable

- -Local Data Storage
- -Visualization Software

4) Feature: Self-Referential: One of the key features of Project DIM is the capability to generate position from solely the data provided from the devices inertial measurement unit. Self Referential devices do not rely on any outside information to provide accurate output. Our device is self referential and uses no outside signals to interpret where it is from an arbitrary starting location. This feature will be referred to as Self-Referential Position Calculation. To fully utilize this feature, it necessitates the utilization of collected data to generate a positional calculation from an initial start position. The best way to understand a feature is to understand the decisions made during its implementation. As our project started it was observed that communication in emergency situation can be unobtainable. It was noted that there were several issues with devices that attempted to communicate in these situations and that these devices had been proven unreliable or the equipment was too heavy, bulky, cumbersome, and expensive this led to these devices being left behind due to its inhibiting nature [?]. These issues were problems that wanted to mitigate and Team DIM decided that our device should be a self contained, lightweight, and low-cost system. This meant that the proposed system had to be made up of inexpensive and lightweight parts. Since obtaining signals out of these environments sometime takes a large antenna and a signal generator to send out we were not able to use communication on our device. Team DIMS device will be able to track its position on its own and could not rely on outside communication for assistance or back up. This became the core parameter of our project and was

taken as a benchmark of reliability to be self sufficient and require no set up of communication devices during a time of crisis. This would minimize impact on the search and rescue teams who implemented our device. Because the hardware was chosen to be both passive and self referential the only visualization tool was through the software package. Thusly the hardware depends on the software to display its results. The hardware and software choices were made as a joint effort such as choosing the file type and choosing what data is saved. Furthermore when testing the device more requests for functionality were made for the software. These features were added to help testing and typically proved to improve the software. The hardware components were also limited to the specificity that they could be wearable. This meant that large and bulky equipment could not be added to the hardware even though initial prototypes had a laptop connected. This also meant that the hardware in order to be self sufficient needed to have its own power supply. This power supply will need to easily last the entire time that a search and rescue mission will last. Lastly the choice of using a micro-controller with limited memory compounded with the self referential inability to wirelessly send the data dictated that external storage was added to the device. This tied together the choice of being self referential and storing data locally. Requiring that to be self referential the device needed to store data locally.

5) Feature: Wearable: Wearability is the feature that in the end will define the actual marketability of the device. All components being equal if the end user cannot affix, remove, comfortably and most importantly safely wear the device it will never be utilized. The required mounting point for the device is on either hip of the end user. The device should be inserted into a soft moisture-resistant exterior housing which will protect the device along with padding the device for user comfort.An appropriate solution was found by using a readily available product. An Energizer Lithium battery was chosen since its output exceeded the real world run time that a user would need.

6) Feature: Local Data Storage: For data storage, team D.I.M. was faced with two options: Wireless off-device data storage, or local data storage. D.I.M. chose the latter rather then the former because, while storing the data on the device seems rudimentary, it is the least likely to have problems created from the environment and relative distance from the main base. This is one of the reasons why wireless data transference and off device storage was decided against. The problem with wireless data transference and off device data storage is threefold. Not only is wireless connectivity not reliable, it is limited by range, and requires significantly more power. The first problem stems from the extreme reliability required by a pedometrical positioning system for reference to previous position. If a single step is erroneously sent or is not received by the base station, then the entirety of the following positions sent will be off by one step in any number of possible directions. The second problem of wireless systems having limited range stems from wireless signals range dropping when there is something blocking the signals path, such as walls, houses, and people. Because of this, it is not guaranteed that the signal will be able to reach the base

station which would create positional accuracy problems. The third and final problem of wireless power consumption, which is observable using the everyday smart phone and YouTube. If one watches an episode of Dragnet on their smart phone from the phones memory, the power usage is much less than if they were to watch the same episode on YouTube using a wireless connection. This extra power consumption would become a problem in wearability as it would require the device to store more power internally and thereby have a larger battery which would increase the weight of the device making it more uncomfortable to wear. Because of the combination of these three problems, it would not be logical to pursue data storage using a wireless off device storage mechanism. Since wireless off-device data storage was decided against, the implementation of local data storage has to be considered. this can be broken into three parts: storage medium, storage connection, and file type. The feature of local data storage depends on the other features and parts of the device in order to work. It requires data to be read from the sensors used in the self referential feature in order to have data to record. The wearability feature requires that the device housing can fit the storage hardware without the device as a whole becoming burdensome to the wearer. The software component feature had the most impact on the local data storage feature because not only did it help determine the file type used to record data to, it also helped determine the data recording medium that would be used based off of what connection ports are available on the common laptop

7) Feature: Visualization Software: The software portion of the system is created using the Unity3D Game Engine. The engine is used to generate a 3D game space capable of providing a visual representation for gait data. The software allows for the loading of multiple search paths, over a user-defined map backdrop, and dynamic point selection. Provided a pre-determined average gait size, each individual gait reading that triggers a pre-defined step threshold, creating a step object. The position of additional step objects generated utilize basic dead-reckoning principles. The software connects these individual step objects, and allows the assignment of color for clarity for the end-user. The primary aim of this implementation is to provide a better visual approximation of search paths, contoured toward search coordinators for improved search efficiency through reduced redundancy. The rendering engine provided by the Unity3D environment acts as the middle layer between recorded gait data and final display. Unity provides the tools to generate a gamespace, which can be populated with game objects. Every visual aspect within the environment is rendered as a game-object. This results in a 3D environment that can be traversed using a flycamera developed by Joseph Youngblood. The overall intent of Project DIM is to facilitate search coordination for Search and Rescue Coordination Managers. The goal of the Dark Inertial Mapping system was to create a device that required minimal interaction could be easily incorporated into search operatives equipment that would allow for search path mapping. While a significant effort was placed on the DIM device to establish a valid kinematic model, the DIM software represents a proposed solution to reducing search path redundancy for human operatives. Upon

each completed search, the search coordinator can load the devices generated data into the software, and better assess scheduling future searches. With an aggregate search map, this can eliminate repeated searches, allowing for a better rate of survival for victim extraction. For functionality, the visualization software aspect has a complete dependency on valid data being generated from the DIM device. The two core components within the visualization are the Pedometer() function and the CalculatingData() function. The responsibility of the Pedometer function is to post-process the accelerometer data generated by the MEMS device on the DIM device. The way this is accomplished is through the implementation of a static gaitthreshold. When a file is loaded, this provides two key values to the software: z-axis acceleration, and heading. Acceleration was provided in gs (9.81 meters per second squared) and heading was generated in degrees anti-clockwise from North. Each line is parsed, values increase over a certain value, a pedometer step is detected, and a game object is created using this data. For testing purposes, a variety of static thresholds were tested to find an acceptable range in which steps were properly tracked. In the internal version of this software, this is a soft-threshold, where the values can be modified to assess the impact of different thresholds on the resulting map. CalculatingData() is the software implementation of a pedestrian dead-reckoning algorithm. This function parses through the game objects generated by the Pedometer() function, and calculates their placement through the Cartesian coordinates of the allocated game space, determining the x and y axis displacements through trigonometric manipulation of the generated heading and pre-determined gait sizes.

8) Current Implementations: Aside from the reliable, yet rudimentary standard search and rescue procedures employed by first response emergency personnel, a number of other approaches have been taken to address interior mapping needs. In an article posted in September 2013 on IEEE Explore, a number of safety systems were evaluated with first responders as their target audience [3]. One particular safety system titled the PASS (Personal Alert Safety System) generates a piercing distress beacon for rescuers to follow. However, in one instance of its use, due to the deteriorating state of the building along with the unfamiliar environment, rescuers could not locate the audio beacon. This eventually led to the loss of life for the lost firefighters, and those who attempted to locate them. In a world that quickly embraced navigation and tracking technology, it would seem that these applications would be an instant panacea for the societal problem. However, indoor environments suffer due to the need for a direct lineof-sight to the satellites, or a pre-set connection to an internal access point. When a disaster has compromised these preestablished services, they offer no valid solution. A detailed assessment of Search and Rescue localization technologies is conducted in the Market Review section of this documentation. One of the key features that numerous mapping systems share is their reliance on an external source. This provides a level of accuracy but contains a high degree of risk when the operation of the entire device rests on connectivity with the base station. Additionally, these systems require additionally rescuers to enter a hazardous environment to extract the "lost" operatives generating a distress beacon. Through the creation of a "dark", self-referential device, Team DIM aimed to provide first response emergency personnel with an apparatus that is free from the complexities of compatibility and reliance with pre-installed systems under time-sensitive situations for selfextraction from hazardous environments. From research conducted, current methods necessitated a dependency on wireless communications systems in order to determine position, and positional displacement. This motivated Team DIM to create a system that they deemed "Self-Referential". Such a system would be realized by ensuring that entire system was capable of attaining positional displacement from an arbitrary starting point without reliance on an external system. One of the ways in which "self-referential" could be attained is by the storage of orientation and displacement data locally. Microcontrollers typically have a limited amount of fast on-board memory reserved for the storage of compiled code. For persistent data collection over a prolonged period of time, this data should be stored locally by way of a storage controller, and a removable disk. Successful recording of useful displacement data allows for the software feature to utilize the data and generate visual representation of positional displacement. For a device to be wearable, the apparatus must be operational without a dependency on a persistent connection to a stationary power source. Given the time constraints of a failing structure, this may necessitate an operating time of less than an hour. In situations where the building is simply difficult to recognize due to its dilapidated nature, it may require a larger battery to fit these requirements. By building a software suite, the locally stored data can be visualized, and from this a rudimentary map providing details about search patterns and traversal paths can be generated. This can provide valuable information that can assist in the reduction in redundancy, and increase search efficiency.

9) Dead-Reckoning Mitigation: The overall breakdown of the software and hardware components is discussed in greater detail in the latter portions of this documentation. As a result of the mitigation plan undertaken, changes were made to the testing plan as well as the design in terms of both kinematic model and the corresponding interpretation of device data in the visualization software. In order to still adhere to the original design idea contract, the changes made to the device and software were strictly internal, producing a similar experience in for demographic usage. This consisted of maintaining the original accelerometer, while integrating a high-precision Honeywell HMC-6343 3-axis compass to accommodate for the problematic magnetometer incorporated in the 9DOF LSM9DS0 package.. To measure the gait of the user, the device is hip-mounted to the operator during a performed search to record gait data. This data is then analyzed by the visualization software, which generates steps objects within the user environment, utilizing PDR with a pre-determined average gait size, as well as a heading value generated from the new compass.

## B. Revised Societal Problem

Based on a better understanding of the project, the group began the Spring 2015 semester with a refinement of the original problem statement. From the original design idea contract, the project's contextual shift moved from the extraction of the individual operator, to enhancing the coordination efforts of search teams through the search coordination manager. Standard operating procedures typically have search operators use visual cues to traverse their environment, when available. Additionally, they employ the use of spray paint to mark and provide information regarding their surroundings to other teams. Due to the amount of improvisation that occurs in this regard, a degree of their success relies on the plans laid out by the coordination manager. Without synchronization of information between rescue teams, victim searches cannot be coordinated efficiently [2]. In some instances, coordination of efforts are addressed by automation in Search and Rescue robots. In this field, similar concerns exist when designing systems that allow automated coordination and communication. An increase in the distance covered reduces the efficiency and likelyhood of communication and collaboration [8]. Additionally, machines are limited to their interpretation of data. While human communication suffers a similar issue regarding communication of contextual visual data, analyzing only team search paths for redundancy can better assist coordination staff to improve repeated search efforts and encompass more terrain in less time. During the Fall 2014 project term, team DIM sought out to design a prototype in order to specifically address the problem statement shortcomings of standard technologies. The problem statement was initially targeted toward the extraction of firefighters in a dynamic environment. The device design consisted of a device that was to be self-referential, portable, and used locally saved data to generate a visual representation of the path the end user traveled. The device was aimed to fufill the woes of indoor navigation as a result of poor GPS reception, and a lack reliability of indoor positioning systems in the face of a disastrous emergency. It was through this effort and an additional mitigation plan that the team understood why the "Right Hand Method" was a preferred method in emergency responder self-extraction. The absolute simplicity, as well as the reliability are the primary reasons for its repeated inclusion in firefighter safety training over the course of several decades. While the possibility exists to replace the technique with technology, prolonged use of a device in extreme heat can tamper with the life of the device as well as reliability of any sensors. As this can result in operator and survivor death, the method was not ideal, and the target demographic was reassessed. The initial assessment of the target demographic behind the societal problem statement was not completely erroneous. It was during validation testing of the visualization software during the mitigation plan that the team members noted the suitability for such an implementation toward the search coordination efforts. Upon further research, the team discovered Search and Rescue teams requesting better methods for enhancing their search pattern efficiency. According to a requisition document authored by the Savannah River National Laboratory in 2004, one of the highest priority requests for Search operations was for Search Tracking and Monitoring. "A detailed process of making physical marks with paint is used to identify areas that have been searched. A more sophisticated, electronic method is desired to record, track, and identify areas that have been searched to prevent unnecessary redundant search." [9] Supplemented with documentation addressing the importance of coordination, and the high cost of failure when it is poorly implemented [10], team DIMs primarily focus remained in Search and Rescue efforts, but aimed to address one of the core aspects of what differentiates a successful search from an unsuccessful search. To determine the value of such a device and the areas in which such a product would thrive, a market review was conducted on a number of devices that were contoured toward Search and Rescue operatives.

## C. Market Review

In order for a product to succeed within its target demographic, the team behind the innovation must understand the market in which device will thrive. This understanding requires knowledge of the consumer base, as well as utilization of such to form a marketing strategy to sell the product successfully. To accomplish this, an analysis of the potential market in both the public and private sector was conducted. Additionally, similar products existing within the same marketplace were analyzed for their merits and shortcomings in addressing the target demographic. This research is utilized to propose a strategy intended to target potential investors as well as clients for market penetration.

1) Public Sector: FEMA: A primary organization that falls within these confines is the Federal Emergency Management Agency (FEMA). From statistics provided FEMA [1], the participants within the National Urban Search and Rescue Response System can be categorized within three distinct groups. FEMA establishes policy and leads coordination of the national systems. Within the continental United States there are 28 FEMA Urban Search and Rescue Task Forces who are trained and equipped by their parent organization to handle structural collapse. Incidental Support Teams are trained to support the Urban Search and Rescue Task Forces by logistical, electronic, and coordination means. Within each Urban Search and Rescue Task Force, there are 62 positions. In the event that an emergency occurs, tasks force typically have more than 130 well-trained emergency responders prepared for action. The teams are composed of firefighters, engineers, medical professionals, canines and their handlers, and Urban Search and Rescue managers. Based on the provided statistics, this shows a potential customer base of over 3000 units for FEMA Urban Search and Rescue alone. In 2004, FEMA sponsored the publication of an article by the National Institute of Justice detailing the technological needs of Urban Search and Rescue. Specifically, they detail the high priority of a technology that could provide search tracking and monitoring to supercede rudimentary physical marking techniques. "A more sophisticated, electronic method is desired to record, track, and identify areas that have been searched to prevent unnecessary redundant search." Since then, an Urban Search and Rescue Report was generated by FEMA in 2011 detailing the Task Force Cache List contents [2]. In this document, there contained no mention of the inclusion of any sort of previously requested indoor tracking technology. Furthermore, the mention of mobile computers being readily available

through the cache position the Dark Inertial Mapping device as a prime candidate to fulfill such a request. The system ideally works for coordination efforts at the hand of the Urban Search and Rescue Incident Support team, applying primarily to collapse search efforts, but also fulfilling the needs of wilderness and certain cave rescue searches. The end-devices form factor as well as its handsfree field operation allow for the inclusion of the device to minimally encumber the FEMA Task Force operatives around existing equipment. Fire Rescue: The original conception of the idea for an indoor tracking system resulted following the teams discussion with retired firefighter Raymond Stedronsky. The topic of the discourse consisted of devising an engineering solution as a means to improve a rudimentary method of extraction. The onset of wearable computing has led to innovations in the field of creating a more futuristic firesuit, among other modern technologically augmented versions of existing hardware. While the initial discussion mentioned the utilty that a heads-up navigational display could provide for fire rescue personnel during emergency extraction, an emphasis was placed on the necessity of a high degree of reliability for the device. Because rescue personnel would be entrusting their well-beings to such a device, failure to do so would marr its reputation in the field, as well as a reluctance to revisit such a device for future use. Due to the extreme nature of the environments for this particular uses, there is a chance that the sensors used for determining orientation and gait could have their accuracies compromised. Furthermore, due to the alteration in gait patterns when traversing a dynamic environment, matching the gait pattern the device detects may not be possible for the reliable generation of search paths. While the device does not ideally fit the needs within fire rescue in a live fire, the device is still capable of fitting the needs of firefighters for postdisaster search and rescue survey for reducing the redundancy in search pattern coordination.

2) Private Sector: While government search and rescue efforts are valuable, private volunteer groups are also part of the search and rescue market. As these groups are nonprofit, most of them have much smaller budgets and must use them wisely. These groups generally are called upon by local emergency response establishments to respond to particular search and rescue events, such as natural disasters and lost persons and are generally formed by persons of the community who are willing to take the time and energy to go through a training program and be willing to assist in a search effort when called upon. This training is either done by more experienced rescue personnel within the organization, as is done by the California Explorer Search and Rescue [11], or by going to a search and rescue school. While there are multiple smaller search and rescue schools, such as Sierra Rescue or King County Search and Rescue Association [12] [13], the main one is the National SAR School run by the US Coast Guard and Air Force [14]. Because these volunteer groups work in unison with local emergency crews, they have a need to communicate what areas they have searched back to the local emergency crews who are organizing the search effort, especially when doing a large scale search for missing person(s). These volunteer search and rescue organizations can be classified by the scale and locations that they serve. Specifically, they can be classified local, state, and international. Local- Local search and rescue is groups like the Eugene Mountain Rescue [15] or Sacramento County Sheriffs Search and Rescue [16] that focus on doing search and rescue in the area around their city or in their county. Because these groups focus on a smaller area, their budgets are smaller and most of the equipment is provided by the volunteers themselves [8]. However, because they are responsible for a smaller area, the teams are smaller (45 persons for Sacramento county sheriffs search and rescue). As such, low cost is important to these organizations. Beyond the previously mentioned organizations, there are at least 10 teams in Will County [17], 1 team in Travis County [18], 40+ volunteers in New York Search and Rescue [11], and 160 volunteers in the El Dorado County Sheriffs Search and Rescue Unit [19]. Even with ignoring city specific search and rescue teams, because the United States contains 3007 counties, if the average was 35 people in search and rescue per county, that is still a market of over 100,000 people. State- State scale search and rescue teams focus their efforts within the confines of their state borders. While the groups are generally larger and slightly better funded, they are comprised mostly of volunteers. With numerous organizations such as Central Massachusetts Search and Rescue Team [20], Texas Search and Rescue [21] and California Explorer Search and Rescue [22], some with over 160 members [14], the possible market increases further. Unfortunately, most of these organizations do not include information about how many members they have, making it difficult to pin down an exact numerical value. That being said, even if the average is 80 members, half of Texas Search and Rescue membership [14], state scale search and rescue organizations would still provides an additional market of 4000 people. International-The main market for private/non-profit organization comes from international organizations because of their massive size. The Red Cross alone has 13 million volunteers [23], while others like the Disaster preparedness and Emergency Response Association do not state their membership count. Other organizations, while they do not specialize in search and rescue, do assist when the need arises [24] or prepare locations for a disaster [25]. All of these organizations, because of their desire to save as many lives as possible, increase the market size immensely, to be in the millions. D. Private Sector For Profit Beyond just search and rescue non-profit organizations, there is a potential market as an augment to metal detectors to improve their search efficiency, theoretically increasing their returns. With the increase of gold prices, high quality metal detector sales have greatly increased to be in the hundreds of millions [26]. This indicates that more people are willing to spend money on this hobby and that they are taking it seriously. In England and Wales alone, it is averaged between 15,449 and 16,777 active metal detector users alone [27]. This user base further increases the potential market for our device. However, there is sure to be competition.

3) Market Competition: Due to the fact that the target market space is specifically emphasizing the Manager and search coordinator, our main competitors are people who are currently able to gather and store data about a persons location. These companies could quickly shift their focus and build a similar product to compete with the DIM system technology. A primary competitors is Wiggle, a company specializing in manufacturing systems that use Wi-Fi mapping. Wi-Fi mapping is based off of the principle of the relative range of wireless access points and signal strength. By knowing the locations and the ranges of each access point, trilateration is utilized using the relative position of the signal detecting device to give an approximate location. Wi-Fi mapping is accomplished by manual location entering and records the wireless access point by using specialized apps. This has been attempted successfully by Skyhook. Skyhook has developed a relative position algorithm using existing technology and infrastructure by using Wi-Fi access points. However, its accuracy is that of 3 to 10 meters, so while it is better than outdoor civilian GPS, it is not accurate enough to determine the position of a person or even the room they are in without creating a site-specific app. Even when there is a site-specific app, the location of a person is only accurate to approximately one and a half meters. The main problem with systems that use Wi-Fi mapping is that they not only require the mapping of the wireless networks before use, it also requires that those networks remain constant thus requiring local power to be available. While it may work in a static environment, any changes will make the pre-existing Wi-Fi maps to become incorrect. This includes changes to the structure of the environment and/or loss of power to those networks. [28] Another competitor is DecaWave who make Radio Frequency Chip systems. They operate by calculating either the time response or the phase response between the transmitter and the battery powered tag to track positional change of the chip while the chip is within range of the transmitter. DecaWave is a wideband wireless transceiver which uses low power tags and radio pulses to measure the distance between the tag and transmitter [29]. The technology boasts an accuracy of 10 cm at up to 5 meters per second. With transmitters being portable, it is possible to use them to find a tagged item. The transmitter has an impressive 300 meter line of sight range. However, the transmitter has at most a range of 40 meters without line of sight. While it is capable of tracking tag locations in real time relative to the transmitter, it lacks an integrated three-dimensional mapping system. Radio Frequency Chips not only require the prior installation of transceivers in most cases, but they also require the installation of battery powered chips. While ideally they could be referenced to a blueprint, they do not have any initial position tracking, nor are they capable of three dimensional positioning. These devices have a heavy introduction cost "...its cost: The components of a portable unit alone add up to between US 2000 and 3000, not counting the fancy battery." [28] This comes at a time "when nearly three-fourths of the countrys fire departments cant even afford spark-safe radios at less than 750 a pop." [28] FalTech GPS makes another competitive product called the GPS repeaters. They operate by relaying the position of the exterior GPS receiver that is located outside the building through a broadcast station. This signal is projected across the interior of a building. [30] "the repeater inside the building will relay the exact coordinates of the outdoor antenna to the interior of the building - it will not provide co-ordinates for the repeater unit itself." [23] The data being received by all GPS devices within the building are actually receiving the GPS location of the external receiver rather than the device. This demonstrates the inability of GPS to provide reliable indoor positioning even with the use of GPS repeaters. Another popular field in search and rescue previously unmentioned was rescue robotics. By creating a network of collaborative automated systems, developers are capable of rapidly checking through terrain that has been ravaged by disaster; This is in part due to awareness in position and environmental telemetry between devices to maximize the search efficiency. While robotic search systems have advanced to the point that they could be tested to be used in urban search and rescue, humans still do not want to trust the robots autonomous systems for search and rescue and prefer to remotely control them rather than allowing the robot to utilize its search programing. Additionally, due to the the dynamic nature of environments, humans are capable of interpreting unfamiliar data reactively, whereas in rescue robotics this can lead to errors based on their programming. Because of this distrust and the cost of the robots, humans will still be needed and used as search personnel, especially in areas that cannot afford the robots. Also, their positioning is still dependent on GPS and sending wireless data back to their base, thus limiting their range because of this communication requirement. An extensive evaluation of competing products was performed in "Location and Navigation Support for EmergencyResponders" authored by Fischer and Gellersen in 2009 [31]. In their study, they analyzed 9 different products using distinct movement models, and provided insight on the shortcomings of each implementation. These varied from positional drift over time, to sensitivity to environmental changes. The methods that employed beacons ran the risk of loss in position as a result of the beacons being moved or destroyed. Wireless implementations frequently had problems with penetrating the thickness of the structure, and experienced loss in signal and tracking over time. Methods that used PDR experienced drift and unpredictable error, but showed that the issues could be ameliorated by fusing the data with correctional methodologies. One particular method used Simultaneous Localization and Mapping in conjunction with PDR. However, due to the low level of visibility in live buildings, this particular method is ineffective incurring errors both fused methods. While there are multiple companies that attempt to accomplish the task of positional tracking using a variety of technology, there is a consistent recurrence of preinstallation requirements with reliance on local power and only twodimensional tracking; therefore, they lack the means to truly address the problem of positional recording of personnel independent of external systems.

4) Market Penetration: One of the greatest challenges in any event command is the coordination or personnel. "A successful outcome can only be accomplished if command and control is established early. The Incident Commander must demonstrate clear, expedited and decisive leadership" [32] illustrates the need for tools to assist in accomplishing the command and control directive. This approach is valid for all emergency events, from active shooter scenarios to urban search and rescue to subterranean extraction. From government run events to organization like Rescue Global who blend NGOs, Military, Government, Academia and private rescue, tools that are portable, require minimal training and are cost effective can provide invaluable assistance in accomplishing command and control. Using positional telemetry in GPS obscured environments, map generation can be accomplished using a fixed initial point. These maps can provide invaluable data to expedite extractions by mapping previously searched areas. Such an application can be used to better coordination efforts by multiple Search and Rescue teams. Telemetry from multiple devices can be combined to generate a composite map, allowing for the assessment of search efficiency. Furthermore, visualization of such data can improve coordination efforts between Search and Rescue teams which can result in a decrease in time-to-rescue, resulting in more lives saved. "Off the shelf components", rapidly deploy-able, low comparative replacement cost are all features that make Team DIMs passive mapping and visualization package a superior choice for dynamic large scale emergency incidents. The device visualization feature can easily be realized with any laptop with ability to run Windows 7 or later. A built-in or external card reader can be used to access the visualized mapped data for dynamic command and control situations. Second, the data acquisition unit is no larger than a box of matches and weighs less than 1 kg. This facilitates the transportation and use of many units to the scene which results in increased data acquisition. Finally, low replacement costs compared with any robotic solution that currently costs as much as 50k USD [26]. Low replacement cost enables the customer to reduce out of service times, by have device inventory redundancy. Between Governments and NGOs there is no device on the market that currently successfully addresss mapping GPS obscured environments. This creates an uncontested opportunity to penetrate this vacant segment. By providing an extremely valuable additional tool to enhance and facilitate the command and control imperative, team DIMs passive mapping device is positioned for a success and to be a leader in this vacant market segment while also potentially reducing casualties.

### III. FINANCIAL RESPONSIBILITY

Due to the boom of maker culture in the past decade, easyto-use sensors and microcontrollers are available from a larger array of vendors at extremely competitive prices. Devices that have very specific uses typically have higher price tags, of which may not fall within the realm of reason for Project DIM and Senior Project. In the event that a part is required but is a significant financial burden, team members will assess the cost of the feature, versus the significance of the feature to the core deliverable. The team will then assess and brainstorm ways in whether this feature could be realized using separate components of more reasonably priced hardware, and attempt to implement the same feature by providing communication and logic between the replacement devices.

## IV. WORK BREAKDOWN STRUCTURE

In this section of the documentation, the specifics and implementation of the features for the device will be described

Product	Price	Qty	Total
SparkFun RedBoard	19.95	4	79.80
NmicroSD Shield	14.95	4	59.8
HMC6343 Breakout	149.95	2	299.9
Stackable Headers	7.50	4	30.0
Stackable Headers	12.95	4	51.8
Logic Level Converter	2.95	4	11.8
Total			533.1

TABLE I: Product pricing

in detail. This includes explanations of the subtasks and the respective activities which comprise the subtasks. These activities will be further broken down if necessary to provide absolute clarity on the tasks required to attain realization of each proposed feature.

### A. Self-Referential

The key feature of the proposed device is that it is selfreferential, meaning that it has no dependency on an external system. This is to circumvent the connectivity problems face by devices that employ GPS or ad-hoc connections. For this feature to be realized, we must first read the sensor data, then calculate position from the data, and finally test the accuracy of the device.

1) Read Sensor Data: Interfacing a sensor with a microcontroller typically necessitates the use of a communication protocol for incoming and outgoing data. Not only does this require the communication protocol, in order to properly calculate the position, it requires that the sampling rate of the data be known as it is vital for kinematics. In order to reduce noise, the sensitivity of the IMUs must be adjusted. This has to be implemented in at least two different types of IMUs, the MPU-6050, and the LSM9DS0. For this subtask to be complete, we must be able to get data from the sensors in which we will post process.

The MPU-6050 uses I2C communication protocol and has a default sensitivity of 2G. This IMU has a low power mode in which it only samples the data at select intervals. This low power mode will be further investigated and most likely included in later versions of the device. The MPU-6050 also has self testing functionality in which each axis can be tested. I2C communication protocol is used to read and write to addressed register banks located in the MPU-6050. This is done by using a two line communication bus. One of these lines, called SCL, is dedicated to the communication clock which is produced by the microcontroller. This clock is used for synchronization between the sensor and the microcontroller for transferring data along the other bus line called SDA. In the SDA line, the changing of bits either up or down for preparation for sending the next bit must happen while the clock on SCL is low, else it could be interpreted as an early termination signal and error can and will occur. Understanding and implementing the I2C protocol to be able to communicate to and from the IMU is the initial sub-activity requiring completion, as the remainder of sensor implementation depends on this task. The MPU-6050 has a number of adjustable on-chip features that become available upon the establishment of the communication bus. As such, changing the sensitivity of the

MPU-6050 is dependent on the I2C communication protocol working. To change the sensitivity of the device, it requires that the microcontroller can write to the internal registers of the IMUs. The address for changing the sensitivity of the device can be located in the devices data sheet. There are a number of benefits and disadvantages to reducing the device sensitivity. One of the primary advantages of reducing the sensitivity is the reduction in noise obtained from the measured data. However, in doing so, acceleration data within the sensitivity dead-zone is not measured. The MPU-6050 has four sensitivity modes for its accelerometer and gyroscope. Research and testing will be conducted to ascertain a nominal setting that provides a minimal noise impact on the data, as well as an acceptable level of sensitivity. The IMU has the ability to lower its sampling rate to reduce its power consumption. In the MPU-6050 data sheet has four low power modes listed with their respective accelerometer current draw. These modes are: 10A at 1.25Hz, 20A at 5Hz, 60A at 20Hz, and 110A at 40Hz [33]. While this produces a reduction in the frequency of data being read from the sensor, it also can produce some power savings. It also provides a fixed sampling rate which will prove to be important for kinematic equation calculations. However, the most important part is how this reduced sampling rate affects the overall quality of the data, specifically, how it will affect the post calculation data. But being able to enable this mode and understanding how it works is required before the analysis of the quality of data is able to taken into consideration. Self Testing is an important feature for any device that is to claim reliability. While the MPU-6050 can run a self test, it requires additional algorithms to fully test the functionality of both the accelerometer and the gyroscope. This enables a warning system to be added to warn if the IMU is not functioning properly. It also allows for a temporary fix of turning off the broken part of the IMU and removing its data from calculations while not losing all functionality. However, for this function to work, it requires that upon startup that the device is holding still and may delay the startup time. The added safety of being able to tell when part of it fails is far more important than shortening the startup time of the device as a malfunction which produces erroneous data could lead to much graver consequences. As such, including the self test is vital for reading data from the MPU-6050.

The LSM9DS0 is an IMU with a total of 9 degrees of measurement. What this means is that beyond the accelerometer and gyro that the MPU-6050 has, it also has a magnetometer, all with adjustable sensitivity. The LSM9DS0 has two communication methods, that of SPI and I2C. This IMU also has a low power mode as well as embedded selftesting. It also has a useful interrupt feature which can be used to start and stop data recording. To communicate with the LSM9DS0, we have the option between two communication protocols: I2C and SPI. The differences between the two communication techniques provide a tricky decision for us, as there are advantages and disadvantages to both. While SPI is faster than I2C at a blazing maximum speed of 10MHz compared to I2Cs maximum speed of 400 kHz, it also requires using three wires, instead of I2Cs two wires [34]. The third wire stems from not having a bidirectional communication system, thereby requiring separate wires to send and receive data. Because of the additional data, the magnetometer data, being communicated to the microcontroller, having a higher data transmission rate than the MPU-6050 could be beneficial. As such, more research into the advantages and disadvantages of both communication systems will have to be done before deciding which to implement. Also, because of the different register structure between the two IMUs, the code can not be simply copied over. Instead it would require adjustments to the register addresses referenced in the code in order to work. Just like with the MPU-6050, the LSM9DS0 has adjustable sensitivity for each of its sensors. However, the adjustable range is different than that of the MPU-6050 in that is has five levels of sensitivity for its accelerometer and only three levels for its gyroscope. The magnetometer has four levels of sensitivity to choose from. Because of the difference in available sensitivity levels and given that it is produced by a different company than the MPU-6050, research and testing will be conducted to ascertain the nominal sensitivity level for each part of sensor. Just like with the MPU-6050, that nominal sensitivity level is the setting in which there is a minimal noise impact on the data but the data still has an acceptable level of sensitivity. The LSM9DS0 has a low power mode which similar but different than the MPU- 6050s low power mode. Instead of having four fixed modes, the sampling rates of the accelerometer and magnetometer are adjustable to reduce their power consumption. For the accelerometer there are 10 available sampling rates to choose from, not including a powered off mode which draws little to no current (the data sheet does not say). The magnetometer has 6 sampling rates to choose from; once again not including when it is in its powered off mode. Because of the large variety of sampling rates and no listing of their respective current usages, testing will have to be done to find the optimal low power state while still transmitting an acceptable level of data. The self testing functionality of the LSM9DS0 allows the IMU to test the functionality of its accelerometer and gyroscope while the device holds still. It then outputs the results of the test in which the microcontroller can read and compare with the known values of what the output should be. Just like with the MPU-6050, including the self test is vital for reading data from the LSM9DS0 for reliability of the device. The interrupt feature can be useful in that it stops the transmission of data between the device and the microcontroller. By not having the IMU consistently communicate to the microcontroller; it will lower the power usage of both devices. It can also be helpful when used for debugging. Since it allows us to enable and disable reading data from the device, thereby allowing us to see how each individual LSM9DS0s data affects the data after post processing.

2) Calculating Position from Data: Calculating position from the data created by reading the sensors is the major subtask of the project because, not only does it require the data received to be converted to acceleration with the correct units, it also requires that noise be filtered out of the system. Once that is completed, by using the kinematic formulas, we have to calculate position. Then, those calculations and filterings need to be tested for drift both when the device is moving and when it is not. Adjustments to the filtering and resetting will occur as necessary.

Because the unit of measurement for both IMUs is in units of g, this requires the received data to be adjusted by a factor of the average gravitation constant of earth. While this is not a difficult calculation, the decimal precision will have to be taken into account as it can introduce error in later calculations. Figuring out how many decimal places are required and how they affect the accuracy of the calculated positional data is another test that will have to run.

Because accelerometers and gyroscopes are naturally prone to noise, filtering is another main activity in the project. The main filtering technique we are attempting to utilize is geometric filtering. Geometric filtering is where multiple sensors are positioned in a known geometric configuration and the position between them is precisely measured and used and a constant for checking their relative positions. Because of the use of multiple IMUs, we can also use weighted filtering for our data, where the sensors whose measured data agree have a greater impact on the output of the filter than those whose measurement data disagrees.

The Kinematic equations state that it is possible to find a new position if a constant acceleration is held for a known duration and the initial position and velocity are known. There is a problem with this for our application, mainly that our acceleration is not constant. To circumvent this problem, we are going to treat the acceleration as if it were a constant value with a duration that is as short as possible. By keeping the duration between sampled acceleration values as small as possible, as that duration approaches zero seconds, it is almost equivalent to the instantaneous acceleration at that point in time. With this approximate instantaneous acceleration we can treat it as a constant for the purpose of positional calculation. However, if this introduces too much error, we can include jerk, the change in acceleration, into our equations. The jerk would be calculated by taking keeping the previous value of acceleration and calculating the difference with the new value of acceleration. This difference would be the jerk which would be used in the calculation of that axiss position using the previous values of acceleration, velocity and position. Also, as the amount of time between acceleration samples is very important for this to be implemented, a measurement algorithm will be required to accurately measure the duration of time between samples since crystal oscillators oscillation rates can vary due to the temperature of the crystal.

Once the data has been filtered and run through the kinematic equations, the data will have to be tested to ensure that it is producing a viable output. As such, the device will be moved in all three axes a set distance and then the data will be run through its conversion, filtering and kinematic formulas. That positional data will then be compared to the measured known distance that the device actually moved. Once that data is valid, the test will be repeated with the device at an angle and then finally while the device is rotating. If the data is acceptable, then the testing for movement is completed.

One of the main problems with accelerometers and gyroscopes is their tendency for noise. When that noise becomes really apparent is when the accelerometers and gyroscopes are not moving as they tend to indicate acceleration when there is none. Because of this, to thoroughly test both our filtering and calculations, when the device is not moving the positional data output must show that it is not moving. When the positional data that is output is acceptable, then the testing for when the device is not moving is complete.

As the probability that the device and algorithms survive both with and without movement tests, the coding and filtering for the device will go through many revisions to adjust the code for those cases. As such, beyond the initial creation of the algorithms, the revisions required to enable the device to pass both tests will take time. This activity also includes further optimization of the code as a result of later testing. As testing and modifying the code will be an ongoing activity for the remainder of the project, adjusting the code when we have further test case data will be ongoing throughout the semesters.

With every modification to the code after it has passed both tests, for optimization or other reasons, the device will have to be tested again both with and without movement to ensure that it still meets an acceptable quality of output data. Just like with adjusting the code in response to test data, retesting will be an ongoing activity throughout the semesters to ensure that we do not break what we have working and to provide additional test data which can be used for further optimization.

3) Test Accuracy : Accuracy is the main concern of the project as if it is not accurate enough while recording data, the data would be useless. To assist in ensuring or at least defining an operational range for the device, a series of tests are required. These tests include short and long distance testing, long recording time testing, testing at various speeds and finally field testing. After each phase of testing, the results will be analyzed and optimizations will be implemented if possible. As such, these tests will be repeated many times.

The small distance tests main purpose is to be a tumbling and odd environment test. This includes but is not limited to walking, skipping, hopping, climbing over tables, and spinning in a chair, stairs, running, jumping, hustle and waltz. These actions would be repeated over a measured known fixed distance. The effect on the accuracy of the measured fixed distance would then be compared to a smooth version of the device traveling the same distance to observe the error created by the erratic movement.

The testing of the effects of going long distances is mainly for observing the error created over a long period of recording time while in motion. This test would be performed by starting the device at a fixed position, and then traveling a long distance only to return the device to the same starting position. By starting and ending at the same point, we can see the amount of drift incurred by a long time recording data while moving without having to measure how long the distance actually is. This method may also be implemented in the small distance test if traveling a fixed distance creates problems. The quality of the data would determine based off of how far apart the starting point and the ending point is.

The long time test is an extended version of the no movement test. In this test, the device is left in a still location and then retrieved after the low battery warning goes off. This is to test the drift that occurs naturally due to noise. From the data gained from this test, the amount of noise from holding still should be more apparent as would its effect on the positional data.

To test the various effects that different speeds have on the accelerometer data, we would test it similarly to the long distance test, in that we would choose a speed to test the device at and then transport the device for a set period of time at that speed. Once that time was completed, the device would be brought back to the exact same starting position and the starting and ending points compared. From this data we could see the effect that traveling at high speeds has on the device.

The very last test to be done is field testing. As this testing is dependent on the device being decently secured and able to be bumped around, it is dependent on the housing of the device being either to the field testing stage, or, preferably, complete. In this test, the device would be given to a third party for them to test while doing training. After the training exercise was completed, they would return the device to the starting point. By doing this, not only could they see the path they took, but also the accuracy of the device. We would give them a questionnaire to fill out and record any interactions with them for the purpose of reviewing it at a later point. We would then take their suggestions, consider them, and implement the ones we deem possible and beneficial. However, once again, field testing is dependent on the completion of the housing for the device. As such, it will be done later in the project.

## B. Wearable

In order for the device to be easily used to record the path taken, the device must be able to be worn by the user. While it may start out being built on a breadboard, for it to be wearable it must run off of battery power. Also, the housing for the device components must be both durable and lightweight. Eventually, the device would have to be tested in the field it was designed for, search and rescue. However, to prevent it from causing problems if there are any malfunctions, it would first be tested during drills. For this feature to be realized, it must not only have decent battery life, but also be light and durable.

1) Build Components on breadboard: Almost every device that includes a microcontroller at some point in time is built on a breadboard to ensure that both the circuit and communication between parts work. This project will also require a breadboard proof to ensure the device is possible. For a better understanding as to what this requires, we shall look at each part to note not only what it requires in terms of wiring, but also what type of communication protocol it uses to exchange data with other parts.

The two IMUs that will be used include the MPU-6050 and the LSM9DS0. The MPU- 6050 requires a minimum of four wires to function, two for data communication, and two for power. A minimum of four or five wires is required for the LSM9DS0, with two or three dedicated to data communication and two for power. While both can be run off of 3.3V, the MPU-6050 can also run off of five volts and does not require the use of a logic level converter, unlike the LSM9DS0. I2C protocol can be used by both IMUs, however, SPI is another communication option for the LSM9DS0 which requires three wires for data communication, hence the variable minimum wire count for this type of IMU. For more information regarding communication methods, refer back to Reading Sensor Data the breakout board for the LSM9DS0 is the large IMU at 1.3" x .6" x .1" while weighing approximately 1.8 grams, compared to the MPU-6050 whose size is .8" x .6" x .1" and weighs 1.1 grams.

The logic level converter, which is used to change the Arduinos 5V output signals to 3.3V and vice-versa, in necessary due to the required voltages for both the Arduino and the LSM9DS0. Since they require different voltages to communicate, it becomes necessary to shift the voltage to the required values for them to properly pass data. The logic level converter does require that it be supplied with both 5V and 3.3V supplies, as well as ground. This means that just for the part to be able to be ready to use it requires 3 wires. That said, it is a relatively small part with a size of 0.63" x 0.52" x .1" and an approximate weight of .75 grams.

The micro-SD shield for the Arduino is designed to fit directly on top of the Arduino to save space. It also converts the Arduinos 5V logic to 3.3V logic so as to not burn out the micro-SD cards. Because it is designed to fit directly over the Arduino, it is one of the larger pieces at 2.125" x 2.375" x .1" weighing approximately 11.6 grams. However, it does provide a prototyping surface located directly above the Arduino which can be utilized to reduce the total required space for the device.

The Arduino microcontroller is by far the largest and heaviest part, but also the most important for the rapid prototyping process. It is used to control and communicate with all the other components of the device as well as acts as a regulated power supply. That being said, it does weight 28 grams and is large in size at 2.95" 2.1" x .33".

The total wiring weight is dependent on how many wires are used as well as what type of wire is used. The standard weight of a 20 gauge wire is 3.5 lb. per 1000 ft. [35], so if the weight of the wire becomes a problem we could switch to using aircraft wire which is much lighter at 1.1 lb. per 1000 ft. [36]. Therefore, we do have options to minimize the total weight of the wiring besides just using less wire in our device.

2) Battery Implementation: A portable system either has a power storage system to power the device for when it is not connected or a form of power generation. However, in order for a device to be wearable it also requires that the device is lightweight. Because of the size and added weight that a power generation system would cause, a battery power supply system is the most viable option. But, not only do we need to test how much power the device is using while running, we also have to research the required runtime in order to select the appropriate battery size.

The testing for how much power the device is using is a straightforward test. It involves running the device in its highest power usage mode, where each IMU sampling rate is at its maximum and it is consistently writing to the MicroSD card, and testing the current draw. This test will be repeated with lower sampling rates depending on how much error they introduce. The former result will be considered the worst case for power usage, while the latter would be the actual power usage. Both of them would be considered at room temperature.

In order for the device to best serve its target audience of first response emergency personnel, we need to know the required runtime for the device. This can be based off of the run time of other pieces of equipment, such as air tanks, which have a fixed runtime. Also, it can be based off of the upper average of the total time they are responding to a call. Whichever is longer will then be considered the lower limit for the required runtime.

With the data acquired from the previous two activities, the required amp-hours can then be calculated. Based off of this and with consideration of weight and cost, a battery can be selected from currently available options, ideally with a higher amp-hour rating than what is required. After the appropriate size battery has been selected, it will be added to the device, and a total runtime test will be done on the device to ensure it meets the runtime requirements.

3) Create Housing: Once all the components have been decided on and the device is working on the breadboard, the total size required by the components would be measured. Then, we would research the thermal limit, durability, weight, and cost of various materials for use in creating the housing of the device. Once a material is selected and its processing limitations are known, we can design the housing and simulate structural stress on the design. Once the housing design is acceptable, we would implement the design and finally secure the components to the housing.

The total size of all the components must be considered when designing the components housing. As such, both the dimensions of the components must be known, but also the weight of the components as to provide a near constant weight of which the housing will add to. The weighing of the components is to ensure that the final device does not weigh an unnecessary amount.

The purpose of researching the possible materials of which the housing can be made of is to produce a list of reference data which includes the maximum temperature, durability, weight and cost. This will then be referenced to during the process of designing the housing of the components in order to provide a sense of both how much the housing would weigh, but also how strong the housing would be.

In order to design the housing, it would first take into consideration the required layout of the components and which ones have to be accessible. Based off this information, the total weight of the components, and the list of different materials, a basic design can start being formulated. Once a basic structural design has been decided on, it then would be modeled using a CAD tool, such as rhino. Once the design has been modeled, it is ready for some simulated stress testing.

Once a CAD model of the device has been created, it enables us to import it into a stress testing simulation environment. One such tool to do this is a program called solid works. This allows us to know which location in the design is likely to break first and thereby fortify that location, and also see the effect different materials have on its structural durability. Once the design has been completed to an acceptable structural strength, weight, size, and cost, it then can go through two stages of implementation. The first being where the design is 3D printed to ensure that the components actually fit and that they work in the designs connection locations. Then, if all the components can be secured to the housing, wired together, and the removable components can be accessed, than the design will be sent out to be constructed out of the chosen material. Once constructed, the components will attached to the new housing and accuracy tests will be run again to ensure that the housing does not interfere with the operation of the device.

4) Field Testing: With the completion of the housing design and the housing for the components having been constructed, this enables us to do field testing with the device. This includes testing the housing strength, testing the thermal response of the housing, and giving the device to be worn by first responders during one of their drills.

Besides the simulated stress testing, we also have to test the actual structural integrity of the device. This would require assistance from the mechanical engineering department as the electrical engineering department does not have the required equipment to run the tests. With this in mind, if we can not get their assistance we will attempt to test the housing durability by other means, such as blunt force, dragging on the ground and dropping from various heights. If the housing does not meet what we deem as an acceptable durability, the housing will return to the design stage.

Because heat can modify the size and shape of various materials, it is important to look at how various levels of heat can warp and possibly weaken the design. This can either be done with the assistance of the mechanical engineering department or with the use of an oven. By repeatedly heating and cooling the device, it should show some physical strain and possibly warping. It will also show if and where thermal insulation should be installed to keep the components from overheating.

The final stage of housing testing is to give the device to actual first responders for them to test using it during a drill. This would require them to start the device at a known fixed location, and then, after the drill is completed, return the device to the same location. Then we would take the recorded data and show them the path they took. Also, this would demonstrate to them the accuracy of the device. After they have had a chance to experience the device and see its capabilities, we would like to get their input to be able to further improve the device, be it about the housing, the functionality, or the softwares user interface. To do this, we will ask them to fill out a feedback questionnaire, which will ask about not only its functionality, but also any features they would like to see implemented. By getting this feedback from the targeted user base, we can use their feedback to improve the design. However, with each attempted improvement or feature we add, we also have to re-run the accuracy tests and housing tests if there are any modifications to the device itself. The only case in which those tests would not have to be run would be for modifications to the software. That is not to say that modifications to the software do not require testing, instead it requires a different kind of testing just on

the software itself.

#### C. Stores Data Locally

Storing data locally serves two purposes. It allows for a log or recollection of the device so that it knows not only its current location but where it has been. It also allows for the bulk of data processing to occur after the device has been used.

1) Select file format: Selecting the file format is an important task because it will standardize our methods of communication. The first thing we would need to do is research some file formats that are available. Then we would proceed by testing how to write the chosen file type and what specifications go into that format. Selecting the format is an important first step. We will need to do some basic research on file formats that are currently used. Once the team has done their research about the file format, we will spend some time discussing what file format will be robust enough to save the sensor data. After the debate we will need to make a final agreement and move forward with that file format.

2) Connect SD shield to microcontroller: After deciding the file format we will need to connect our microcontroller to a Micro SD card. This will be done using a micro-SD shield. This shield attached directly to the top of the Arduino microcontroller.

Testing the connection from the microcontroller to the SD card is an important task. This will be our first time we will be using the equipment so we need to plan extra time in order to account for our beginner level experience with the hardware. It will be vital to test all aspects of the connection and the file making sure the file can handle data logs that become large and won't need the file to be split. C. Write code to save data to SD card After getting the connection to the SD card code for the micro controller is required to send and store the information on the SD card.

Validate file format Once the code has been written and the Microcontroller is interfacing properly with the secondary storage retesting the file is required. Testing the file being created for both the accuracy of the data and the accuracy of the file format.

The last test we need to do is ensuring that the U.S.B. device is working properly. Test for robustness of the device as well as ensuring it can work in conditions that the device will be operated in.

3) Test file compatibility with software: Testing the file that is compatible is depended on the software we write to interface with the file. This means that we must come to a beginning point of the software before we can test the connection between the device and the software.

4) Make a button to enable/disable data recording: One issue with this implementation is that the file may need to be finalized or not being written to before the SD card can be removed. This would stop any issue with file corruption from the ejection of a device prematurely

Adding in an on off switch would allow the device to be turned into recording from recording mode to passive mode and a light would indicate which mode that was set in. This could be done by using a multicolored LED, or using a flashing LED to indicate when the device is recording.

## D. Software

Software is an important feature of any device. It allows for the user to quickly get useful information from the device and use that information to make better decisions. The software we write needs to be basic enough for any user to use however complex enough to work in all situations. The software we write needs to be portable and quickly deployable on almost any type of machine.

1) Data Manipulation: First part of the software development cycle is too gathered from the device. This means that the software has to be able to read in the information weather it is from a file or, through a direct connection to the device. After gathering the information from the IMU the software will need to do some post processing on the data and calculate or filter out any data to get more precise measurements.

Inputting the data into the software requires that the device allows for the easy transfer of detailed data regarding past calculated positions. The most simplistic method to loading the data would be done through a file transfer. As we previously mentioned the device we build will have local storage that keeps a running log of position. Saving this file and moving it from the U.S.B. card to the software, is the first step of building the software.

Taking this information and calculating or filtering unnecessary data could be done to a deeper extent post processing. This ensures that minimum processing is done on the device to ensure the lowest power consumption.

Outputting the data is an important feature of any software, this allows the user to graphically view the data efficiently and glean the important details. By using a graphical interface, the use will be able to interpret the data without having to go through extensive training.

2) *Drawing:* While seeing the data represented by a series of numerical values would be useful, the software becomes invaluable, when it represents data graphically and illustrates the data as a map in 3 dimensional spaces.

Drawing the data into 3dimensional space is inherently difficult to facilitate this; the software package unity will be implemented to create the visualization of the data points. The first step is to draw all of the positional points without any connection.

After placing the dots in the 3 dimensional planes we will connect the dots with lines in order to show movement over time. This not only helps to show movement, it also helps to keep seperate runs identifiable.

3) Graphical User Interface: All programs need to allow for users to easily use the built in functions. Usually this entails the use of a graphical user interface. Allowing for users to select obvious on screen options to control their program will create a friendly user experience.

Because the data is represented in 3 dimensional form the user needs to be able to traverse the data moving up down left and right as well as rotating the camera. Another navigational tool that will be implemented is a menu system. Menus are vital to all software that allow for user input. They give the user an efficient logical interface to the software.

4) Debugging: One of the last steps after producing a working piece of software is to field test it. In order to field test it the software needs to go out to a group of people with varying technical abilities and ask them to complete a task monitoring how they work though the software and whether or not they are able to complete the given task. Taking this information into account you should make your final changes and move to the last step.

5) Deployment: Deploying software is the last portion of building but it is very important that your target your deployment to the platforms that you want to use. This means targeting windows computers and creating a simple installer so that the software can be quickly installed and run on any windows machine.

## V. TARGET DEADLINES

The following is a tabulated list of expected due dates and actual completion dates of the projects tasks. To avoid unnecessary filler and to assist in limiting the fatigue of reading a massive document, the tasks and their explanation is listed only once in the Task Completed in Fall 2014 and Tasks undertaken in Spring 2015 sections of this report. These deadlines were used to help keep us on track throughout the semester, and most of them were met. While the team did miss a few deadlines, because of the modularity of the tasks, explained in the Tasks undertaken in Spring 2015 section of this report, the project was not delayed very much if one feature was delayed.

Task	Due Date	Date Completed
Fall Assignment 1	9/8/14	9/8/14
Fall Assignment 2	9/15/14	9/15/14
Fall Assignment 3	9/22/14	9/22/14
Fall Assignment 4	9/29/14	9/29/14
Fall Assignment 5	10/7/14	10/7/14
Fall Assignment 6	10/21/14	10/21/14
Fall Assignment 7	12/1/14	12/1/14
Spring Assignment 1	12/8/14	12/8/14
Spring Assignment 2	2/9/15	2/9/15
Spring Assignment 3	3/2/15	3/2/15
Spring Assignment 3	3/16/15	3/16/15
Spring Assignment 4	4/6/15	4/6/15
Spring Assignment 5	4/20/15	4/20/15
Spring Assignment 6	5/4/15	5/4/15
Spring Assignment 7	5/11/15	5/11/15

TABLE II: Feature: Documentation Deadlines and Completion Dates

Task	Due Date	Date Completed
Read Accelerometer data Fuse Gyroscope Data Fuse Magnetometer Data Data Filtering Function for Position	10/1/14 10/2/14 10/21/14 11/25/14 2/11/15	9/24/14 11/10/14 11/17/14 11/17/14 12/28/14

TABLE III: Feature: Self-Referential Deadlines and Completion Dates

Task	Due Date	Date Completed
Record to Micro SD card	10/21/14	10/1/14
Implement Data Storage Protocol	4/20/15	4/18/15
New Search Session	4/18/15	4/20/15

TABLE IV: Feature: Local Data Storage Deadlines and Completion Dates

Task	Due Date	Date Completed
Camera Rotation	10/27/14	10/27/14
Menu	10/30/14	10/27/14
Exit	10/31/14	10/31/14
Load	11/6/14	11/6/14
Display	11/26/14	11/26/14
Render Points	10/31/14	10/31/14
Compile	10/29/14	10/29/14
Sample Test	3/5/15	3/14/15
Recorded File Test	2/18/15	2/18/15
Lines	11/6/14	11/6/14
Filter Data	3/15/15	3/30/15

TABLE V: Feature: Software Deadlines and Completion Dates

Task	Due Date	Date Completed
3D print housing Battery selection Performance Testing Integrate Device into Housing	4/9/15 2/9/15 4/18/15 4/12/15	2/11/15 2/9/15 3/22/15 2/21/15

TABLE VI: Feature: Wearability - Deadlines and Completion Dates

#### VI. RISK ASSESSMENT

With a project proposal comes a level of anticipated risk. The members of Team DIM have assessed to the best of their knowledge the potential risks they may encounter during the entirety of the course, categorized based on the impact on the overall project such a risk may pose. Based on these risks, individual mitigation plans were produced. Table 2 provides a Risk Assessment Matrix in which each risk has been classified based on their severity, as well as the anticipated frequency of their appearance.

### A. Severe

1) Lose Accuracy Due to Drift: In the event that our data loses accuracy due to the effects of drift, we will first check to see if it falls within the accuracy specifications we desire. If it does, then the problem becomes less severe, otherwise we will test and observe what would cause this drift and then instigate the integration of a drift filtering algorithm which would negate the effects of the drift based on the observed, tested and calculated average drift that is observed. This would be further mitigated by introducing a longer startup time in which the drift is observed by the device and adjusts the variables within a filtering algorithm to adjust its measurements so as to remove most of its effects. Also, by the inclusion of multiple IMUs either by the same or different manufacturers in a geometric pattern with known distances, the drift of the IMUs will be



Fig. 1: HMC-6343 Three-Axis Compass (By: Team D.I.M.)

in different directions with different values, but by using a relative positioning constant filter, the effects of the drift by the devices is reduced.

2) Design Unable to Meet Expectations: While it is still possible for the design to be unable to meet the expectations of the design idea contract, by having the contract be loosely worded and semi-vague, it is less likely that the design will not meet the listed expectations. However, in the event that it is unable to, a design idea contract change will be requested if and only if all other means or solutions have resulted in unacceptable results. However, in the event that it is unable to, a design idea contract change will be requested if and only if all other means or solutions have resulted in unacceptable results. This particular risk actually occurred during the first project term for Team DIM in the Self-Referential feature of the project. Upon discovering that the team struggled to get an accurate Yaw through the LSM9DS0 9DOF sensor, the team conducted research on devices that could provide accurate yaw. Of the devices found, the team decided to purchase the Honeywell HMC-6343 Three-Axis Compass.

This particular risk actually occurred during the first project term for Team DIM in the Self-Referential feature of the project. Upon discovering that the team struggled to get an accurate Yaw through the LSM9DS0 9DOF sensor, the team conducted research on devices that could provide accurate yaw. Of the devices found, the team decided to purchase the Honeywell HMC-6343 Three-Axis Compass.

3) Positional Measurements Unavailable: The effect of positional measurements being unavailable is being reduced by using the kinematic equations so as to enable position to be calculated from either velocity, acceleration, or jerk measurements when the increment between measurements is known. While this can introduce some error, the error can be reduced by appropriate filtering algorithms.

4) Overworked/Exhaustion: The effects of being overworked to exhaustion is being reduced by including time for a minimum of two hour naps while pulling all-nighters and using stimulants (mainly caffeine). The risk of this is also being reduced by including a day within our schedule to not work on senior project that day. In order to reduce the probability of this occurring, the weeks tasks will be assessed following the course lecture on Tuesdays, with a timeline for what needs to be completed throughout the week, in order of priority. These tasks will be distributed to team members according to their current progress on their currently assigned task, as well as based on their expertise and familiarity in the area. The Gantt chart will assist in ensuring that the team does not fall too far behind, and provides quickly available information for upcoming milestones and deadlines.

5) Lack of In-house Skill: The risk of having a lack of inhouse skills was first mitigated during the creation of the team as our members have a wide variety of skills and experience. However, in the event that this occurs, two members of the team will be assigned to learn the required skill so as to enable more than one person to have that skill later on if one is incapacitated.

### B. Major

1) Feature Exceeds Time Quota: The corrective action for this risk is dependent on which feature exceeds the time quota. Since calculation of the correct positional data is the krux of our project, exceeding the time quota would the semester time to do the project. So there is no resolution or alternative to this feature. Conversely, on a less critical feature, the feature might be omitted base upon factors such as time left in the semester and the criticality of that feature. An exact resolution cannot be articulated since the current project status and time remaining are unknown.

2) Software Incompatibility: If software incompatibility occurs, we will need to port our algorithms to the new language. This scenario is if we transition from the arduino to a more powerful micro controller to process multiple IMU data. The impact is solely related to time required to debug converted code.

# 3) Communication Breakdown: Communication breakdown

is intrinsic to every group project. Since a break down will always occur at some point, once it is discovered the group must briefly examine the assigned task documentation along with meeting notes and figure out where the communication went wrong if possible. The damage if any must be assessed followed by a solution to repair any damage from the breakdown caused. Finally the repair must be implemented.

4) Insufficient Hardware: If the currently selected hardware configuration is insufficient then a more appropriate alternative must be ordered and implemented in a timely manner as to not .

5) Delay do to non-project priorities: These delays will inevitably occur. Most are easily solved by communicating to teammates your current status and how long the delay will impact your particular portion. This must then be measured against any dependencies that exist as a result of the delay. Another team member may have to increase their workload to compensate for the delay.

## C. Medium

1) Hive Down: From team members experience in a previous class that used the ECS Hive service, it was not uncommon for the ECS Hive to experience downtime. The Senior Design course uses the Hive as a content management system as well as a submission portal for course documents. In the event connectivity to the Hive is unavailable, all of the assignment prompts and lectures have been uploaded to the teams Google Documents server for access. If the Hive is unavailable during a submission period, each teammate will attempt to submit the assignment in the event that the issue is client-based. If the issue is server based, the instructor will be notified, and provided a copy for review via e-mail before the hand-in deadline, with all group members CCd. To best mitigate this issue, assignments and weekly reports should be completed and submitted by the end of the previous day.

2) File Structure Exceeds Filestructure Limitations: The FAT32 file structure has an individual file limitation of 4.0GB per individual file. This initially posed a concern to the group because of the uncertainty of length for individual rescue operations. Because of the large amounts of individual positional datum of insigificant size, it would require a significant session of continuous data logging to exceed this filesize. To accommodate for this potential risk, field technicians who operate in first response emergency rescue will be contacted to provide information regarding the average time of rescue in specific circumstances, in order to ensure the final product meets these needs without imposing on the hard limits of the filestructure, potentially by segmenting long files.

3) Interface Unintuitive: It is not atypical for software user interfaces to be designed in a manner that is clunky and difficult to use for the end-user. This is usually the result of engineers being wrapped up in their work to the point of having complete familiarity, only to be unaware of the amount of time they invested to develop the prerequisite knowledge to operate the device as intended. To ensure that the final deliverable does not become so difficult to use that the target demographic disregards it, the development team will communicate with field technicians and garner feedback from mockups to see what sort of designs are usable. In the event that usability or key components of the featureset are compromised, a reasonable middle ground along with wellwritten operation documentation for the end user operation should be the solution.

4) Financial Burden: Due to the boom of maker culture in the past decade, easy-to-use sensors and microcontrollers are available from a larger array of vendors at extremely competitive prices. Devices that have very specific uses typically have higher price tags, of which may not fall within the realm of reason for Project DIM and Senior Project. In the event that a part is required but is a significant financial burden, team members will assess the cost of the feature, versus the significance of the feature to the core deliverable. The team will then assess and brainstorm ways in whether this feature could be realized using separate components of more reasonably priced hardware, and attempt to implement the same feature by providing communication and logic between the replacement devices.

5) Ideal Sampling Rate Unattainable: Sampling rate is important to the overall accuracy of the device because a larger provides more samples which allow for better precision and accuracy. Because a significant amount of data will need to be logged, an external card is necessary. File write operations require an fileOpen before each write begins, and a fileClose following each write. In the event that the sampling rate exceeds that of the write speed, data loss will occur. To mitigate this, extensive testing will be done to find a storage medium that allows for fast write speeds without running into the loss of data. This test will run into the positional data, to see what sorts of impacts a lowered sampling rate has directly on the algorithm developed. Faster class cards will be purchased in order to see if this has any sort of impact on read/write speed. In the event hardware cannot mitigate the issue, research will be done on more efficient software SD write algorithms, with further implementation based on the results from the research.

## D. Minor

1) Team Member Absence: In the event that a team member is absent we will continue working without them. In case of team votes there vote will be counted as abstained making a two person majority all that is needed. If an extended absence is required then we will make plans and divide the work assigned to that member to the rest of the team and do our best to incorporate them in any votes we are having during their time away.

2) Power Consumption Concerns/Expectations: Because of our inexperience with using devices powered by battery we have included extra time and resources dedicated to getting the device to work on a battery power supply. We are predicting that there will be issues either getting the power supply to work and function properly with the device. We have dedicated extra time in order to complete this task as well as extra funds predicting that we may ruin one or two things while working with the device. We have also considered the fact that we may get done sooner than expected and if that is the case we will add extra tasks to the design requiring that we create a breadcrumb trail back to the starting location.

3) Equipment Loss: While this issue could be caused by many different things such as a car break in or even equipment being misplaced we have a simple solution to buy a new one. We will have to take into account the shipping time of most of the equipment we buy and delay sections of the project possibly moving ahead of tasks we are on and attempting to predict their outcome in order to work on subsequent tasks.

4) Personal Computer Failure: Similar to the issue listed above with a similar standing policy with a slight twist. We all have personal computers and because of that we would be able to use the machine that was given to us and already in senior design lab to work on in the case that our machine failed. Additionally, collaborative documentation work is started in Google Documents. Completion of any Project DIM-related work will be uploaded to Google Documents, as well as the ECS Hive if it is required for submission. Completed code is uploaded to the BitBucket account, with proper documentation attached to each users commit.

5) Data Hosting Downtime: We have decided to back up all of our online data in order to minimize the possibility of this issue. Since our data is hosted on both Google drive and on the hive we see a very low possibility that we would be unable to get to any of these services in order to get our information

VII. TASKS COMPLETED IN FALL 2014

Task	Alex	Andre	Joey	Yan	Total Hrs
Read Accelerometer data	32	23	11	18	84
Fuse Gyroscope Data	19	33	6	17	75
Fuse Magnetometer Data	3	24	0	24	51
Data Filtering	13	20	7	15	55
Function for Position	37	34	9	17	97

TABLE VII: Feature: Self-Referential Fall 2014 Completed Tasks with Hour Allocation

Task	Alex	Andre	Joey	Yan	Total Hrs
Record to Micro SD card	1	3	6	23	33

TABLE VIII: Feature: Local Data Storage Fall 2014 Completed Tasks with Hour Allocation

Task	Alex	Andre	Joey	Yan	Total Hrs
Battery Interfacing	6	0	0	0	6

TABLE IX: Feature: Wearable Fall 2014 Completed Tasks with Hour Allocation

Task	Alex	Andre	Joey	Yan	Total Hrs
Camera Rotation	0	0	8	0	8
Menu	0	0	15	0	15
Exit	0	0	9	0	9
Load	0	0	21	0	21
Display	0	0	20	0	20
Render Points	0	0	18	0	18
Compile	0	0	7	0	7
Sample Test	4	0	2	0	6
Recorded File Test	2	1	3	10	16

TABLE X: Feature: Software Fall 2014 Completed Tasks with Hour Allocation

#### VIII. TASKS UNDERTAKEN IN SPRING 2015

The intention behind the proposed featureset was to promote modularity in design, and to allow for team members to work within their areas of expertise. As such, Yan Lin and Andre Julien focused primarily on the device interfacing on the AtMega328 platform. This particular aspect of the

Task	Alex	Andre	Joey	Yan	Total Hrs
Assignment 1	15.5	15.5	15.5	15.5	62
Assignment 2	10	24.25	16.5	30	80.75
Assignment 3	19	5	58	0	32
Assignment 4	14	7.5	1.9	17.5	40.9
Assignment 5	2	1.5	8	12.5	24
Assignment 6	80.5	116	101	115.75	413.25
Assignment 7	23.5	29.5	22	52	127

TABLE XI: Completed Course Assignments Fall 2014 Individual Member Contributions

self-referential feature involved the selection of sensors, and establishing a kinematic model. Additionally, this team also implemented the Local Data Storage feature due to it being the key component to the integration of both hardware and software aspects. Due to his expertise within the Unity3D software development kit, Joseph Youngblood was assigned to the entirety of the visualization software component. The team assigned to Self-Referential had to work closely with Youngblood in order to maintain parity data generated from the hardware and its respective interpretation on the software front. Having a strong background in electronics engineering along with past experience with both designing and fabricating 3D printed components, Alexander Meadows was assigned with handling the Wearable feature.

### A. Problem Statement Revision

Revising our problem statement report to show our further understanding of the social problem we are solving. This was broken into two parts, research and writing. The task of finding and summarizing sources was given to Joey and Alex, while the main task of writing was given to Yan with support from Alex. This is the only task assigned to Yan this week.

# B. Device Test Plan

This assignment tasked us with writing a report about our test plan for our device. The report was broken down into four main parts, test plan writing, metrics, timeline, and review. Yan and Joey were assigned to write the test plan (how the device or software is going to be tested). Yan was assigned this part due to his writing skills, while Joey was assigned because of his familiarity with software testing. The creation of metrics and their explanation of why they are reasonable was assigned to Andre because of his down to earth nature. The timeline portion of the report was assigned to the current team leader Alex because it falls into the realm of organization and planning. The final review of the report is to be performed by all members of the team to provide the highest probability that any typos will be caught.

## C. Market Review

The market research report tasks us with not only understanding and presenting what market and which consumers would purchase the device. Also, any competition and related pricing information needs to be researched and detailed to show what market the device would enter if it is to go to market. Using the information gained from our research we will then put together a plan for how to market our device. This information will be presented in the report as well as orally in the presentation.

## D. Mid-Term Progress Review

Is the Individual Feature Presentation. This assignment has both a written and an oral component. The assignment will start, with coordination of our lab instructor, selecting one of our design idea features to discuss. We will then write on the suggested topics from the assignment 5 documents and put that information into our specific Latex document which will format it into IEEE format for us. Lastly we will prepare a verbal presentations.

#### E. Feature Presentations and Reports

The internal team meeting consisted of a short discussion of where the team is currently and what needs to be done this week beyond the usual weekly report. We talked about the current status of using the DMP built in functions to calculate step rather than using our own post-processing function.

### F. Deployable Prototype Review

Prepare our team Deployable prototype presentation. Taking a lot of information from our previous presentations adding in more quantitative analysis to our presentations according to professor Tatros request. Practiced our presentation as individuals and as a group to ensure that it flowed and was best quality. We did multiple full walk through demos to make sure that the device worked as promised

#### G. End of Project Documentation

Final documentation is the last set of documentation we turn in and is due may 4th and encompasses our entire project. We will take a lot of information from our first end of term documentation to avoid re writing a lot of sections. This should keep the work down but this document will take 2 weeks to finalize and will be a group effort to finish.

Task	Andre	Alex	Joey	Yan	Total Hrs
Assignment 1	6.1	10	15	10	41.1
Assignment 2	11	15	12	12	42
Assignment 3	13	14	6	13	46
Assignment 4	08	14	17	23	62
Assignment 5	17	6	15	31	69
Assignment 6	12	8	11	20	51
Assignment 7	24	17	25	30	96

TABLE XII: Feature: Documentation Spring 2015 Completed Tasks with Hour Allocation

## H. Self-Referential

1) Sensor Selection: In order to have a means to generate displacement data, the motion of the operator needed to be analyzed. The key component of the hardware device aspect of the Dark Inertial Mapping system was the utilization of MEMS

Task	Andre	Alex	Joey	Yan	Total Hrs
Display	00	00	25	00	25
Lines	00	00	15	00	15
Compile Code	00	00	14	00	14
Filter Data	00	00	14	00	14

TABLE XIII: Feature: Software - Spring 2015 - Completed Tasks with Hour Allocation

Task	Andre	Alex	Joey	Yan	Total Hrs
Implement Data Storage Protocol	00	09	00	11	20
New Search Session	00	05	00	00	05

TABLE XIV: Feature: Local Data Storage - Spring 2015 -Completed Tasks with Hour Allocation

Task	Alex	Andre	Joey	Yan	Total Hrs
3D print housing	20	00	00	00	20
Battery selection	15	00	00	00	15
Performance Testing	18	00	00	00	18
Integrate Device into Housing	18	00	00	00	18

TABLE XV: Feature: Wearability - Spring 2015 - Completed Tasks with Hour Allocation

devices. The MPU6050 and the LSM9DS0 were primary candidates for the selection. The LSM9DS0 was the primary sensor used in the Fall project term for the initial kinematic model. Additionally, the Honeywell HMC-6343 compass was included to address a deficiency the team had regarding heading. The MPU6050 had a built-in Digital Motion Processor that allowed for minimal overhead in generating metrics such as yaw, pitch, roll, as well as a pedometer step counter. Both Yan and Andre spent an extensive amount of time attempting to implement this feature, but were unable to meet the internal deadlines set by the team to get this properly working. As such, the final implementation uses the LSM9DS0 in conjunction with the HMC-6343 to generate data capable of producing both a step and a direction. This task encompassed 100 hours during the Spring project term.

2) Establish Kinematic Model: The original kinematic model aimed to double-integrate acceleration data detected by the accelerometer. Using fused data from both the gyroscope and the magnetometer, the team would be able to detect the overall displacement of the device when moved. However, this implementation has well-known problems that is the result of accumulated error. Accelerometers are inherently noisy devices and as such the integration of minor acceleration readings over a long period of time lead to a large amount of unintended displacement. Additionally, because the Z axis of rotation having no cardinal reference such as gravity, rotations are prone to drift. Due to all of the aforementioned issues along with the LSM9DS0 providing non-uniform heading values from the stock vendor drivers, the team opted to change kinematic models between project terms to address the design deficiency. As such, the team utilized PDR, using the same MEMS components, but interpreting the data provided in a different way in order to still meet the key feature set enumerated in the Design Idea Contract. This task was

addressed by all members of the group, culminating in 300 hours across two project terms.

3) Interface Devices: Test sketches were analyzed in order to verify functionality of the individual components. To adhere to the proposed kinematic model, this was the preliminary task that was undertaken to ensure that any further development could continue on the hardware aspect of the device. This task primarily involved writing an Arduino sketch that was capable of reading sensor data, and writing to an external storage interface. While minor adjustments and sensor selection differed during the two project terms, the majority of sketches produced originated from the original codebase presented during the Breadboard Proof assignment. In the spring project term, this was revised to include start and stop functionality to enhance overall user experience. Overall, 50 hours were required to complete device interfacing.

4) Device Testing: Details regarding device testing are detailed in the latter half of the documentation and were the emphasis of project-oriented work hours during the Spring 2015 project term. To meet internal metrics, 250 hours were cumulatively spent by all members of the group towards the extensive testing and improvement of the device given test results.

#### I. Local Data Storage

1) Implement Data Storage Protocol: With the primary focus of rapid prototyping, the default Arduino SdFat library was used in order to store sensor data onto an external storage medium. This task was completed prior to the breadboard proof and was completed by Yan Lin in 20 work hours.

2) New Search Session: In order to allow multiple files to be stored on a single device, and prevent new search sessions from overwriting previously recorded data, a mechanism was implemented detect existing files within the file system and name new search sessions in ascending order This task was completed by Yan Lin in 5 hours.

## J. Software

The software implementation will make use of the local storage feature, in order to generate a visual realization of recorded positional data. This task was allocated 48 total work days in order to complete all individual sub tasks required for full realization of the software interface feature, but does not include the time spent testing complementary functionality of the device.

1) Camera Rotation: Using Euler angles we will rotate the camera or the field of view that the user gets while looking through our program, allowing for the arrows to represent commands of lateral movement then using the mouse to represent pitch and roll. Commands will be very similar to a video game allowing users who are experienced with video games to quickly figure out the software. Furthermore we have pre-written script to handle the movement that we can quickly import and use with our code. This task was completed within the allocated time of 30 hours.

2) Menu: A basic Menu system will allow the user to select through familiar options to quickly load data and select other options typical to most other software packages. Basic menu to include options such as change the color of the lines as well as the individual points that are being generated. other options include loading images in the background to overlay the data on top, and options to load other csv files generated from the device. This task was completed within 20 hours of work in the fall semester.

3) Exit: An important feature sometimes forgotten about being able to properly close the program. This task was assigned and completed in 1 work hour in the fall semester.

4) Load: A basic feature to load the data that was saved from the device. Loading will take place over USB but the file must be saved locally before it is loaded otherwise we can run into errors after the card has been ejected. This task was completed after 16 hours of work in the fall semester.

5) Display: Displaying the data is a simple feature after it has been loaded to load the data into a three dimensional plane. Creating points as small 3d game objects creating them at specific points x,y,z. Upon completion of the interface, rendering of the visual component will be implemented. This will be accomplished by using the positional data generated by the device that is written to portable storage. This task was completed within 40 non consecutive hours fall and spring semester.

6) Draw position point on 3 axis space: As described previously, to draw the points, 3D game engine objects will be created, and rendered at the positions specified in the CSV file. This task was completed within 15 hours in the spring semester.

7) *Lines:* Connecting points with lines will require use of another gameobject to connect the small points. This gameobject will have a user selected color and will require some extra logic when laying down its point. This task was completed in 8 hours in the spring semester.

8) Compile Code: To make use of the logic written in high level language, the code must be Executable on an operating system. To accomplish this, the high-level code must be free of bugs, and produce expected logic outputs, given a set of inputs. Compiling the code is the last step and is an important feature because it is where we select our target platform. During this task, code will be parsed for compilation error, to ensure intended operation of individual functions. The development platform selected to render graphics will be robust enough to easily switch between web/browser deployments, all the way to deployment to an Android or iPhone device. In order to develop a software environment that is stable, extensive bug testing must be performed. The following list comprises of the tasks that result in the full realization of a stable environment. This task was completed in 10 hours total for full hand debugging to generate a working executable to be used by other members in the spring semester.

## K. Wearability

1) 3D print housing: For the deployable prototype to be ready for deployment, an external housing was fabricated to

protect the device from external forces. This case underwent multiple revisions to switch from a waist-mounted apparatus to a hip mounted one. Given his experience in the Rhino software, Alexander Meadows assumed responsibility for this task and provided 70 hours toward the design and printing of all enclosures.

2) Battery selection: The initial design of the product interfaced with the AtMega328 serial terminal for debugging purposes. This required that the device be connected to the USB interface when in operation. To undergo field testing, the device required an auxiliary power source. Alexander spent 15 hours surveying multiple batteries and performing timed tests to assess the quantitative performance of 3 off-the-shelf 9V batteries.

3) Integrate Device into Housing: Integrating the test platform hardware into the printed enclosure required time to re-assemble, re-wire, and thoroughly test basic functionality operation of the device outside of the test environment. This was initially performed by Alexander Meadows (30 hours), but was taken over by the remaining team members (10 hours each) prior to the Deployable Prototype Review.

4) *Performance Testing:* Performance testing details are documented further on in the document in the Wearability section of Hardware Test Plans and Results. These were completed by Alexander Meadows and encompassed 150 work hours.

### IX. USER MANUAL

This manual will provide the user with an overview of how to setup and use the currently implemented hardware and software features of the prototyped device. The setup and operation instructions provided are predicated on the use of the same hard and soft components listed. For hardware and software to work as designed the steps in this manual must be followed in the order presented. This user manual will to cover any any Windows Operating system setup or configuration guides please contact Microsoft customer service with OS related questions.

## A. Device Usage Overview

The dark inertial mapping device is intended to map one or more user(s) traveled path within a 2 dimensional plane while attached to the user's person. The mapping data then may be extracted from the device via a microSD card and docked with a computer running the accompanying visualization software. From the mapped paths displayed on the visualization software, user(s) can extrapolated the path(s) traveled by the device wearer.

## B. Limitations

The limitation of this device are enumerated below but not limited to:

*1)* Accuracy: At 90 meters traveled the device will generate and error greater than or equal to 5% deviating from the true path by 5 meters from the radius of the 92 meter recorded point



Fig. 2: Device Interior (By: Team D.I.M.)

2) Usability Conditions: This device is intended for use in a human survival environment were breathing apparatus are not required. For usable data the device should be used in and area less than 50 meters such as a medium to small structures or areas.

3) *Run-time:* The run-time of the device has been tested to exceed 9 hours of powered or running use using at minimum a new Energizer Lithium battery. [37]

4) Memory Card: The Micro SD card has a maximum of 100 logged runs. After 100 runs, the individual runs must be deleted from the SD card for data gathering to continue. The limitation is file names not their respective size. The file size limitation correlates to the storage size of the Micro SD card selected.

### C. Hardware Usage

The order of the items enumerated in this section are also the operation order. For example the MicroSD card must me installed into its respective shield on the device before it is powered (Battery is connected).

1) Memory Card: The SD card must be inserted or removed only into the second shield Fig. 2 when the battery is disconnected. The SD card must be firmly inserted (a locking click will be heard). For extraction the end of the SD card must be pressed towards the the non-open end of the housing (an unlocking click will be heard). The SD card is now ready for insertion into the card reader and the associated computer containing the visualization software.

2) Battery Install Instructions: The battery must be connect to the leads shown in Fig. 2 then placed top of the SD shield (large flat side of the battery down) and must be wrapped in and appropriated insulating material. Verity that the red LED is on signifying the device is ready to begin data recording.

3) User Attachment: Once the battery has been installed the device housing must be closed. The orange bottom plate on the left side of Fig. 2 must be inserted through backing plate shown in Fig. 3. Then the colored securing bands orange and blue must be wrapped around the device. The device is now



Fig. 3: Device Strap Back (By: Team D.I.M.)



Fig. 4: Device Active Front (By: Team D.I.M.)

ready for attachment to the user. The device <u>MUST</u> be secured to the users right leg using the elastic Velcro straps between the knee and the inseam. Preferably clothes with non-smooth textures should be worn.

4) Data Gathering Activation: The device has an exposed LED which indicates the device status along with a button to enable and disable data recording. When the LED is solid red Fig. 4 the device is ready to record. When the LED is flashing rapidly that indicates the device is recording data. Once the device is on (battery attached) and the LED is solid red, depressing the button will begin data recording. Pressing the button again will disable data recording and create a new file on the SD card with the file name number one greater than the previous.

## D. Software Usage

The host PC must Microsoft Windows 8.1 or greater. The supplied DIM.exe should be place in a folder on the computer where the visualization will be viewed. All imported run data will need to be placed with in the same folder as the



Fig. 5: UI Functionality Navigation taken by team D.I.M.



Fig. 6: UI Functionality Run Color (By: Team D.I.M.)

executable. Insert the card reader into the PC and transfer the run files to the folder containing the executable. Double click on the DIM.exe and launch the program.

UI navigation is as follows:

- See Fig. 5
- 1) Click the Icon then specify the settings you want to run on.
- Move the data .csv files into the same directory as the program.
- 3) Select the file from the select file button in the same directory.
- 4) Select the color you want to use in the top right Fig. 6
- 5) Right click to select where you started.
- 6) Set your scale and step size as well as your offset if needed.
- 7) Select Pedometer Function.

Controls:

- Camera movement can be controlled by both directional arrow KEYS and W, A,S,D keys (screen relative.
- Scroll wheel moves the camera stickily in the z axis up and down.
- CTRL key locks the inputs not allowing for any input.
- Mouse movement turns the camera.
- R key resets the position that you are starting from to (0,0,0) the center of the map.
- Right click anywhere on the map to set the starting point of the graphing the data.

Buttons:

- Clear clears the screen.
- Pedometer uses the pedometer function to claclulate the persons location

- Load Image loads the image selected from the select File
- Calculate uses a more robust method to calculate the persons position
- · Select File: Opens up the select file feature

Options:

- Scale the scale of the squares IE 1 is 1 meter by 1 meter.
- Step Size the size of one calculated step.
- BG Length Length of the underlayed image.
- BG Height Height ""
- Offset If this is = -1 it will find the offset from the file else it this will be the offset to subtract from the data

## E. Hardware Components

The device hardware components are as follows:

- SparkFun RedBoard DEV-12757 ROHS.
- SparkRun 9 Degrees of Freedom IMU Breakout -LSM9DSO SEN - 12636 ROHS
- SparkFun HMC-6343 Honeywell Three-Axis Compass.
- 1 MicroSD Card 4GB.
- MicroSD Shield Retail RTL-09899 ROHS.
- Wiring and connectors.
- SparkFun Logic Level Converter Bi-Directional BOB-12009 ROHS.
- Computer able to run Windows 8.1 or greater.
- MicroSD card reader for a Windows PC running Windows 8.1 or greater.
- Custom 3D printed enclosure.
- Velcro elastic straps.

## F. Software Components

The below list is pertains to the components used to generated the device functionality and UI environment.

- Arduio IDE.
- Windows 8.1 or Greater.
- Unity Game engine 4.5 or later.

### X. DESIGN DOCUMENTATION

During the Fall project term, Team DIM placed a heavy emphasis on the implementation of the Self-Referential features. By determining the kinematic model at an early stage, a visualization software prototype was drafted with the intention of use for both final system implementation, as well as validation of sensor data. Selection of the AtMega328 microcontroller platform was primarily for the intent of rapid prototyping. The team acknowledge the limitations of such a device, prepared for any changes in deployment platform if deemed necessary. For the course deadline of the bread-board proof, the team created a hardware sketch for the AtMega328 platform capable of capturing sensor data, and saving it to a local storage medium, while internally maintaining a steady sampling rate. This fulfilled a majority of the Local Data Storage feature at an early stage, and allowed a majority of the first project term's prototyping efforts to be focused on the Self-Referential and Visualization Software aspects of the

proposed system. In order to generate a positional estimation from human movement, Team DIM purchased a number of IMUs to better understand the capture of movement in discrete time, and to test with a selected microcontroller. The first two sets of IMUs ordered consisted of the Invensense MPU-6050, and the Sparkfun distribution of the LSM9DS0. The MPU-6050 is a cheap, popular IMU among microcontroller enthusiasts, whereas the LSM9DS0 was slightly more expensive, but contained better documentation, and an example code repository to start from. Based on the ease of setup and the amount of available documentation, the team opted to continue forward with the LSM9DS0. As a requirement for the first technical milestone, the team was tasked to present one significant feature of the prototype. Between the four features, the team narrowed the demonstration candidates down to Self-Referential and Local Data Storage. It was at this time the team arrived at its first real standstill: half of the team voted to present an element of the self-referential feature by way of manipulating the recorded sensor data, whereas the other 2 members wanted to also incorporate local data storage. The dispute was resolved by coin flip, and in retrospect, pursuing local data storage was the correct path to take. The team ordered 4 Arduino microSD shields in order to easily interface the storage device to the microcontroller through the Serial Periphery Interface (SPI). For the first milestone, Team DIM presented the manipulation of captured acceleration data in the serial terminal as velocity, which was then stored in a Comma-Separated Value file format to the microSD storage device. A sampling rate of 20Hz was attained during the demonstration by utilizing the millis() function within the Arduino library to create a fixed cycle length for each iteration of the logic loop in the microcontroller code. Following the breadboard proof, the team opted to employ a division of labor, and assign each member with an aspect that could lead to the realization of a feature. Joseph Youngblood was tasked with the creation of the visualization software whereas Andre Julien and Alexander Meadows would focus on selfreferential positional displacement generation, leaving Yan Lin to improve the local data storage feature. Leading up to second milestone of the Midterm Technical Review, the project made good progress with all tasked features up to the completion of sensor fusion. The visualization software was developed through the Unity Software Development Kit, a graphicallyoriented game engine. The software was capable of loading a CSV file, and plotting the calculated displacement on the virtual plane. This plane could be navigated through a flycamera, that ould allow for the repositioning of the view of the plotted positional displacement. Similarly, the SD write code from the first milestone was cleaned up and repurposed from a code segment into a function. This function was tested in a number of sketches, and successfully generated a CSV file with the desired parameters. Due to an oversight on the Breadboard Proof milestone, the team was able to optimize the millis() comparison logic, and use micros() for better precision. As a result, sampling rate was improved to exceed 20Hz, but its bounds remain to be tested, pending the finalization of the self-referential algorithm. The greatest struggle came to the implementation of the Self-Referential feature. Initial

tests to validate the positional generation were performed through a slide test, collecting acceleration data across one of the sensors axis on a float surface. Following the law of Newtons third law of motion, in order to detect a stop, the device necessitates an equal force in the opposite direction. To implement this, the Arduino sketch included an accumulator value that added all mono-directional accelerations up to the detection of an opposite force. When the force of an opposing magnitude was detected, it applied the accumulated value to the velocity value in order to differentiate between a true stop state, and a situation where velocity is constant, resulting in zero acceleration. Figure 10 depicts the sliding test, showing the results of the added compensation. The device always starts at the origin, which is denoted by the light blue vertical stripe. The device was moved along the positive X axis, stopped, then moved along the negativeX axis. Vertical shifts occurred during these slide tests and were caused by the sensitivity in the Y axis. Oftentimes the slide test would be stopped but due to a non-ideal stop motion caused by the tester, this would result in accelerations read in the Y axis. Due to an unforeseen circumstance regarding human gait movement, the displacement algorithm was unable to generate acceleration for walking patterns. Despite having success with sensor fusion to generate fused data necessary for compensating for change in orientation of the device along the X and Y axes (pitch and roll, respectively), the team was unable to produce a solution to generate a reliable change along the Z axis (yaw). Due to the concern for a lack of progress in this regard, the team opted to undertake a mitigation plan involving a change in kinematic model to a pedestrian-based approach utilizing a dead-reckoning algorithm to generate displacement. The initial hardware sketch produced to assess the viability of such a mechanism proved promising. Given the middle row of tables around inside Riverside 3001 spanning 9.144 meters by 3.048 meters, a path was successfully traversed around, resulting within one step of the starting distance. While initially distance traversed was a concern, the fact that the algorithm properly returned to the starting position, just as it had empirically was a positive step forward for overall progress. Using this new kinematic model, the team would progress into the Spring project term where a focus was placed on testing the functionality of the device using internal metrics. Additionally, to progress toward a deployable prototype, Alexander Meadows began work on the device enclosure which would allow the wearable feature to be realized. Using the Rhino3D software, an initial enclosure was created to fit the size of the device, with ample room for affixing the sensors as well as a battery. During this time, device testing took place. The team divided the labor to maximize test efforts with an emphasis of the tests being placed on feature functionality. Based on the primitive nature of the visualization software at this point, a number of changes were made by the hardware test team to ease the interpretation of data. These requests eventually led to the implementation of a number of key features that contributed to end-user usability. During short test runs, it was easy to distinguish discrete steps. As paths increased in size, discrete step objects denoting paths would overlap making it difficult to determine if a path turned when it intersected another path. Additionally, multiple search paths were difficult to distinguish from each other as loading path data by default rendered as a red step object. These were remedied with the addition of step objects to connect consecutive step objects, as well as a swatch palette to assign colors to individual search paths. This level of clarity expedited the empirical data interpretation process during testing. For additional clarity, a map underlay option was added to the software in order to allow SAR coordinators to import a scalable map underlay for enhanced visual context of generated data. Accompanying this was the ability for dynamic path placement so that operatives could start from different points. Initial stress tests with casing proved that the case fabricated contained structural weaknesses. While the case was capable of withstanding an 11kg weight resting on it, it did not survive drop tests. While successfully surviving stress tests were not required as part of the test process, they provided a means to determine if they met deployable prototype standards. Due to the change in kinematic model, team DIM decided to revisit the MPU-6050 sensor. The original IMU selection process was based off of the LSM9DS0 having a built-in magnetometer, which when fused properly would provide the device with a reference for orientation. However, due to uniform flux and the risk behind making last minute changes, the device was kept strictly for the use of the accelerometer. The reasoning for continuing work with the MPU-6050 was due to the vendor Invensense advertising that the device had a builtin "Digital Motion Processor", capable of generating yaw, pitch and roll at high sampling rates along with a number of other parameters, one of which happened to include a built-in pedometer feature. While the team was aware that the vendor-provided documentation was sparse at best, Jeff Rowberg of I2Cdevlib wrote numerous sample sketches that exhibited the operation of the libraries, with well commented code pertaining to each function call, and its input and output parameters. However, despite improved documentation, and progress made beyond initial compiler errors, team DIM was unable to successfully implement pedometer functionality into their hardware sketch. Using the experience with mitigation during the change in kinematic model, the choice to run testing on the LSM9DS0 implementation while exploring the MPU-6050 option mitigated the risk of an unfinished product.

## XI. HARDWARE

## A. Block Diagram and Document at Block Level

A flowchart has been included to provide a better understanding of how our system works and what will happen if specific thing occur, like pushing the devices button. This details when a file is created and when the files is closed. It also explains what will happen if the user attempts to record without an SDcard in the device.



# B. Schematics and Documentation to Component Level

If it is ever required for repair, the wiring schematic of the device is included on the following page. The inclusion of this wiring schematic does not authorise the reader/users to do repairs or duplicate the device. Do not tamper with and/or destroy the device. Team D.I.M. and its members are not responsible for any damage, whether it be property, physical, and/or emotional, that occurs during the use, maintenance and/or existence of the device.





Fig. 7: Basic Data flow (By: Team D.I.M.)



Fig. 8: Basic Step Calculation (By: Team D.I.M.)

#### XII. SOFTWARE

#### A. Block Diagram and Document at Block Level

The Dark Inertial Mapping project consists of a software and a hardware portion. The visualization software was developed in the Unity3D game development engine. Using the kinematic model and testing results, additional features within the software component were added to facilitate internal testing and promote ease of use for the target demographic of search and rescue coordination managers.

# *B.* Flowcharts, Pseudocode, and Documentation to Subroutine Level

The SD card shield was implemented to fulfill requirements for the Breadboard Proof deadline and demonstrated the local storage feature which was detailed in the Design Idea Contract It also limited the print statements and created less overhead. The SparkFun microSD shield was selected as our storageinterfacing unit, as installation would provide straight-forward given the use of the stacking headers. The use of stackable headers allow for an expansion in utility while maximizing the amount of free pins on the microcontroller. The implementation consisted of a test sketch that acceleration data from an IMU sensor, pushed the data on the stack, calculated the velocity on the microcontroller, and stored the velocity data locally on a microSD card via the Sparkfun microSD shield.

void logData() {
// Write data to file. Start with
log time in micros.
// Write ADC data to CSV record.
for (uint8\_t i = 0; i < DEVICE\_COUNT; i++)
file.print((data[i][0]),5);</pre>

Software: Dark Inertial Mapping



Fig. 9: Basic Step Calculation (By: Team D.I.M.)

```
file . write (', ');
// file . print (0);
file . print (( data [ i ] [ 1 ] ) ,5);
file . write (', ');
// file . print (0);
file . print (( data [ i ] [ 2 ] ) ,5);
file . write (', ');
file . print (( data [ i ] [ 3 ] ) ,5);
}
file . println ();
}
```

After we did that we realized that the Arduino was not deterministic so we added in functionality to calculate the difference in time between the two measurements this would give our post processing calculations a new level of accuracy sense they use a delta time in them. We originally had a bad implementation and this caused the time to be calculated each clock cycle this slowed down the sketch this led us to switch from a function called millis to a function called micros however this did not speed up performance either. Finally we realized that it was our bad implementation and changed the code to resemble the code below.

```
"setup (){
    setup time ...
42
logTime =
micros()/(1000UL*SAMPLE_INTERVAL_MS) + 1;
logTime *= 1000UL*SAMPLE_INTERVAL_MS;
}
loop(){
...time between samples ...
```

```
logTime += 1000UL*SAMPLE_INTERVAL_MS;
}"
```

The software package has gone through massive changes starting as a command line driven program that was written in C++ and over time becoming a GUI with some simple controls to view the data. At the beginning of the year we needed some basic software to help with rapid prototyping of our device. The features needed to be that it was easy to edit for any team member could be quickly deployed. We came up with a basic {console C++ program to fit those exact needs the code was easy to change and you only needed the one file in order to run it. The basic pseudo code listed below was built of the core mathematical functions calculate velocity and generate position. This code was initially tested for along a single axis to validate the position algorithm.

```
// set values initially to 0
float
previous_acceleration, previous_position,
current_position, current_velocity,
current_position = 0;
int clock_frequency = 50;
float delta_time = 1/clock_frequency;
//Open file
while (file has contents) {
input_from_file = file.readFloat();
calculate_current_velocity
= calculateVelocity();
calculate_current_position
= calculatePosition();
printf ("the current position is: %f
and the current velocity is %f",
current_position , current_acceleration );
// set values
previous_acceleration=current_acceleration;
precvious_position = current_position;
previous_velocity = current_velocity ;
}
```

Over time our needs started to change and the ability to see the data being changed became more and more a necessity we then switched to using C# in a game engine called unity. While this code was much more robust and would require a build before running it allowed for more features to be implemented. Most important of those features was the ability to use 3 dimensions to view the data. We also started adding basic filters to our system such as the empirical filter that would take the initial inputs and set them all to 0 and the bound filter that would drive values that were close to 0 within a range of 0 to 0. These filters would help smooth the data and added another level of complexity to our software. Furthermore we would parse the data into an object allowing for quick manupulation of a list of objects after the list is populated from the files. We also added a basic weighted filter to the heading to smooth it and attempt to filter out some of the bad data we are getting from the compass.

```
public class Line
{
public float AccelerationX {get; set;}
public float TimeSlice {get; set;}
public float Heading {get; set;}
}
foreach (var item in input_list){
if(in_step == true){
_headAverage = (item.Heading + .6f*
(_headAverage - item.Heading));
_curIndex++;
```

```
if (item . Acceleration X - average < offset)
// step_counter ++;
in_step = false;
can_step = 1;
_headAverage =( _headAverage * Mathf. PI )/180;
current_position = previous_position
+ new Vector3 ((step_size *
Mathf. Sin(_headAverage)/_scale), 0.0f,
(step_size * Mathf. Cos(_headAverage)/_scale));
if (previous_position != StartPoint){
DrawLineLocal (current_position,
previous_position , CurrentSet);
previous_position = current_position;
ł
else {
if(can_step < 0)
if ((item.AccelerationX - average) > offset){
in_step = true;
_headAverage = item.Heading;
\_curIndex = 2;
else {
can_step --;
}
}
```

This version of the software would see many tweaks to get to the current version. These tweaks were mostly aesthetic and the most major allowed the user to graph multiple sets of data on top of each other starting at different positions. We did this by adding a parent object typed as graphed data and when the clear data on screen is selected we find all objects with that type and delete them. While these changes added small amounts in code they improved the functionality tremendously.

## XIII. MECHANICAL

Since this solution to the social problem was devised, designed and created by a team of electrical and computer engineers, the solution devised did not include what the team considers any mechanical aspects. However, the part of the device which could be interpreted as a mechanical aspect is the device housing which will be displayed and its design explained. All design and models are property of Alexander Meadows. Do not attempt to copy or duplicate the design of any part of this device housing.

### A. Device Housing Main Body

The main body of the case is shown in Figure 10 with arrows added for clarity. The red arrows point to the screw holes which attach the Arduino microcontroller to the case. This is to help keep the wiring attached, the MicroSD card attached, and to assist in reducing the amount of impact wear that the microcontroller will experience. The yellow arrow



Fig. 10: Main case body (By: Team D.I.M.)



Fig. 11: Main case, the other parts (By: Team D.I.M.)

points to the hole that the wiring for the LED goes through. The blue arrow points to the hole in the case that the button attaches to. The extra plastic surrounding the hole is to ensure that that point doesn't become a weak point of the design. The Orange arrow points to the slot in which the base of the case can slide in and out. The slot is to provide both access to the device, and stability to the device when it is fully closed.

#### B. The Other Parts

The main part of the case that attaches to the user's body is shown in Figure 11 by purple arrows. These slots are used to hold the elastic and Velcro straps that attach the device housing to the user. In the final design, it is secured to the top of the case by epoxy. However, the prototype used rubber bands ( Figure 4 The part indicated by the pink arrow is used for attaching the sensors to the housing. It slides into the topmost slot of the device housing, which allows access to the sensors during the prototyping stages. However, it is affixed in place in the final design. The orange arrow points to the bottom of the housing. It is designed in a grid for both strength and weight. There is a small notch designed in so as to make grabbing the piece easier as it is the internal access point of the device.



Fig. 12: Static Time Test

#### C. Part Creation

The parts of the device housing were made using 3D printing on a Duplicator 4 using PLA with 30% infill. The choice of having the parts have infill was threefold. The first reason is that it makes the parts more flexible, and thereby less likely to break. Second, it makes the device slightly lighter as the case contains less plastic. The final deciding factor was that it shortened the printing time, which was about nine hours for all the parts. Since the device housing is made out of plastic, finding out how the housing handles being dropped was something we needed to test.

# XIV. TEST PLAN AND TEST RESULTS FOR HARDWARE

## A. Static Timed Test

Run the device in a fixed location at California State University, Sacramento CA 95819, inside Riverside Hall 3rd floor. Test times are 1, 5, 15, 30, 12 and 24 hours.

1) Test results: Initially the inclusion of the test was the result of the double-integration of accelerometer data to get displacement. However, this test was adapted as a way to test sensor stability over time. Because of how the static threshold is implemented, it was expected that after 24 hours of runtime, no steps would be generated, as none of the accelerometer values generated by the z-axis would generate values that crossed the threshold. As expected, the resulting path did not deviate from the point of origin as depicted in Figure 5.

## B. Speed Test

Walk a repeatedly a mapped path while on the same data logging run at California State University, Sacramento CA 95819, inside Riverside Hall 3rd floor.

1) Test Results: With the current gait implementation, the static gait trigger is triggered based on the accelerometer's z-axis reading crossing a pre-determined thresh-hold, an increase in gait speed lead to potentially increasing the stepcounter. Running versus walking presented a unique situation where the gait length was so wide that it actually led to assumed



Fig. 13: Drop Test resulted in a fracture in the enclosure (By: Team D.I.M.)

shorter distances due to the pre-determined gait length. While it may be possible to finding a way to adaptively determine movement speed, it lies outside of the primary scope of testing.

### C. Drop Test

This test was done in two parts, a 1 meter test and a 3.658 meter test. Both tests were run without the hardware in the housing as it would not be financially responsible to do so at this point in time.

1) Test Results: The purpose of the 1 meter test is to observe the effect of a typical drop that the device could experience. This test was performed by Alexander Meadows in Riverside 3001. Using a tape measure, he measured out a meter in height and aligned the bottom of the case with the one meter mark. Moving the tape measure away, he then dropped the housing onto the tiled floor, picked it up, and looked over the case for damage. There was no visible damage to the housing. This means that the housing can hold up to the average drop from about waist height.

The purpose of the 3.658 meter drop test is to observe the effects of a more significant drop, such as falling from one floor to another. This test was performed by Yan Lin in the Riverside stairway. He took the housing to the stairway and dropped it from the second floor. The housing split apart into individual pieces because the top cover was not glued on. That being said, the housing did take damage from the fall, creating a crack along the corner and edge of one side as seen in the figure below which was an expected result. What this result means is that the device housing could be further reinforced and that future drop tests should be done on a housing which has the removable side glued shut as planned for the final device.

## D. Crush Test

This test was run by Alex at the Well weight lifting area on concrete. While the device was powered and recording data, Alex placed a 5lb dumbbell on the device and observed its effects for one minute (4 minutes for the 25lb weight). He then took off the 5lb dumbbell and replaced it with a 10lb dumbbell, once again observing the added weights effects. He repeated this process with 15lb, 20lb, and 25lb dumbbells each time observing the effects of the weight on the housing.

1) Test Results: No observable deformities were seen, nor any cracking under any weight. The data recording and power was maintained throughout the test. This test went better than expected result of having some deformities and cracking. What this result means is that the case can withstand having some weight put on it.

## E. Shake Test

The Housing will be vigorously shaken by hand for 3 minutes at California State University, Sacramento CA 95819 Rm 3001.

1) Test Results: Upon initial inspection of hardware assembly inside the enclosure, it was apparent that the components were not securely affixed to enclosure. The test was aborted in order to mitigate imminent hardware component damage. The inclusion of an Arduino base holder in the next housing version will enable the test to be performed at a later time.

# F. Dissipation Test

The dissipation test was not run because it became obsolete due to there being no heat problems or observable heat generation during prolonged usage of the device. This was determined during the battery runtime test which determined that the battery would run out before the device generated any notable amount of heat.

## G. Moisture Test

The moisture test was put on hold initially due to the lack of a suitable neoprene case. While the test was to occur once a neoprene case had been purchased, it eventually was not run due a combination of a lack of time and the actual case.

## H. Shatter Test

The Shatter Test became obsolete when the housing partly broke during the drop test. If the housing could not normally survive the fall from one story, it most likely would not survive it if it was supercooled either.

#### I. Battery Runtime Test

This test was run by and at the home of Alexander Meadows. A web camera was setup to record the device during the testing process. The device was then powered on (being powered by a new Energizer lithium ion battery or a new Duracell alkaline battery) and the recording started.



Fig. 14: Short Distance Test around center tables in 3001 Riverside Hall (By: Team D.I.M.)

1) Test Results: The device ran for over 10.5hrs before losing power, which is longer than the expected time between most search and rescue personnel's report periods. What this result means is that a 9V can supply power to the device for an extended period of time. Only the lithium ion worked for our application as it not only powered the device, it did so for an extended period of time. The N-type battery did not successfully power the Arduino, and the alkaline 9V did not power the device for quite as long (8hrs), so it did not provide as large a time buffer for search and rescue personnel to return.

# XV. TEST PLAN AND TEST RESULTS FOR SOFTWARE

# A. Short Distance Test

The Short Distance Test was conducted within the Senior Design Room. Roughly 50 trials were conducted around the center table. The perimeter of the path traversed consisted of a rectangular path of 9.144 meters by 3.048 meters. The average gait length for a standard adult male is 2.5 feet or 0.762 meters. Using a fixed gait length of 0.725 to accomodate for short steps because of a non-standard movement path, the test was run five times around the perimeter of the tables.

1) Test Results: Figure 14 shows the result of running the same test around the table with accumulated error over time. Each separate color indicates a complete lap around the table. What initially was targeted to be a poor static thresh hold for the pedometer function was actually the result of magnetic interference in the testing environment with the HMC-6343 Honeywell Three-Axis Compass. This is validated in Figure 2 where a line test is conducted. The length of 9.144 meters is traversed with a total of 13 actual steps taken. The point at the origin is always generated as part of the startup procedure. Comparing the real world movement against the sensor data, the user moves forward in a directly straight line. However, kept relatively stable, the compass fluctuates exhibiting amorphous behavior in terms of heading.



Fig. 15: Line Test (By: Team D.I.M.)



Fig. 16: Medium Distance Test on 3rd Floor Hallway, Riverside Hall (By: Team D.I.M.)

# B. Medium Distance Test

Walk a repeatedly a mapped path 5 times while on the same data logging run at California State University, Sacramento CA 95819, inside Riverside Hall 3rd floor. The Medium Distance Test was conducted on the second floor of the Riverside Engineering building following the completion of the Short Distance test. The subject began at the South side stairwell to the corner of the hallway for a distance of 11.2776 meters. Upon reaching the corner, the subject would turn 90 degrees anti-clockwise, and traverse the distance to the end of the hall for a distance of 52.73 meters. The subject would then turn another 90 degrees anti-clockwise and walk another 11.2776 meters to the exit to the North Stairwell. To test for the amount of error during the traversal, the same path was taken in reverse to view how far from the start point the search path deviated.

1) Test Results: Figure 3 shows the device results from conducting the medium distance test. Upon initial inspection, the heading was compromised following the 90 degree an-


Fig. 17: Long Distance Test between Riverside Hall and Sequoia Hall (By: Team D.I.M.)

ticlockwise turn. To find out the cause of these issues, a serial terminal print test was performed to evalulate the sensor readings for the duration of the test. Walking North through the hallway gave a rough heading between 350 degrees and 10 degrees During multiple instances, when the team walked past Riverside 3005, the heading flipped 180 degrees to point in the south direction until a certain distance past the entrance of the room. This location is portrayed in Figure 3 using the yellow arrow. The repeated paths across the entrance of this room yielded compromised heading results, which was noted and the basis for discovering the impact of magnetic interference caused by wireless network transmitters.

#### XVI. LONG DISTANCE TEST

Walk any repeatable path outside, on campus at California State University, Sacramento CA 95819.

2) Test Results: The results from the Medium and Short distance tests suggested that the use of the Dark Inertial Mapping device indoors would be susceptible to magnetic interference generated by electrical wiring and wireless repeaters. This test was conducted outdoors both to encompass a larger search path, as well as using a third case to test for sensor interference. The tested path consisted of the South east corner of the Riverside building, northward to Sequoia Hall, a displacement westward, then a return path to the original position. The result shown in Figure 4 suggest that the Long distance test may suffer greatest from the static step threshhold generated by the pedometer algorithm due to a lack of precision.

#### A. Unit Test

Unit tests are a secondary project that is built that imports the function of the project you are building and runs it and then you write cases for the program to alert you if the function has failed.

1) Test Results: The main testing decided upon for the software was unit testing. Originally, it was believed that unit tests would help to gage the validity of the software and ensure that its functionality was all correctly configured. Due to the complexity of creating a unit test for a software that has already been written it was decided that the results obtained from these test would not be worth the effort to create them. To eliminate these potential issue most unit tests that are written while the software is being developed. Typically the development team will use the method of test driven development. This method changes the order of writing code writing the unit tests first then writing the code that you are testing. This lengthens the time of development but reduces the time needed for changes or updates allowing the developer to use the previously developed test cases to ensure that the software is working correctly even after changes.

Unit test are tough to write in post production because they do not import the entire project. To ensure precision the unit test only opens and uses that set of code for that individual function. However, this means that unit testing only works well in cases where the code is only dependent on itself and not any other functions. There are ways around using other functions and you can write in most libraries using something called fakes which represent outside software using predefined values. However the more fakes you have to write the more you have to define your inputs and change the data that you are getting from the unit tests.

After finishing our basic unit testing and seeing that those test only test one function at a time it was easy to see that the data was not worth the time it would take to build the unit tests. After determining that the data gathered from this was not pertinent and would not help the software improve there was a testing switch and the software was then ran through a graphical user interface test. The users of the software responded very positively to the new tools added into the software and tested all their functionality to ensure that the software has no hiccups in its workings.

#### B. Sample Data Test

Create test cases from a predefined output and run them through the software comparing the output to the predefined output.

1) Test Results: These test used test cases that were created using predefined data that was built from a desired output. These test were used for both types of files that are being passed to the software: The preprocessed file that passes when a step occurred and what the persons heading was when that step occurred, as well as the non processed raw data file. For this test the test cases would run through the software and the output would be manually compared to the expected output. These test were run and had no complications giving the expected output. These fake data sets will be used for future cases to ensure that the software is working before releasing a build. This test does not work with the unit test and will need to be manually run. Ensure that the file saved by the device is readable by the program.

#### A. Test Results

Using the results obtained in the distance testing it is evident that the data recorded by the device is being parsed correctly by the software.

#### XVIII. OVERLOAD TEST

Testing where the user running the software attempts to break the software by clicking many inputs in rapid succession.

#### A. Test Results

This test resulted in crashes because Unity projects do not do a good job of memory management. In order to keep these crashes to a minimal there will be an introduction of a lock after clicking. This will lock the user from clicking too many things at once and will result in less crashes.

### XIX. CONCLUSION

Urban Search and Rescue operations typically suffer from signal damping and reflection as a result of building geometry. In order to address the needs of Search and Rescue teams under these conditions, Team DIM assessed the responsibilities of both Task Forces and Incident Support teams. Based on the interaction, a wearable solution was created that generated kinematic data. Upon completion of a search path, this data was interpreted with respect to human gait patterns. Using this data, search maps were created within a virtual space to simulate the paths travelled by the operatives, allowing for the generation of localization data in areas that were unable to sustain a reliable level of GPS service. The two project terms spanning Fall 2014 through Spring 2015 provided the timeline for the creation of the Dark Inertial Mapping system. The fall term was used to find a societal problem with compelling evidence supporting this claim, and to propose an engineering solution to address it. Despite a number of changes that occurred as the result of the prototype review at the end of the Fall project term, the managed to revise the device to reflect a new kinematic model while still falling within feature set enumerated in the original Design Idea Contract. A market view was conducted in order to assess the With an enhanced understanding of the device's intended use and role, the original problem statement was revised to target Search Coordination Managers. As a result, software was updated accordingly to provide a better user experience for the end product. The system as a whole intends to increase the rate of survival in search and rescue operations by enhancing coordination through the reduction of redundancy in search path planning.

#### GLOSSARY

### IMU

Inertial Measurement Unit. 8–11, 13–15, 20, 24 infill

The process of creating a honey comb like structure to use less plastic while 3D printing. 26

PDR

Pedestrian Dead Reckoning, A mapping method employing the use of human gait patterns to generate displacement path from a previously known position.. 1, 4, 7, 17

PLA

Polylactic Acid. 26

#### ACKNOWLEDGMENT

The authors would like to thank Raymond Stedronsky for his technical consultation regarding standard operating procedures of search and rescue work. Additionally, the authors would like to thank Professor Russ Tatro, Dr. Fethi Belkhouche for their feedback and support regarding implementation and design.

#### REFERENCES

- [1] "Urban search —& rescue participants," Jul. 2014. [Online]. Available: https://www.fema.gov/urban-search-rescue-participants
- [2] A. Kleiner. (2007) Mapping and exploration for search and rescue with humans and mobile robots. [Online]. Available: http://www.freidok. unifreiburg.de/volltexte/4548/pdf/diss.pdf
- [3] U. F. Administration. (2011) Summary incident report 2005-2010. [Online]. Available: http://apps.usfa.fema.gov/firefighter-fatalities/ fatalityData/reportBuilder
- [4] J. S. T.L. Holzner. (2013) Global earthquake fatalities and population. [Online]. Available: https://profile.usgs.gov/myscience/upload\_folder/ ci2013Feb2015013642954Holzer\%20Savage\%20Fatalities\%20EQ\ %20Spectra\%202013.pdf
- [5] D. S. R. Lin, R. Xia. (2014) Uc quake researchers give l.a. list of old concrete buildings. [Online]. Available: http://www.latimes.com/local/ lanow/la-me-ln-concrete-list-earthquake-20140121-story.html
- [6] G. G. L. Mulvany. (2011) Firefighters balk at post-9/11 digital radios, as failures risk lives. [Online]. Available: http://www.mcclatchydc.com/ 2011/09/06/123290\_firefighters-balk-at-new-digital.html?rh=1
- [7] S. M. F. S. . Ray Stedronsky. Retired Firefighter, Interview, 2014, 2015.
- [8] P. H. H. Jones. (2002) Extreme work teams: Using swat teams as a model for coording distributed robot. [Online]. Available: http://web.stanford.edu/~phinds/PDFs/Jones-Hinds-2002.pdf
- [9] C. R. James Wong. (2004) Urban search and rescue technology needs: Identification of needs. [Online]. Available: http://www.ncjrs. gov/pdffiles1/nij/grants/207771.pdf
- [10] K. (1993) The collapse Weick. of sensemaking in organizations: The mann gulch disaster. [Online]. Available: http://www.nifc.gov/safety/mann\_gulch/suggested\_reading/The\_ Collapse\_of\_Sensemaking\_in\_Organizations\_The\_Mann\_Gulch.pdf
- [11] F. Bauer. (2014, Oct.) Welcome to cal-esar training. [Online]. Available: http://training.cal-esar.org/
- [12] S. Rescue. (2015, Mar.) Home. [Online]. Available: https://sierrarescue. com/
- [13] D. Burchard. (2014, Oct.) Kcsara academy. [Online]. Available: http://kcsara.org/training/kcsara\_academy
- [14] U. S. Coastguard. (2015, Mar.) National search and rescue school. [Online]. Available: http://www.uscg.mil/tcyorktown/Ops/SAR/default. asp
- [15] E. M. Rescue. (2014, Mar.) About emr. [Online]. Available: http://www.eugenemountainrescue.org/how-to-join
- [16] S. C. Search and Rescue. (2015, Mar.) About us a breif history. [Online]. Available: http://sacsar.com/aboutus.html
- [17] willcountyema.org. (2015, Mar.) Search —& rescue (sar) team. [Online]. Available: http://www.willcountyema.org/searchrescue.php

- [18] K. Grimes. (2015, Mar.) Travis county search —& rescue. [Online]. Available: http://www.tcsar.org/
- [19] C. of El Dorado. (2015, Mar.) Search and rescue. [Online]. Available: http://www.edcgov.us/Government/Sheriff/Divisions/Support/ Search\_and\_Rescue.aspx
- [20] CMSART. (2015, Mar.) About cmsart. [Online]. Available: http: //www.cmsart.org/cmsart/
- [21] T. Search and Rescue. (2015, Mar.) About texsar. [Online]. Available: http://www.texsar.org/about/
- [22] Cal-ESAR. (2015, Mar.) About cal-esar. [Online]. Available: http: //www.cal-esar.org/home
- [23] T. A. N. R. Cross. (2015, Mar.) The global red cross network. [Online]. Available: http://www.redcross.org/about-us/history/ global-red-cross-network
- [24] W. Stafford. (2015, Mar.) Poised to build haiti's future. [Online]. Available: http://www.compassion.com/sponsordonor/compassionmagazine/ poisedtorebuildhaitisfuturesummer2010/
- [25] I. R. Committee. (2015, Mar.) Emergency response relief. [Online]. Available: http://www.rescue.org/our-work/emergency-response
- [26] J. Nicas. (2011) Metal detectors hit the jackpot. [Online]. Available: http://www.wsj.com/articles/ SB10001424053111904233404576462161651031814e
- [27]
- [28] D. Schneider. (2013, Nov.) New indoor navigation technologies work work where can't. gps [Online]. http://spectrum.ieee.org/telecom/wireless/ Available: new-indoor-navigation-technologies-work-where-gps-cant
- [29] Decawave. (2013, Nov.) Product overview. [Online]. Available: http://www.decawave.com/products/overview
- [30] gps repeaters.com. (2013, Nov.) Indoor gps coverage solutions. [Online]. Available: http://www.gps-repeaters.com
- [31] H. G. Carl Fischer, "Location and navigation support for emergency responders: A survey," *Pervasive Computing, IEEE*, no. 1, pp. 201–213, 38 2009.
- [32] P. E. R. Forum. (2014) The police response to active shooter incidents. [Online]. Available: http://www.policeforum.org/assets/ docs/Critical\_Issues\_Series/the\%20police\%20response\%20to\ %20active\%20shooter\%20incidents\%202014.pdf
- [33] Invensense. (2013, Oct.) Mpu-6000 and mpu-6050 product specification revision 3.4. [Online]. Available: http://invensense.com/mems/gyro/ documents/PS-MPU-6000A-00v3.4.pdf
- [34] STMicroelectronics. (2013) Lsm9ds0. [Online]. Available: https: //cdn.sparkfun.com/assets/f/6/1/f/0/LSM9DS0.pdf
- [35] K. Blattenberger. (2014) Copper wire properties —& gauge conversions. [Online]. Available: http://www.rfcafe.com/references/electrical/wire-cu. htm
- [36] A. Spruce and S. Co. (2014) Shielded wire mil-c-27500. [Online]. Available: http://www.aircraftspruce.com/catalog/elpages/shieldwire.php
- [37] E. H. INC., "Product datasheet," http://data.energizer.com/PDFs/la522.pdf, 2015 (accessed April 04, 2015).

APPENDICES

A. Data sheets



# Honeywell

## **3-Axis Compass with Algorithms** HMC6343

The Honeywell HMC6343 is a fully integrated compass module that includes firmware for heading computation and calibration for magnetic distortions. The module combines 3-axis magneto-resistive sensors and 3-axis MEMS accelerometers, analog and digital support circuits, microprocessor and algorithms required for heading computation. By combining the sensor elements, processing electronics, and firmware into a 9.0mm by 9.0mm by 1.9mm LCC package, Honeywell offers a complete, ready to use tilt-compensated electronic compass. This provides design engineers with the simplest solution to integrate high volume, cost effective compasses into binoculars, cameras, night vision optics, laser ranger finders, antenna positioning, and other industrial compassing applications.



The HMC6343 utilizes Honeywell's Anisotropic Magnetoresistive (AMR) technology that provides advantages over other magnetic sensor technologies. The sensors feature precision sensitivity and linearity, solid-state construction with very low cross-axis sensitivity designed to measure both direction and magnitude of Earth's magnetic fields. Honeywell's Magnetic Sensors are among the most sensitive and reliable low-field sensors in the industry.

Honeywell continues to maintain product excellence and performance by introducing innovative solid-state magnetic sensor solutions. Honeywell's magnetic sensor solutions provide real solutions you can count on.

### FEATURES

### BENEFITS

►	Compass with Heading/Tilt Outputs	►	A complete compass solution including compass firmware
▶	3-axis MR Sensors, Accelerometers and a Microprocessor in a Single Package	•	A digital compass solution with heading and tilt angle outputs in a chip-scale package
▶	Compass Algorithms	►	For computation of heading, and magnetic calibration for hard-iror
►	9 x 9 x 1.9mm LCC Surface Mount Package	•	Small size, easy to assemble and compatible with high speed surface mount technology assembly
▶	Low Voltage Operations	►	Compatible with battery powered applications
►	EEPROM Memory	►	To store compass data for processor routines
►	Digital Serial Data Interface	►	I <sup>2</sup> C Interface, easy to use 2-wire communication for heading output
►	Moderate Precision Outputs	►	Typical 2° Heading Accuracy with 1° Pitch and Roll Accuracy
►	Lead Free Package Construction	►	Complies with RoHS environmental standards
►	Flexible Mounting	►	Can be mounted on horizontal or vertical circuit boards

### SPECIFICATIONS

Characteristics	Conditions*	Min	Тур	Max	Units
Power Supply					
Supply Voltage	VDD Referenced to GND	2.7	3.3	3.6	Volts
Current	All VDD pins connected together				
	Run Mode (10Hz Output)	3.5	4.5	5.5	mA
	Standby Mode			1.0	mA
	Sleep mode		10		μA
	Power-up peak (VDD = 3.3V)		8		mA
Power-on Rate	Minimum rise time for POR	0.05	-	-	V/msec
Compass Function					
Field Range	total applied magnetic field		±1	±2	gauss
	(de-gauss if exposed to >5gauss)				
Heading Accuracy	At Level, +3.3V	1.0	2.0	3.0	deg RMS
	±15° tilt		3.0		
	±60° tilt		4.0		
Heading Resolution	Output Data		0.1		degrees
Heading	Output Data (1σ)		±0.3		degrees
Repeatability					
Heading Hysteresis	Output Data (1σ)		±0.3		degrees
Update Rate	Run Mode (1, 5, 10Hz)	1	5	10	Hz
Tilt Range	From Horizontal		±80		degrees
Tilt Accuracy	0° to ±15°, +3.3V		±1		degrees
	±15° to ±60°		<u>+2</u>		
Tilt Resolution Output Data			0.1		degrees
Tilt Repeatability	Output Data (1o)		±0.2		degrees
Offset Straps					
Resistance	Measured from OFF+ to OFF-	5	8	11	ohms
Offset	DC Current		10		mA/gauss
Constant	Field applied in sensitive direction				
Resistance	T <sub>A</sub> =-40 to 125°C	1800	2700	4500	ppm/°C
Tempco					
General					
Operating	Ambient	-40		80	°C
Temperature					
Storage	Ambient, unbiased	-55		125	°C
Temperature					
Weight			0.32		grams
ESD Voltage				400	V
MSL	Moisture Sensitivity Level	3			-
Solder Temp Peak Reflow Temp (< 30 seconds)				250	°C

\* Tested at 25°C and 3.3V except stated otherwise.

### **FUNCTIONAL DIAGRAM**



### **PIN CONFIGURATIONS**

Pin Number Description		Pin Number	Description
1	NC	19	Y OFF-
2	NC	20	Y OFF+
3	VDD1	21	VDD2
4	NC	22	CS
5	NC	23	X OFF-
6	NC	24	X OFF+
7	NC	25	GND2
8	NC	26	NC
9	NC	27	NC
10	NC	28	NC
11	VDD3	29	GND1
12	12 NC		NC
13	NC	31	NC
14	NC	32	SCK/SCL
15	Z OFF-	33	NC
16	Z OFF+	34	NC
17	17 NC		CS_CTRL
18	NC	36	SDA



### **PACKAGE OUTLINES**



### PACKAGE DRAWING HMC6343 (32-PIN LPCC, dimensions in millimeters)

Dimensions (mm)	Minimum	Nominal	Maximum
A (height)	1.73	1.87	2.02
D	-	9.00 BSC	-
D1	-	6.40 BSC	-
E	-	9.00 BSC	-
E1	-	6.40 BSC	-
е	-	0.8 Basic	-

### **MOUNTING CONSIDERATIONS**

The following is the recommend printed circuit board (PCB) footprint for the HMC6343. All dimensions are nominal and in millimeters.

### **Stencil Design and Solder Paste**

A 4-6 mil stencil and 100% paste coverage is recommended for the electrical contact pads. The HMC6343 has been assembled successfully with no-clean solder paste.

.

### **BASIC DEVICE OPERATION**

The Honeywell HMC6343 magnetoresistive sensor circuit is a trio of magnetic sensors, accelerometers, and analog support circuits to measure magnetic fields. Additionally a microcontroller is integrated for computation of direction and calibration. With power supply applied, the sensor converts any incident magnetic field in the sensitive axis direction to a differential voltage output. In addition to the bridge circuit, the sensors have on-chip magnetically coupled offset straps for incident field adjustment.

The circuit is sensitive to power supply noise, and adding a 1.0 microfarad ceramic capacitor is recommended on the positive supply to help reduce noise. Also careful layout practices should be enforced to keep high current traces (>10mA) a few millimeters away from the sensors. Also, since the sensors are typically sensing the earth's magnetic field direction, avoid employing RF/EMI shields using ferrous metals or coatings.

### **BASIC SCHEMATIC INTERFACE**



#### **Offset Straps**

The three offset straps have a spiral of metallization that couples in the sensor element's sensitive axis. The straps will handle currents to buck or boost fields through the  $\pm 4$  gauss linear measurement range, but designers should note the thermal heating on the die when doing so.

With most applications, the offset strap is not utilized and can be ignored. Designers can leave one or both strap connections (Off- and Off+) open circuited, or ground one connection node.

#### **Operational Modes**

The HMC6343 has three operational modes; Sleep, Standby, and Run. Sleep mode is defined as having the analog circuitry powered off, and has the lowest power consumption while power is applied to the VDD pins. Standby mode has the HMC6343 fully powered, but with no measurements performed and the processor is waiting for commands to perform. Run mode is fully engaged in continuous measurements at the set rate, and ready to receive further commands. The operational mode settings are stored in EEPROM register 0x04, and shown further the HMC6343 protocol definition.

### **Mounting Orientations**

The HMC6343 provides for three standard mounting orientations, with a flat horizontal orientation (Level) as the factory default. For vertical mounting, there are two upright orientations with either the X-axis or the Z-axis designated as the forward reference directions. To change the forward reference direction temporarily, send the appropriate command byte (0x72, 0x73, or 0x74) for level or upright orientations. For other orientations, you can add or subtract 90 degree increments of deviation angle as required from the three choices. The figure below shows pictorially the orientations.

To permanently change orientation, poke EEPROM Operational Mode Register 1 (0x04) with the appropriate binary bits set for Level, Upright Edge (UE), or Upright Front (UF). The HMC6343 will operate in the selected orientation after a power-up or reset command. More on the EEPROM registers in the following sections.



### **Red Arrow is the Forward Direction**

### I<sup>2</sup>C COMMUNICATION PROTOCOL

The HMC6343 communicates via a two-wire  $\hat{f}C$  bus system as a slave device. The HMC6343 uses a layered protocol with the interface protocol defined by the  $\hat{f}C$  bus specification, and the lower command protocol defined by Honeywell. The data rate is the standard-mode 100kbps rate as defined in the  $\hat{f}C$  Bus Specification 2.1. The bus bit format is an 8-bit Data/Address send and a 1-bit acknowledge bit. The format of the data bytes (payload) shall be case sensitive ASCII characters or binary data to the HMC6343 slave, and binary data returned. Negative binary values will be in two's complement form. The default (factory) HMC6343 7-bit slave address is 0x32 for write operations, or 0x33 for read operations.

The HMC6343 Serial Clock (SCL) and Serial Data (SDA) lines do not have internal pull-up resistors, and require resistive pull-ups (Rp) between the master device (usually a host microprocessor) and the HMC6343. Pull-up resistance values of about 10k ohms are recommended with a nominal 3.3-volt supply voltage. Other values may be used as defined in the  $l^2C$  Bus Specification 2.1.

The SCL and SDA lines in this bus specification can be connected to a host of devices. The bus can be a single master to multiple slaves, or it can be a multiple master configuration. All data transfers are initiated by the master device which is responsible for generating the clock signal, and the data transfers are 8 bit long. All devices are addressed by  $^{2}C$ 's unique 7 bit address. After each 8-bit transfer, the master device generates a 9 th clock pulse, and releases the SDA line.

The receiving device (addressed slave) will pull the SDA line low to acknowledge (ACK) the successful transfer or leave the SDA high to negative acknowledge (NACK).

Per the <sup>2</sup>C spec, all transitions in the SDA line must occur when SCL is low. This requirement leads to two unique conditions on the bus associated with the SDA transitions when SCL is high. Master device pulling the SDA line low while the SCL line is high indicates the Start (S) condition, and the Stop (P) condition is when the SDA line is pulled high while the SCL line is high. The <sup>2</sup>C protocol also allows for the Restart condition in which the master device issues a second start condition without issuing a stop.

All bus transactions begin with the master device issuing the start sequence followed by the slave address byte. The address byte contains the slave address; the upper 7 bits (bits7-1), and the Least Significant bit (LSb). The LSb of the address byte designates if the operation is a read (LSb=1) or a write (LSb=0). At the 9<sup>th</sup> clock pulse, the receiving slave device will issue the ACK (or NACK). Following these bus events, the master will send data bytes for a write operation, or the slave will clock out data with a read operation. All bus transactions are terminated with the master issuing a stop sequence.

I<sup>2</sup>C bus control can be implemented with either hardware logic or in software. Typical hardware designs will release the SDA and SCL lines as appropriate to allow the slave device to manipulate these lines. In a software implementation, care must be taken to perform these tasks in code.

#### I<sup>2</sup>C Slave Address

The m fC slave address byte consists of the 7 most significant bits with the least significant bit zero filled. As described earlier, the default (factory) value is 0x32 and the legal m fC bounded values are between 0x10 and 0xF6. This slave address is in EEPROM address 0x00. Users can change the slave address by writing to this location. Any address updates will become effective after the next power up or after a reset command.

#### **Software Version**

This EEPROM software version number byte contains the binary value of the programmed software. Values of 0x05 and beyond are considered production software.

#### **Deviation Angle Correction**

Typically the HMC6343 X-axis (or Z-axis) is designated the forward direction of the compass, and is placed mechanically towards the forward direction of the end user product. The deviation angle is used to correct for mechanical angle errors in package orientation by adding the deviation angle to the internal compass heading before the result is placed as the computed heading. Two EEPROM Bytes are used to store the deviation angle, and the binary value is in tenths of a degree and in two's complement form for a ±1800 representation. The deviation angle MSB is located in EEPROM register 0x0B and the LSB in 0x0A.

#### Variation Angle Correction

The variation angle or declination angle of the HMC6343 is the number of degree that must be added to the internal compass heading to convert the magnetic north reference direction to the geographic (true) north reference direction. This angle information is provided to the HMC6343 from external latitude and longitude data processed through a World Magnetic Model equation to compute variation angle, or by lookup table. Two EEPROM Bytes are used to store the variation angle, and the binary value is in tenths of a degree and in two's complement form for a ±1800 representation. The deviation angle MSB is located in EEPROM register 0x0D and the LSB in 0x0C.

#### Magnetometer Offsets

The Magnetometer Offset bytes are the values stored after the completion of the last factory or user hard-iron calibration routine. Additional value changes are possible, but will be overwritten when the next calibration routine is completed. Note that these offset values are added to the sensor offset values computed by the set/reset routine to convert the raw magnetometer data to the compensated magnetometer data. These values are written into EEPROM addresses 0x0E to 0x13 and loaded to RAM on the power up.

### **Heading Filter**

This allows for an Infinite Impulse Response (IIR) filter to be employed on current and previous heading data outputs. Typical values are 0 to 15 with a factory default of zero. The filter is only applied in run mode where a continuous stream of data is present. At the 5Hz default update rate, a filter value of 4 would weight the latest heading with the previous four headings of regressive weightings for a second's worth of filtering.

#### **EEPROM Registers**

The HMC6343 contains EEPROM non-volatile memory locations (registers) to store useful compass data for processor routines. The following Table shows the register locations, content, description, and factory shipped defaults.

### Table 1 – EEPROM Registers

EEPROM Location	Content	Description	Factory Default
0x00	Slave Address	I2C Slave Address	0x32
0x01	Reserved		
0x02	S/W_Version	Software Version Number	
0x03	Reserved		
0x04	OP_Mode1	Operational Mode Register 1	0x11
0x05	OP_Mode2	Operational Mode Register 2	0x01
0x06	S/N LSB	Device Serial Number	
0x07	S/N MSB	Device Serial Number	
0x08	Date Code: YY	Package Date Code: Last Two Digits of the Year	Year
0x09	Date Code: WW	Package Date Code: Fiscal Week	Week
0x0A	Deviation LSB	Deviation Angle (±1800) in tenths of a degree	0x00
0x0B	Deviation MSB	Deviation Angle (±1800) in tenths of a degree	0x00
0x0C	Variation LSB	Variation Angle (±1800) in tenths of a degree	0x00
0x0D	Variation MSB	Variation Angle (±1800) in tenths of a degree	0x00
0x0E	X_Offset LSB	Hard-Iron Calibration Offset for the X-axis	0x00
0x0F	X_Offset MSB	Hard-Iron Calibration Offset for the X-axis	0x00
0x10	Y_Offset LSB	Hard-Iron Calibration Offset for the Y-axis	0x00
0x11	Y_Offset MSB	Hard-Iron Calibration Offset for the Y-axis	0x00
0x12	Z_Offset LSB	Hard-Iron Calibration Offset for the Z-axis	0x00
0x13	Z Offset MSB	Hard-Iron Calibration Offset for the Z-axis	0x00
0x14	Filter LSB	Heading IIR Filter (0x00 to 0x0F typical)	0x00
0x15	Filter MSB	Heading IIR Filter (set at zero)	0x00

### **Command Protocol**

The command protocol defines the content of the data (payload) bytes of  $m f^2C$  protocol sent by the master, and the slave device (HMC6343). Note that angular outputs are in tenths of a degree (0-3600 heading, ±0-900 tilt).

After the master device sends the 7-bit slave address, the 1-bit Read/Write, and gets the 1-bit slave device acknowledge bit returned; the next one to three sent data bytes are defined as the input command and argument bytes. To conserve data traffic, all response data (Reads) will be context sensitive to the last command (Write) sent. All write commands shall have the address byte least significant bit cleared (factory default 0x32). These commands then follow with the command byte and command specific binary formatted argument bytes in the general form of:

(Command Byte) (Argument Binary MS Byte) (Argument Binary LS Byte)

The slave (HMC6343) shall provide the acknowledge bits between each data byte per the  $^{2}C$  protocol. Response byte reads are done by sending the address byte (factory default 0x33) with the least significant bit set, and then clocking back response bytes, last command dependant. Table 2 shows the HMC6343 command and response data flow.

#### Table 2 – HMC6343 Interface Commands/Responses

Command Byte (hex)	Argument 1 Byte (Binary)	Argument 2 Byte (Binary)	Response Bytes (Binary)	Command Description
(0x40)			MSB/LSB Data (6 Bytes)	Post Accel Data. AxMSB, AxLSB, AyMSB, AyLSB, AzMSB, AzLSB
(0x45)			MSB/LSB Data (6 Bytes)	Post Mag Data. MxMSB, MxLSB, MyMSB, MyLSB, MzMSB, MzLSB
(0x50)			MSB/LSB Data (6 Bytes)	Post Heading Data. HeadMSB, HeadLSB, PitchMSB, PitchLSB, RollMSB, RollLSB
(0x55)			MSB/LSB Data (6 Bytes)	Post Tilt Data. PitchMSB, PitchLSB, RollMSB, RollLSB, TempMSB, TempLSB
(0x65)			Post OP Mode 1	Read the current value of OP Mode 1
(0x71)				Enter User Calibration Mode
(0x72)				Level Orientation (X=forward, +Z=up) (default)
(0x73)				Upright Sideways Orientation (X=forward, Y=up)
(0x74)				Upright Flat Front Orientation (Z=forward, -X=up)
(0x75)				Enter Run Mode (from Standby Mode)
(0x76)				Enter Standby Mode (from Run Mode)
(0x7E)				Exit User Calibration Mode
(0x82)				Reset the Processor
(0x83)				Enter Sleep Mode (from Run Mode)
(0x84)				Exit Sleep Mode (to Standby Mode)
(0xE1)	EEPROM Address		Data (1 Byte)	Read from EEPROM
(0xF1)	EEPROM Address	Data		Write to EEPROM

### Timing

Upon power application to the HMC6343, wait nominally 500 milli-seconds before sending the first I2C command (typically a 0x32 byte followed by a 0x50 byte for the usual heading/pitch/roll). Depending on the command sent, a delay time should be inserted before clocking out the response bytes (send 0x33, clock back response bytes). The following table indicates the response delay times for various commands.

Table 3 – HMC6343	Command to	Response	<b>Delay Times</b>
-------------------	------------	----------	--------------------

Prior Command (hex)	Commanded Action	Response Bytes & Description	Response/Delay Time (milli-seconds)
Power Applied	VDD1-3 low to high	No Response Data	500 nominally
0x40	Post Accel Data.	6 binary data Bytes. AxMSB, AxLSB, AyMSB, AyLSB, AzMSB, AzLSB	1
0x45	Post Mag Data.	6 binary data Bytes. MxMSB, MxLSB, MyMSB, MyLSB, MzMSB, MzLSB	1
0x50	Post Heading Data.	6 binary data Bytes. HeadMSB, HeadLSB, PitchMSB, PitchLSB, RollMSB, RollLSB	1
0x55	Post Tilt Data.	6 binary data Bytes. PitchMSB, PitchLSB, RollMSB, RollLSB, TempMSB, TempLSB	1
0x65	Post OP Mode 1	OP Mode 1	1
0x71	Enter User Calibration Mode	No Response Data	0.3
0x72	Level Orientation	(X=forward, +Z=up) (default) No Response Data	0.3
0x73	Upright Sideways Orientation	(X=forward, Y=up) No Response Data	0.3
0x74	Upright Flat Front Orientation	(Z=forward, -X=up) No Response Data	0.3
0x75	Enter Run Mode	No Response Data	0.3
0x76	Enter Standby Mode	No Response Data	0.3
0x7E	Exit User Calibration Mode	No Response Data	50
0x82	Reset the Processor	No Response Data	500
0x83	Enter Sleep Mode	No Response Data	1
0x84	Exit Sleep Mode	No Response Data	20
0xE1	Read from EEPROM, RAM	1 binary data Byte	10
0xF1	Write to EEPROM, RAM	No Response Data. Data Settling Time	10

www.honeywell.com

### **Operational Mode Registers**

EEPROM registers 0x04 and 0x05 contain bits that are read for operational mode status and for setting the Run Mode measurement rate. The tables below describe the register contents and interpretation. It is recommended that Operational Mode Register 1 and 2 written only to change default orientation and update measurement rate.

Table 4 – Operational Mode Register 1 (EEPROM 0x04)

OM1_7	OM1_6	OM1_5	OM1_4	OM1_3	OM1_2	OM1_1	OM1_0
Comp(0)	Cal(0)	Filter(0)	Run(1)	Stdby(0)	UF(0)	UE(0)	Level(1)

Table 5 – Operational Mode Register 1 Bit Designations

Location	Name	Description
OM1_7	Comp	Calculating compass data if set. (read only)
OM1_6	Cal	Calculating calibration offsets if set. (read only)
OM1_5	Filter	IIR Heading Filter used if set.
OM1_4	Run	Run Mode if set.
OM1_3	Stdby	Standby Mode if set.
OM1_2	UF	Upright Front Orientation if set.
OM1_1	UE	Upright Edge Orientation if set.
OM1_0	Level	Level Orientation if set

### Table 6 – Operational Mode Register 2 (EEPROM 0x05)

OM2_7	OM2_6	OM2_5	OM2_4	OM2_3	OM2_2	OM2_1	OM2_0
(0)	(0)	(0)	(0)	(0)	(0)	MR1(0)	MR0(1)

### Table 7 – Operational Mode Register 2 Bit Designations

Location	Name	Description
OM2_7 to OM2_2	0	These bits must be cleared for correct operation.
OM2_1 to OM2_0	MR1, MR0	Measurement Rate 0,0 = 1Hz 0,1 = 5Hz (default) 1,0 = 10Hz 1,1 = Not Assigned

### **User Hard-Iron Calibration**

The HMC6343 provides a user calibration routine with the 0x71 command permitting entry into the calibration mode and the 0x7E command to exit the calibration mode.

After entering the calibration mode, rotate the device reasonably steady for 360 degrees about the Y (Left - Right) axis and then 360 degrees about Z (Up - Down) axis. During the first rotation, maintain the Y axis at Level as much as possible. Maintain the Z axis upright as much as possible during the second rotation and until the exit calibration

command is issued. The first rotation can also be done by rotating 360 degrees about X (Fore -Aft) axis. Then exit calibration.

The calibration routine collects these readings to correct for hard-iron distortions of the magnetic field. These hard-iron effects are due to magnetized materials nearby the HMC6343 part that in a fixed position with respect to the end user platform. An example would be the magnetized chassis or engine block of a vehicle in which the compass is mounted onto. Upon exiting the calibration mode, the resulting magnetometer offsets are updated.

### Example Communication

For basic power up and compassing using the defaults, the flowing order of operations is recommended:

- 1. Apply power to the VDD pins (nominally +3.3 volts)
- 2. Wait at least 500 milli-seconds for device initialization. The HMC6343 is in the default Run Mode.
- 3. Send 0x32 and 0x50 to command the Heading and Tilt Data to be clocked out next.
- 4. Wait at least 1 milli-second to allow the HMC6343 to process the command.
- Send 0x33 and clock back six more response Bytes from the HMC6343. These will be the Heading, Pitch and Roll Byte pairs; binary format in tenths of a degree with 2's compliment on pitch and roll angles. (0 to 3600 heading, ±900 pitch, and ±900 roll)
- 6. Repeat steps 3 5 every 200 milli-seconds or longer to get fresh data from the default 5Hz update rate.

### **ORDERING INFORMATION**

Ordering Number	Product	Packaging
HMC6343	3 axis Compass with Algorithms	Tubes
HMC6343-demo	Development Kit	Demo Board, USB Cable and Demo Software
HMC6343-eval	Evaluation Board	Board



### **FIND OUT MORE**

For more information on Honeywell's Magnetic Sensors visit us online at www.magneticsensors.com or contact us at 800-323-8295 (763-954-2474 internationally).

The application circuits herein constitute typical usage and interface of Honeywell product. Honeywell does not warranty or assume liability of customerdesigned circuits derived from this description or depiction.

Honeywell reserves the right to make changes to improve reliability, function or design. Honeywell does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.

U.S. Patents 4,441,072, 4,533,872, 4,569,742, 4,681,812, 4,847,584 and 6,529,114 apply to the technology described



**CAUTION: ESDS CAT. 1A** 

Honeywell 12001 Highway 55 Plymouth, MN 55441 Tel: 800-323-8295 www.honeywell.com

Form #900357 October 2008 ©2008 Honeywell International Inc.

## Honeywell



## LSM9DS0

iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer

Datasheet - production data



### Features

- 3 acceleration channels, 3 angular rate channels, 3 magnetic field channels
- $\pm 2/\pm 4/\pm 6/\pm 8/\pm 16$  g linear acceleration full scale
- ±2/±4/±8/±12 gauss magnetic full scale
- ±245/±500/±2000 dps angular rate full scale
- 16-bit data output
- SPI / I<sup>2</sup>C serial interfaces
- Analog supply voltage 2.4 V to 3.6 V
- Power-down mode / low-power mode
- Programmable interrupt generators
- Embedded self-test
- Embedded temperature sensor
- Embedded FIFO
- Position and motion detection functions
- Click/double-click recognition

This is information on a product in full production.

- Intelligent power saving for handheld devices
- ECOPACK<sup>®</sup>, RoHS and "Green" compliant

### Applications

- Indoor navigation
- Smart user interfaces
- Advanced gesture recognition
- · Gaming and virtual reality input devices
- Display/map orientation and browsing

### Description

The LSM9DS0 is a system-in-package featuring a 3D digital linear acceleration sensor, a 3D digital angular rate sensor, and a 3D digital magnetic sensor.

The LSM9DS0 has a linear acceleration full scale of  $\pm 2g/\pm 4g/\pm 6g/\pm 8g/\pm 16g$ , a magnetic field full scale of  $\pm 2/\pm 4/\pm 8/\pm 12$  gauss and an angular rate of  $\pm 245/\pm 500/\pm 2000$  dps.

The LSM9DS0 includes an  $I^2C$  serial bus interface supporting standard and fast mode (100 kHz and 400 kHz) and an SPI serial standard interface.

The system can be configured to generate interrupt signals on dedicated pins and is capable of motion and magnetic field detection. Thresholds and timing of interrupt generators are programmable by the end user.

Magnetic, accelerometer and gyroscope sensing can be enabled or set in power-down mode separately for smart power management.

The LSM9DS0 is available in a plastic land grid array package (LGA) and it is guaranteed to operate over an extended temperature range from -40 °C to +85 °C.

#### Table 1. Device summary

Part number	Temperature range [°C]	Package	Packing
LSM9DS0	-40 to +85	LGA-24	Tray
LSM9DS0TR	-40 to +85	LGA-24	Tape and reel

August 2013

LSM9DS0

1	Block	diagra	m and pin description	10	
	1.1	Block di	agram	10	
	1.2	Pin dese	cription	11	
2	Module specifications				
	2.1	Sensor	characteristics	13	
	2.2	Tempera	ature sensor characteristics	15	
	2.3	Electrica	Electrical characteristics		
	2.4	Commu	nication interface characteristics	17	
		2.4.1	SPI - serial peripheral interface	17	
		2.4.2	Sensor I <sup>2</sup> C - inter-IC control interface	18	
	2.5	Absolute	e maximum ratings	19	
3	Termi	nology		20	
	3.1	Set / res	set pulse	20	
	3.2	Sensitiv	ity	20	
		3.2.1	Linear acceleration sensor sensitivity	20	
		3.2.2	Magnetic sensor sensitivity	20	
		3.2.3	Angular rate sensitivity	20	
		3.2.4	Zero-g level	20	
		3.2.5	Zero-gauss level	21	
		3.2.6	Zero-rate level	21	
4	Funct	ionality	• • • • • • • • • • • • • • • • • • • •	22	
	4.1	Self-test	t	22	
		4.1.1	Accelerometer	22	
		4.1.2	Gyroscope	22	
	4.2	Linear a	cceleration main digital blocks	22	
		4.2.1	FIFO	22	
		4.2.2	Bypass mode	23	
		4.2.3	FIFO mode	23	
		4.2.4	Stream mode	23	
		4.2.5	Stream-to-FIFO mode	23	



2/74

		4.2.6	Retrieving data from FIFO		
	4.3	Gyrosco	ope digital main blocks		
		4.3.1	FIFO		
		4.3.2	Bypass mode		
		4.3.3	FIFO mode		
		4.3.4	Stream mode		
		4.3.5	Bypass-to-stream mode		
		4.3.6	Stream-to-FIFO mode		
		4.3.7	Retrieving data from FIFO		
	4.4	Temper	ature sensor		
	4.5	Factory	calibration		
5	Annli	cation b	ninte 20		
5		Extorno			
	5.1	Externa	i capacitors		
	5.2	Solderir			
	5.3	High cu	rrent wiring effects		
6	Digita	al interfa	aces		
	6.1	I <sup>2</sup> C seri	I <sup>2</sup> C serial interface		
		6.1.1	I <sup>2</sup> C operation		
	6.2	SPI bus	niterface		
		6.2.1	SPI read		
		6.2.2	SPI write		
		6.2.3	SPI read in 3-wire mode		
7	Rogis	tor mar	nning 38		
•	Regic		, , , , , , , , , , , , , , , , , , ,		
8	Regis	ster des	cription		
	8.1	WHO_A	M_I_G (0Fh)		
	8.2	CTRL I	REG1 G (20h)		
	8.3	CTRL I	REG2 G (21h)		
	84	CTRI I	REG3 G (22h) 43		
	8.5	CTRL I	REG4 G (23h) 44		
	8.6	CTRL I	REG5 G (24h)		
	0.0 9 7				
	0.7	OTATU	LINUL/DATAGAFTURE_G (2011)		
	8.8	STATUS	5_KEG_G (2/N) 45		



8.9	OUT_X_L_G (28h), OUT_X_H_G (29h)	46
8.10	OUT_Y_L_G (2Ah), OUT_Y_H_G (2Bh)	46
8.11	OUT_Z_L_G (2Ch), OUT_Z_H_G (2Dh)	46
8.12	FIFO_CTRL_REG_G (2Eh)	46
8.13	FIFO_SRC_REG_G (2Fh)	47
8.14	INT1_CFG_G (30h)	47
8.15	INT1_SRC_G (31h)	48
8.16	INT1_THS_XH_G (32h)	48
8.17	INT1_THS_XL_G (33h)	49
8.18	INT1_THS_YH_G (34h)	49
8.19	INT1_THS_YL_G (35h)	49
8.20	INT1_THS_ZH_G (36h)	49
8.21	INT1_THS_ZL_G (37h) 5	50
8.22	INT1_DURATION_G (38h) 5	50
8.23	OUT_TEMP_L_XM (05h), OUT_TEMP_H_XM (06h)	52
8.24	STATUS_REG_M (07h) 5	52
8.25	OUT_X_L_M (08h), OUT_X_H_M (09h) 8	52
8.26	OUT_Y_L_M (0Ah), OUT_Y_H_M (0Bh) 5	52
8.27	OUT_Z_L_M (0Ch), OUT_Z_H_M (0Dh) 8	53
8.28	WHO_AM_I_XM (0Fh) 5	53
8.29	INT_CTRL_REG_M (12h) 8	53
8.30	INT_SRC_REG_M (13h) 5	54
8.31	INT_THS_L_M (14h), INT_THS_H_M (15h)	54
8.32	OFFSET_X_L_M (16h), OFFSET_X_H_M (17h) 5	54
8.33	OFFSET_Y_L_M (18h), OFFSET_Y_H_M (19h)	54
8.34	OFFSET_Z_L_M (1Ah), OFFSET_Z_H_M (1Bh)	54
8.35	REFERENCE_X (1Ch)	55
8.36	REFERENCE_Y (1Dh) 5	55
8.37	REFERENCE_Z (1Eh) 5	55
8.38	CTRL_REG0_XM (1Fh) 5	55
8.39	CTRL_REG1_XM (20h) 5	55
8.40	CTRL_REG2_XM (21h) 8	56
8.41	CTRL_REG3_XM (22h) 8	57



CTRL_REG4_XM (23h)
CTRL_REG5_XM (24h) 59
CTRL_REG6_XM (25h) 59
CTRL_REG7_XM (26h) 60
STATUS_REG_A (27h) 61
OUT_X_L_A (28h), OUT_X_H_A (29h)
OUT_Y_L_A (2Ah), OUT_Y_H_A (2Bh)
OUT_Z_L_A (2Ch), OUT_Z_H_A (2Dh)
FIFO_CTRL_REG (2Eh) 62
FIFO_SRC_REG (2Fh) 62
INT_GEN_1_REG (30h) 63
INT_GEN_1_SRC (31h) 64
INT_GEN_1_THS (32h) 64
INT_GEN_1_DURATION (33h) 65
INT_GEN_2_REG (34h) 65
INT_GEN_2_SRC (35h) 66
INT_GEN_2_THS (36h) 67
INT_GEN_2_DURATION (37h) 67
CLICK_CFG (38h) 67
CLICK_SRC (39h) 68
CLICK_THS (3Ah) 68
TIME_LIMIT (3Bh) 69
TIME_LATENCY (3Ch) 69
TIME WINDOW (3Dh) 69
Act_THS (3Eh) 69
Act_DUR (3Fh)
age information
sion history



9

10

## List of tables

Table 1.	Device summary	. 1
Table 2.	Pin description	12
Table 3.	Sensor characteristics.	13
Table 4.	Temperature sensor electrical characteristics	15
Table 5.	Electrical characteristics	16
Table 6.	SPI slave timing values.	17
Table 7.	I <sup>2</sup> C slave timing values	18
Table 8.	Absolute maximum ratings	19
Table 9	Serial interface pin description	31
Table 10	l <sup>2</sup> C terminology	31
Table 11	Transfer when master is writing one byte to slave	32
Table 12	Transfer when master is writing multiple bytes to slave	32
Table 13	Transfer when master is receiving (reading) one byte of data from slave	32
Table 14	Transfer when master is receiving (reading) one byte of data from slave	32
Table 14.	Linear acceleration and magnetic sensor SAD+read/write patterns	33
Table 15.	Angular rate SAD+read/write patterns	22
Table 10.	Angulai Tale SAD+Teau/while pallettis	22
		30
Table 18.		41
Table 19.	OTRL_REG1_G register	41
Table 20.		41
Table 21.	DR and BW configuration setting	41
Table 22.	Power mode selection configuration.	42
Table 23.	CTRL_REG2_G register	42
Table 24.	CTRL_REG2_G description	42
Table 25.	High-pass filter mode configuration	43
Table 26.	High-pass filter cutoff frequency configuration (Hz)	43
Table 27.	CTRL_REG3_G register	43
Table 28.	CTRL_REG3_G description	43
Table 29.	CTRL_REG4_G register	44
Table 30.	CTRL_REG4_G description	44
Table 31.	Self-test mode configuration	44
Table 32.	CTRL_REG5_G register	44
Table 33.	CTRL_REG5_G description	44
Table 34.	REFERENCE/DATACAPTURE G register	45
Table 35.	REFERENCE/DATACAPTURE G description	45
Table 36.	STATUS REG G register	45
Table 37.	STATUS REG G description.	45
Table 38	FIFO CTRI REG G register	46
Table 39	FIFO CTRL REG G description	46
Table 40	FIFO mode configuration	46
Table 41	FIFO SBC REG G register	47
Table 42	FIFO_SRC_REG_G description	47
Table 43	INT1 CEG G register	47
Table 44	INT1_CFG_G description	47
Table 15	INT1_OF_C_C_conjeter	-+ <i>i</i> 10
Table 40.	INTI SPC C description	+0 ∕10
		40 10
		40
1 able 48.		4ð



56

List	of	tab	les
	•••		

Table 49.	INT1 THS XL G register
Table 50.	INT1_THS_XL_G description
Table 51.	INT1_THS_YH_G register
Table 52.	INT1_THS_YH_G description
Table 53.	INT1_THS_YL_G register
Table 54.	INT1_THS_YL_G description
Table 55.	INT1_THS_ZH_G register
Table 56.	INT1_THS_ZH_G description
Table 57.	INT1_THS_ZL_G register
Table 58.	INT1_THS_ZL_G description
Table 59.	INT1_DURATION_G register
Table 60.	INT1_DURATION_G description
Table 61.	STATUS_REG_M register
Table 62.	STATUS_REG_M description
Table 63.	WHO_AM_I_XM register
Table 64.	INT_CTRL_REG_M register
Table 65.	INT_CTRL_REG_M description
Table 66.	INT_SRC_REG_M register
Table 67.	INT_SRC_REG_M description
Table 68.	CTRL_REG0_XM register
Table 69.	CTRL_REG0_XM description
Table 70.	CTRL_REG1_XM register
Table 71.	CTRL_REG1_XM description
Table 72.	Acceleration data rate configuration
Table 73.	CTRL_REG2_XM register
Table 74.	CTRL_REG2_XM description
Table 75.	Acceleration anti-alias filter bandwidth
Table 76.	Acceleration full-scale selection
Table 77.	Self-test mode configuration
Table 78.	CTRL_REG3_XM register
Table 79.	CTRL_REG3_XM description
Table 80.	CTRL_REG4_XM register
Table 81.	CTRL_REG4_XM description
Table 82.	CTRL_REG5_XM register
Table 83.	CTRL_REG5_XM description
Table 84.	Magnetic data rate configuration
Table 85.	CTRL_REG6_XM register
Table 86.	CTRL_REG6_XM description
Table 87.	Magnetic full-scale selection
Table 88.	CTRL_REG7_XM register
Table 89.	CTRL_REG7_XM description
Table 90.	High-pass filter mode selection
Table 91.	
Table 92.	STATUS_REG_A register
Table 93.	STATUS_REG_A description
Table 94.	FIFO_CTRL_REG register
Table 95.	FIFO_CIRL_REG description
Table 96.	FIFO mode configuration
Table 97.	FIFO_SRC_REG register
Table 98.	FIFU_SKU_KEG description
Table 99.	INI_GEN_1_REG register
l'able 100.	IN I_GEN_1_REG description



Table 101	Interrupt mode 63
Table 102	INT GEN 1 SRC register 64
Table 103	INT_GEN_1_SRC description 64
Table 104	INT1 THS register 64
Table 105	INT1_THS description 64
Table 106	INT1_DURATION register 65
Table 107	INT1_DURATION description 65
Table 108.	INT GEN 2 REG register 65
Table 109	INT GEN 2 REG description 65
Table 110.	Interrupt mode
Table 111.	INT GEN 2 SRC register
Table 112.	INT GEN 2 SRC description
Table 113.	INT GEN 2 THS register
Table 114.	INT GEN 2 THS description
Table 115.	INT GEN 2 DURATION register
Table 116.	INT GEN 2 DURATION description
Table 117.	CLICK_CFG register
Table 118.	CLICK_CFG description
Table 119.	CLICK_SRC register
Table 120.	CLICK_SRC description
Table 121.	CLICK_THS register
Table 122.	CLICK_SRC description
Table 123.	TIME_LIMIT register
Table 124.	TIME_LIMIT description
Table 125.	TIME_LATENCY register
Table 126.	TIME_LATENCY description
Table 127.	TIME_WINDOW register
Table 128.	TIME_WINDOW description
Table 129.	TIME_WINDOW register
Table 130.	TIME_WINDOW description
Table 131.	Act_DUR register
Table 132.	Act_DUR description
Table 133.	LGA 4x4x1 mm 24-lead mechanical data (see note 1 and 2)
Table 134.	Document revision history



## List of figures

Figure 1.	Block diagram	10
Figure 2.	Pin connections	11
Figure 3.	SPI slave timing diagram	17
Figure 4.	I <sup>2</sup> C slave timing diagram	18
Figure 5.	Gyroscope block diagram	24
Figure 6.	Bypass mode	25
Figure 7.	FIFO mode	25
Figure 8.	Stream mode	26
Figure 9.	Bypass-to-stream mode	27
Figure 10.	Stream-to-FIFO mode	27
Figure 11.	LSM9DS0 electrical connections	<u>29</u>
Figure 12.	Read and write protocol	34
Figure 13.	SPI read protocol	35
Figure 14.	Multiple byte SPI read protocol (2-byte example)	35
Figure 15.	SPI write protocol	36
Figure 16.	Multiple byte SPI write protocol (2-byte example)	36
Figure 17.	SPI read protocol in 3-wire mode	37
Figure 18.	INT1_Sel and Out_Sel configuration block diagram4	45
Figure 19.	Wait bit disabled	51
Figure 20.	Wait bit enabled	51
Figure 21.	LGA 4x4x1 mm 24-lead outline	72



### 1 Block diagram and pin description

### 1.1 Block diagram



Figure 1. Block diagram

LSM9DS0

59



### LSM9DS0

### 1.2 Pin description





### LSM9DS0

Pin#	Name	Function			
1	Reserved	Leave unconnected			
2	Reserved	Connect to GND			
3	Reserved	Connect to GND			
4	Reserved	Connect to GND			
5	GND	0 V supply			
6	GND	0 V supply			
7	C1_XM	Capacitor connection (C1)			
8	SETC_XM	S/R capacitor connection (C2)			
9	SETP_XM	S/R capacitor connection (C2)			
10	DEN_G	Gyroscope data enable			
11	INT_G	Gyroscope programmable interrupt			
12	DRDY_G	Gyroscope data ready			
13	INT1_XM	Accelerometer and magnetic sensor interrupt 1			
14	INT2_XM	Accelerometer and magnetic sensor interrupt 2			
15	Vdd	Power supply			
16	Vdd	Power supply			
17	Vdd	Power supply			
18	Vdd_IO	Power supply for I/O pins			
19	CS_G	Gyroscope I <sup>2</sup> C/SPI mode selection 1: SPI idle mode / I <sup>2</sup> C communication enabled 0: SPI communication mode / I <sup>2</sup> C disabled			
20	CS_XM	Accelerometer and magnetic sensor SPI enabled I <sup>2</sup> C/SPI mode selection 1: SPI idle mode / I <sup>2</sup> C communication enabled 0: SPI communication mode / I <sup>2</sup> C disabled			
21	SCL SPC	I <sup>2</sup> C serial clock (SCL) SPI serial port clock (SPC)			
22	SDO_G SA0_G	Gyroscope serial data output (SDO) Angular rate sensor I <sup>2</sup> C less significant bit of the device address (SA0)			
23	SDO_XM SA0_XM	Accelerometer and magnetic sensor SPI serial data output (SDO) Accelerometer and magnetic sensor I <sup>2</sup> C less significant bit of the device address (SA0)			
24	SDA	I <sup>2</sup> C serial data (SDA)			

Table 2. Pin description



## 2 Module specifications

### 2.1 Sensor characteristics

@ Vdd = 3.0 V, T = 25 °C unless otherwise noted<sup>(a)</sup>

Symbol	Parameter	Test conditions	Min.	Typ. <sup>(1)</sup>	Max.	Unit
				±2		
	Linear acceleration			±4		
LA_FS				±6		g
				±8		
				±16		
				±2		
МЕС	Magnetic			±4		~~~~~
M_F5	measurement range			±8		gauss
				±12		
				±245		
G_FS	Angular rate measurement range			±500		dps
				±2000		
	Linear acceleration sensitivity	Linear acceleration FS = $\pm 2 g$		0.061		mg/LSB
		Linear acceleration FS = $\pm 4 g$		0.122		
LA_So		Linear acceleration FS = $\pm 6 g$		0.183		
		Linear acceleration FS = $\pm 8 g$		0.244		
		Linear acceleration FS = $\pm 16 g$		0.732		
		Magnetic FS = ±2 gauss		0.08		
MCN	Magnetic consitivity	Magnetic FS = ±4 gauss		0.16		mgauss/
	Magnetic sensitivity	Magnetic FS = ±8 gauss		0.32		LSB
		Magnetic FS = ±12 gauss		0.48		
		Angular rate FS = ±245 dps		8.75		
G_So	Angular rate sensitivity	Angular rate FS = ±500 dps		17.50		mdps/ digit
		Angular rate FS = ±2000 dps		70		uigit
LA_TCSo	Linear acceleration sensitivity change vs. temperature	From -40 °C to +85 °C		±1.5		%
M_TCSo	Magnetic sensitivity change vs. temperature	From -40 °C to +85 °C		±3		%

Table	3.	Sensor	characteristics	5
-------	----	--------	-----------------	---

a. The product is factory calibrated at 3.0 V. The operational power supply range is from 2.4 V to 3.6 V.



#### **Module specifications**

#### LSM9DS0

Symbol	Parameter	Test conditions	Min.	Typ. <sup>(1)</sup>	Max.	Unit
G_SoDr	Angular rate sensitivity change vs. temperature	From -40 °C to +85 °C		±2		%
LA_TyOff	Linear acceleration typical zero-g level offset accuracy <sup>(3)(4)</sup>			±60		mg
		FS = 245 dps		±10		
G_TyOff	Angular rate	FS = 500 dps		±15		dps
		FS = 2000 dps		±25		
LA_TCOff	Linear acceleration zero-g level change vs. temperature	Max delta from 25 °C		±0.5		m <i>g</i> /°C
G_TCOff	Zero-rate level change vs. temperature			±0.05		dps/°C
M_EF	Maximum exposed field	No perming effect on zero reading			10000	gauss
M_DF	Magnetic disturbing field	Sensitivity starts to degrade. Automatic S/R pulse restores the sensitivity <sup>(5)</sup>			20	gauss
LA_ST	Linear acceleration self-test positive difference <sup>(6)(7)</sup>	±2 g range, X, Y, Z-axis AST1:0 = 01 see <i>Table 74</i>	60		1700	mg
G_ST		FS = 245 dps	20		250	
	Angular rate self-test output change <sup>(8)(9)</sup>	FS = 500 dps	70		400	dps
		FS = 2000 dps	150		1000	
Тор	Operating temperature range		-40		+85	°C

Table 3. Sensor characteristics (continued)

1. Typical specifications are not guaranteed

2. Verified by wafer level test and measurement of initial offset and sensitivity

3. Typical zero-g level offset value after MSL3 preconditioning

4. Offset can be eliminated by enabling the built-in high-pass filter

5. Set / Reset Pulse is automatically applied at each conversion cycle

"Self-test output change" is defined as: OUTPUT[mg]<sub>(</sub>CTRL\_REG2\_XM (21h) AST1:0 enabled) - OUTPUT[mg]<sub>(</sub>CTRL\_REG2\_XM (21h) AST1:0 disabled)

7. For polarity refer to Table 77: Self-test mode configuration

"Self-test output change" is defined as: OUTPUT[mg]<sub>(</sub>CTRL\_REG4\_G (23h) ST1:0 enabled) - OUTPUT[mg]<sub>(</sub>CTRL\_REG4\_G (23h) ST1:0 disabled)

9. For polarity refer to Table 31: Self-test mode configuration



### 2.2 Temperature sensor characteristics

The electrical characteristics concerning the temperature sensor are given in the table below.

@ Vdd = 3.0 V, T=25 °C unless otherwise noted.

Symbol	Parameter	Test conditions	Min.	Тур. <sup>(1)</sup>	Max.	Unit
TSDr	Temperature sensor output change vs. temperature			8		LSB/°C
TODR	Temperature refresh rate	-		M_ODR [2:0] <sup>(2)</sup>		Hz
Тор	Operating temperature range		-40		+85	°C

Table 4. Temperature sensor electrical characteristic	CS
---	----

1. Typical specifications are not guaranteed.

2. Refer to Table 84: Magnetic data rate configuration.



### 2.3 Electrical characteristics

@ Vdd = 3.0V, T = 25 °C unless otherwise noted<sup>(b)</sup>

Symbol	Parameter	Test conditions	Min.	Typ. <sup>(1)</sup>	Max.	Unit
Vdd	Supply voltage		2.4		3.6	V
Vdd_IO	Module power supply for I/O		1.71	1.8	Vdd+0.1	
ldd_XM	Current consumption of the accelerometer and magnetic sensor in normal mode <sup>(2)</sup>		HR setting CTRL_REG5 _XM (M_RES [1,0]) = 11b, see CTRL_REG5 _XM (24h)	350		μA
ldd_G	Gyroscope current consumption in normal mode <sup>(3)</sup>			6.1		mA
ldd_G_LP	Gyroscope supply current in sleep mode <sup>(4)</sup>			2		mA
ldd_Pdn	Current consumption in power-down mode <sup>(5)</sup>			6		μA
VIH	Digital high-level input voltage		0.8*Vdd_IO			V
VIL	Digital low-level input voltage				0.2*Vdd_IO	V
VOH	High-level output voltage		0.9*Vdd_IO			V
VOL	Low-level output voltage				0.1*Vdd_IO	V
Тор	Operating temperature range		-40		+85	°C

### Table 5. Electrical characteristics

1. Typical specifications are not guaranteed

2. Magnetic sensor setting ODR =6.25 Hz, Accelerometer sensor ODR =50 Hz, gyroscope in power-down mode

3. Accelerometer and magnetic sensor in power-down mode

4. Sleep mode introduces a faster turn-on time compared to power-down mode. Accelerometer and magnetic sensor in power-down mode.

5. Linear accelerometer, magnetic sensor and gyroscope in power-down mode

b. LSM9DS0 is factory calibrated at 3.0 V

LSM9DS0

#### LSM9DS0

### 2.4 Communication interface characteristics

### 2.4.1 SPI - serial peripheral interface

Subject to general operating conditions for Vdd and Top.

Symbol	Parameter	Valu	Unit	
Symbol	Farameter	Min	Мах	Unit
t <sub>c(SPC)</sub>	SPI clock cycle	100		ns
f <sub>c(SPC)</sub>	SPI clock frequency		10	MHz
t <sub>su(CS)</sub>	CS setup time	5		
t <sub>h(CS)</sub>	CS hold time	20		
t <sub>su(SI)</sub>	SDI input setup time	5		
t <sub>h(SI)</sub>	SDI input hold time	15		ns
t <sub>v(SO)</sub>	SDO valid output time		50	
t <sub>h(SO)</sub>	SDO output hold time	5		
t <sub>dis(SO)</sub>	SDO output disable time		50	

#### Table 6. SPI slave timing values

1. Values are guaranteed at 10 MHz clock frequency for SPI with both 4 and 3 wires, based on characterization results, not tested in production.





*Note: Measurement points are done at* 0.2·Vdd\_IO and 0.8·Vdd\_IO, for both input and output ports.



### 2.4.2 Sensor I<sup>2</sup>C - inter-IC control interface

Subject to general operating conditions for Vdd and Top.

Symbol	Peremeter	I <sup>2</sup> C standard mode <sup>(1)</sup>		I <sup>2</sup> C fast mode <sup>(1)</sup>		Unit
Symbol	Parameter	Min	Max	Min	Max	
f <sub>(SCL)</sub>	SCL clock frequency	0	100	0	400	kHz
t <sub>w(SCLL)</sub>	SCL clock low time	4.7		1.3		
t <sub>w(SCLH)</sub>	SCL clock high time	4.0		0.6		μ5
t <sub>su(SDA)</sub>	SDA setup time	250		100		ns
t <sub>h(SDA)</sub>	SDA data hold time	0	3.45	0	0.9	μs
t <sub>h(ST)</sub>	START condition hold time	4		0.6		
t <sub>su(SR)</sub>	Repeated START condition setup time	4.7		0.6		
t <sub>su(SP)</sub>	STOP condition setup time	4		0.6		- μs
t <sub>w(SP:SR)</sub>	Bus free time between STOP and START condition	4.7		1.3		

Table 7.	I <sup>2</sup> C slav	ve timing	values
----------	-----------------------	-----------	--------

1. Data based on standard I<sup>2</sup>C protocol requirement, not tested in production.



### Figure 4. I<sup>2</sup>C slave timing diagram



Measurement points are done at 0.2 Vdd\_IO and 0.8 Vdd\_IO, for both ports.


## 2.5 Absolute maximum ratings

Stresses above those listed as "absolute maximum ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Symbol	Ratings	Maximum value	Unit
Vdd	Supply voltage	-0.3 to 4.8	V
Vdd_IO	I/O pins supply voltage	-0.3 to 4.8	V
Vin	Input voltage on any control pin (SCL/SPC, SDA, SDO_XM/SA0_XM, SDO_G/SA0_G, CS_G, CS_XM, DEN_G)	-0.3 to Vdd_IO +0.3	V
Δ	Acceleration (any axis, nowered $V(dd = 2.5 V)$	3,000 for 0.5 ms	g
POW	Acceleration (any axis, powered, volu = 2.5 v)	10,000 for 0.1 ms	g
٨	Acceleration (any axis unnowered)	3,000 for 0.5 ms	g
TUNP	Acceleration (any axis, unpowered)	10,000 for 0.1 ms	g
T <sub>OP</sub>	Operating temperature range	-40 to +85	°C
T <sub>STG</sub>	Storage temperature range	-40 to +125	°C
ESD	Electrostatic discharge protection	2 (HBM)	kV

Table 8. Absolute maximum ratings

Note:

Supply voltage on any pin should never exceed 4.8 V





X

This is an electrostatic-sensitive device (ESD), improper handling can cause permanent damage to the part.



# 3 Terminology

### 3.1 Set / reset pulse

The set / reset pulse is an automatic operation performed before each magnetic acquisition cycle to degauss the sensor and to ensure alignment of the magnetic dipoles and thus the linearity of the sensor itself.

# 3.2 Sensitivity

The methods to determine sensitivity and offset are given below in the following paragraphs.

### 3.2.1 Linear acceleration sensor sensitivity

Sensitivity describes the gain of the sensor and can be determined by applying 1 g acceleration to it. As the sensor can measure DC accelerations this can be done easily by pointing the axis of interest towards the center of the Earth, noting the output value, rotating the sensor by 180 degrees (pointing to the sky) and noting the output value again. By doing so,  $\pm 1 g$  acceleration is applied to the sensor. Subtracting the larger output value from the smaller one, and dividing the result by 2, leads to the actual sensitivity of the sensor. This value changes very little over temperature and time. The sensitivity tolerance describes the range of sensitivities of a large population of sensors.

#### 3.2.2 Magnetic sensor sensitivity

Sensitivity describes the gain of the sensor and can be determined by applying a magnetic field of 1 *gauss* to it.

### 3.2.3 Angular rate sensitivity

An angular rate gyroscope is a device that produces a positive-going digital output for counter-clockwise rotation around the sensitive axis considered. Sensitivity describes the gain of the sensor and can be determined by applying a defined angular velocity to it. This value changes very little over temperature and time.

### 3.2.4 Zero-g level

The zero-*g* level offset (TyOff) describes the deviation of an actual output signal from the ideal output signal if no acceleration is present. A sensor in a steady state on a horizontal surface will measure 0 *g* for the X-axis and 0 *g* for the Y-axis whereas the Z-axis will measure 1 *g*. The output is ideally in the middle of the dynamic range of the sensor (content of OUT registers 00h, data expressed as two's complement number). A deviation from the ideal value in this case is called Zero-*g* offset. Offset is to some extent a result of stress to the MEMS sensor and therefore the offset can slightly change after mounting the sensor onto a printed circuit board or exposing it to extensive mechanical stress. Offset changes little over temperature, see "Zero-*g* level change vs. temperature" (LA\_TCOff in *Table 3*). The Zero-*g* level tolerance (TyOff) describes the standard deviation of the range of Zero-*g* levels of a population of sensors.



### 3.2.5 Zero-gauss level

The zero-gauss level offset describes the deviation of an actual output signal from the ideal output if no magnetic field is present. Thanks to the Set/Reset Pulse and to the magnetic sensor readout chain, the offset is dynamically cancelled. The Zero-gauss level does not show any dependency on temperature or power supply.

### 3.2.6 Zero-rate level

The zero-rate level describes the actual output signal if there is no angular rate present. The zero-rate level of highly accurate MEMS sensors is, to some extent, a result of stress to the sensor and therefore the zero-rate level can slightly change after mounting the sensor onto a printed circuit board or after exposing it to extensive mechanical stress. This value changes very little over temperature and time.



# 4 Functionality

The LSM9DS0 is a system-in-package featuring a 3D digital accelerometer, a 3D digital magnetometer, and a 3D digital gyroscope.

The device includes specific sensing elements and two IC interfaces capable of measuring both the acceleration/magnetometer and angular rate applied to the module and to provide a signal to external applications through an SPI/I<sup>2</sup>C serial interface.

The various sensing elements are manufactured using specialized micromachining processes, while the IC interfaces are developed using a CMOS technology that allows the design of a dedicated circuit which is trimmed to better match the sensing element characteristics.

The LSM9DS0 may also be configured to generate an inertial *wake-up* and *free-fall* interrupt signal according to a programmed acceleration event along the enabled axes.

### 4.1 Self-test

### 4.1.1 Accelerometer

The self-test allows the linear acceleration sensor functionality to be tested without moving it. The self-test function is off when the self-test bit (ST) is programmed to '0'. When the self-test bit is programmed to '1' an actuation force is applied to the sensor, simulating a definite input acceleration. In this case the sensor outputs exhibit a change in their DC levels which are related to the selected full scale through the device sensitivity. When the self-test is activated, the device output level is given by the algebraic sum of the signals produced by the acceleration acting on the sensor and by the electrostatic test-force. If the output signals change within the amplitude specified inside *Section 2.1: Sensor characteristics*, then the sensor is working properly and the parameters of the interface chip are within the defined specifications.

### 4.1.2 Gyroscope

The self-test allows to test the mechanical and electric part of the sensor, allowing the seismic mass to be moved by means of an electrostatic test-force. When the ST is activated by the IC, an actuation force is applied to the sensor, emulating a definite Coriolis force. In this case the sensor output will exhibit an output change.

When the ST is active, the device output is given by the algebraic sum of the signals produced by the velocity acting on the sensor and by the electrostatic test-force.

For polarity please refer to Table 31: Self-test mode configuration.

## 4.2 Linear acceleration main digital blocks

#### 4.2.1 FIFO

The LSM9DS0 embeds 32 slots of data FIFO for each of the three output channels: X, Y and Z. This allows consistent power saving for the system, since the host processor does not need to continuously poll data from the sensor, but it can wake up only when needed



and burst the significant data out from the FIFO. This buffer can work accordingly in four different modes: Bypass mode, FIFO mode, Stream mode and Stream-to-FIFO mode. Each mode is selected by the FIFO\_MODE bits in *FIFO\_SRC\_REG (2Fh)*. Programmable watermark level, FIFO\_Empty or FIFO\_Full events can be enabled to generate dedicated interrupts on the INT1\_XM/INT2\_XM pin (configured through *FIFO\_SRC\_REG (2Fh)*).

#### 4.2.2 Bypass mode

In Bypass mode, the FIFO is not operational and for this reason it remains empty. For each channel only the first address is used. The remaining FIFO slots are empty.

### 4.2.3 FIFO mode

In FIFO mode, data from the X, Y and Z channels are stored in the FIFO. A watermark interrupt can be enabled (FIFO\_WTMK\_EN bit in *FIFO\_CTRL\_REG (2Eh)*) in order to be raised when the FIFO is filled to the level specified in the FIFO\_WTMK\_LEVEL bits of *FIFO\_CTRL\_REG (2Eh)*. The FIFO continues filling until it is full (32 slots of data for X, Y and Z). When full, the FIFO stops collecting data from the input channels.

#### 4.2.4 Stream mode

In Stream mode, data from the X, Y and Z measurements are stored in the FIFO. A watermark interrupt can be enabled and set as in FIFO mode. The FIFO continues filling until it is full (32 slots of data for X, Y and Z). When full, the FIFO discards the older data as the new data arrives.

#### 4.2.5 Stream-to-FIFO mode

In Stream-to-FIFO mode, data from the X, Y and Z measurements is stored in the FIFO. A watermark interrupt can be enabled (FIFO\_WTMK\_EN bit in *FIFO\_CTRL\_REG (2Eh)*) in order to be raised when the FIFO is filled to the level specified in the FIFO\_WTMK\_LEVEL bits of *FIFO\_CTRL\_REG (2Eh)*. The FIFO continues filling until it is full (32 slots of 8-bit data for X, Y and Z). When full, the FIFO discards the older data as the data new arrives. Once a trigger event occurs, the FIFO starts operating in FIFO mode.

#### 4.2.6 Retrieving data from FIFO

A read operation to the  $OUT\_X\_L\_A$  (28h),  $OUT\_X\_H\_A$  (29h),  $OUT\_Y\_L\_A$  (2Ah),  $OUT\_Y\_H\_A$  (2Bh) or  $OUT\_Z\_L\_A$  (2Ch),  $OUT\_Z\_H\_A$  (2Dh) registers provides the data stored in the FIFO. Each time data is read from the FIFO, the oldest X, Y and Z data are placed in the  $OUT\_X\_L\_A$  (28h),  $OUT\_X\_H\_A$  (29h),  $OUT\_Y\_L\_A$  (2Ah),  $OUT\_Y\_H\_A$ (2Bh) and  $OUT\_Z\_L\_A$  (2Ch),  $OUT\_Z\_H\_A$  (2Dh) registers and both single read and read\\_burst operations can be used.



## 4.3 Gyroscope digital main blocks



Figure 5. Gyroscope block diagram

### 4.3.1 FIFO

The LSM9DS0 embeds 32 slots of 16-bit data FIFO for each of the three output channels: yaw, pitch and roll. This allows consistent power saving for the system, since the host processor does not need to continuously poll data from the sensor, but can wake up only when needed and burst the significant data out from the FIFO. This buffer can work accordingly in five different modes: Bypass mode, FIFO mode, Stream mode, Bypass-to-Stream mode and Stream-to-FIFO mode. Each mode is selected by the FIFO\_MODE bits in *FIFO\_CTRL\_REG\_G (2Eh)*. A programmable watermark level, FIFO\_Empty or FIFO\_Full events can be enabled to generate dedicated interrupts on the DRDY\_G pin (configured through *CTRL\_REG\_G (2Eh)*. The watermark level can be configured to WTM4:0 in *FIFO\_CTRL\_REG\_G (2Eh)*.

### 4.3.2 Bypass mode

In Bypass mode, the FIFO is not operational and for this reason it remains empty. As described in *Figure 6*, for each channel only the first address is used. The remaining FIFO slots are empty. When new data is available, the old data is overwritten.





### 4.3.3 FIFO mode

In FIFO mode, data from the yaw, pitch and roll channels is stored in the FIFO. A watermark interrupt can be enabled (I2\_WMK bit in *CTRL\_REG3\_G* (22*h*)) in order to be raised when the FIFO is filled to the level specified in the WTM 4:0 bits of *FIFO\_CTRL\_REG\_G* (22*h*). The FIFO continues filling until it is full (32 slots of 16-bit data for yaw, pitch and roll). When full, the FIFO stops collecting data from the input channels. To restart data collection, *FIFO\_CTRL\_REG\_G* (22*h*) must be written back to Bypass mode.

FIFO mode is represented in Figure 7.



Figure 7. FIFO mode

57

DocID024763 Rev 2

25/74

75

### 4.3.4 Stream mode

In Stream mode, data from the yaw, pitch and roll measurements is stored in the FIFO. A watermark interrupt can be enabled and set as in FIFO mode. The FIFO continues filling until it is full (32 slots of 16-bit data for yaw, pitch and roll). When full, the FIFO discards the older data as the new data arrives. Programmable watermark level events can be enabled to generate dedicated interrupts on the DRDY\_G pin (configured through *CTRL\_REG3\_G* (22h).

Stream mode is represented in Figure 8.



Figure 8. Stream mode



### 4.3.5 Bypass-to-stream mode

In Bypass-to-stream mode, the FIFO starts operating in Bypass mode and once a trigger event occurs (related to *INT1\_CFG\_G (30h)* events) the FIFO starts operating in Stream mode. Refer to *Figure 9* below.



#### 4.3.6 Stream-to-FIFO mode

In Stream-to-FIFO mode, data from the yaw, pitch and roll measurement is stored in the FIFO. A watermark interrupt can be enabled on pin DRDY\_G by setting the I2\_WTM bit in *CTRL\_REG3\_G (22h)* to be raised when the FIFO is filled to the level specified in the WTM4:0 bits of *FIFO\_CTRL\_REG\_G (2Eh)*. The FIFO continues filling until it is full (32 slots of 16-bit data for yaw, pitch and roll). When full, the FIFO discards the older data as the new data arrives. Once a trigger event occurs (related to *INT1\_CFG\_G (30h)* events), the FIFO starts operating in FIFO mode. Refer to *Figure 10*.



Figure 10. Stream-to-FIFO mode

57

### 4.3.7 Retrieving data from FIFO

A read operation from the  $OUT_X_L_G$  (28h),  $OUT_X_H_G$  (29h),  $OUT_Y_L_G$  (2Ah),  $OUT_Y_H_G$  (2Bh) or  $OUT_Z_L_G$  (2Ch),  $OUT_Z_H_G$  (2Dh) registers provides the data stored in the FIFO. Each time data is read from the FIFO, the oldest pitch, roll and yaw data are placed in the  $OUT_X_L_G$  (28h),  $OUT_X_H_G$  (29h),  $OUT_Y_L_G$  (2Ah),  $OUT_Y_H_G$  (2Bh) and  $OUT_Z_L_G$  (2Ch),  $OUT_Z_H_G$  (2Dh) registers and both single read and read\_burst (X,Y & Z with auto-incremental address) operations can be used. When data included in  $OUT_Z_H_G$  is read, the system again starts to read information from addr  $OUT_X_L_G$ .

### 4.4 Temperature sensor

The LSM9DS0 features an embedded temperature sensor.

Temperature data can be enabled by setting the TEMP\_EN bit in the *CTRL\_REG5\_XM* (24h) register to 1.

Both OUT\_TEMP\_H\_XM and OUT\_TEMP\_L\_XM registers must be read.

Temperature data is stored inside *OUT\_TEMP\_L\_XM (05h)*, *OUT\_TEMP\_H\_XM (06h)* as two's complement data in 12-bit format, right justified.

The output data rate of the temperature sensor is set by M\_ODR in CTRL\_REG5\_XM (24h) and is equal to the magnetic sensor output data rate.

### 4.5 Factory calibration

The IC interface is factory calibrated. The trimming values are stored inside the non-volatile memory of the device. Any time the device is turned on, the trimming parameters are downloaded into the registers to be used during normal operation. This allows the using the device without further calibration.



# 5 Application hints



## 5.1 External capacitors

The C1 and C2 external capacitors should be low SR value, ceramic type construction (typ recommended value 200 mOhm). Reservoir capacitor C1 is nominally 4.7  $\mu$ F in capacitance, with the set/reset capacitor C2 nominally 0.22  $\mu$ F in capacitance.

The device core is supplied through the Vdd line. Power supply decoupling capacitors (C4 = 100 nF ceramic, C3 = 10  $\mu$ F Al) should be placed as near as possible to the supply pin of the device (common design practice). All the voltage and ground supplies must be present at the same time to have proper behavior of the IC (refer to *Figure 11*).

The functions of the device and the measured acceleration/magnetic field data are selectable and accessible through the  $I^2C$  / SPI interfaces.

The functions, the threshold and the timing of the three interrupt pins (INT\_G, INT1\_XM and INT2\_XM) can be completely programmed by the user through the  $I^2C$  / SPI interfaces.



### 5.2 Soldering information

The LGA package is compliant with the ECOPACK<sup>®</sup>, RoHS and "Green" standard. It is qualified for soldering heat resistance according to JEDEC J-STD-020.

Leave "Pin 1 Indicator" unconnected during soldering.

Land pattern and soldering recommendation are available at <u>www.st.com/mems</u>.

### 5.3 High current wiring effects

High current in wiring and printed circuit traces can be culprits in causing errors in magnetic field measurements for compassing.

Conductor-generated magnetic fields will add to the Earth's magnetic field leading to errors in compass-heading computation.

Keep currents higher than 10 mA a few millimeters further away from the sensor IC.



# 6 Digital interfaces

The registers embedded in the LSM9DS0 may be accessed through both the  $I^2C$  and SPI serial interfaces. The latter may be SW-configured to operate either in 3-wire or 4-wire interface mode.

The serial interfaces are mapped to the same pins. To select/exploit the  $I^2C$  interface, the CS line must be tied high (i.e connected to Vdd\_IO).

Pin name	Pin description
CS	I <sup>2</sup> C/SPI mode selection 1: SPI idle mode / I <sup>2</sup> C communication enabled 0: SPI communication mode / I <sup>2</sup> C disabled
SCL/SPC	I <sup>2</sup> C serial clock (SCL) SPI serial port clock (SPC)
SDA/SDI/SDO	I <sup>2</sup> C serial data (SDA) SPI serial data input (SDI) 3-wire interface serial data output (SDO)
SDO	SPI serial data output (SDO) I <sup>2</sup> C less significant bit of the device address

Table 9. Serial interface pin description

# 6.1 I<sup>2</sup>C serial interface

The LSM9DS0  $I^2C$  is a bus slave. The  $I^2C$  is employed to write data into registers whose content can also be read back.

The relevant  $I^2C$  terminology is given in the table below.

Term	Description
Transmitter	The device which sends data to the bus
Receiver	The device which receives data from the bus
Master	The device which initiates a transfer, generates clock signals and terminates a transfer
Slave	The device addressed by the master

Table 10. I<sup>2</sup>C terminology

There are two signals associated with the  $I^2C$  bus: the serial clock line (SCL) and the serial data line (SDA). The latter is a bidirectional line used for sending and receiving the data to/from the interface. Both lines must be connected to Vdd\_IO through external pull-up resistors. When the bus is free, both lines are high.

The  $I^2C$  interface is compliant with fast mode (400 kHz)  $I^2C$  standards as well as with normal mode.



### 6.1.1 I<sup>2</sup>C operation

The transaction on the bus is started through a START (ST) signal. A START condition is defined as a HIGH to LOW transition on the data line while the SCL line is held HIGH. After this has been transmitted by the master, the bus is considered busy. The next byte of data transmitted after the start condition contains the address of the slave in the first 7 bits and the eighth bit tells whether the master is receiving data from the slave or transmitting data to the slave. When an address is sent, each device in the system compares the first seven bits after a start condition with its own address. If they match, the device considers itself addressed by the master.

Data transfer with acknowledge is mandatory. The transmitter must release the SDA line during the acknowledge pulse. The receiver must then pull the data line LOW so that it remains stable low during the HIGH period of the acknowledge clock pulse. A receiver which has been addressed is obliged to generate an acknowledge after each byte of data received.

The I<sup>2</sup>C embedded inside the LSM9DS0 behaves like a slave device and the following protocol must be adhered to. After the start condition (ST) a slave address is sent, once a slave acknowledge (SAK) has been returned, an 8-bit sub-address (SUB) will be transmitted: the 7 LSb represents the actual register address while the MSB enables the address auto increment. If the MSb of the SUB field is '1', the SUB (register address) will be automatically increased to allow multiple data read/writes.

Master	ST	SAD + W		SUB		DATA		SP
Slave			SAK		SAK		SAK	

#### Table 11. Transfer when master is writing one byte to slave

Table 12. Transfer when master is writing multiple bytes to slave										
Master	ST	SAD + W		SUB		DATA		DATA		SP
Slave			SAK		SAK		SAK		SAK	

#### Table 13. Transfer when master is receiving (reading) one byte of data from slave

Master	ST	SAD + W		SUB		SR	SAD + R			NMAK	SP
Slave			SAK		SAK			SAK	DATA		

#### Table 14. Transfer when master is receiving (reading) multiple bytes of data from slave

Master	ST	SAD+W		SUB		SR	SAD+R			MAK		MAK		NMAK	SP
Slave			SAK		SAK			SAK	DATA		DAT A		DAT A		

Data are transmitted in byte format (DATA). Each data transfer contains 8 bits. The number of bytes transferred per transfer is unlimited. Data is transferred with the Most Significant bit (MSb) first. If a receiver can't receive another complete byte of data until it has performed



some other function, it can hold the clock line, SCL LOW to force the transmitter into a wait state. Data transfer only continues when the receiver is ready for another byte and releases the data line. If a slave receiver doesn't acknowledge the slave address (i.e. it is not able to receive because it is performing some real-time function) the data line must be left HIGH by the slave. The master can then abort the transfer. A LOW to HIGH transition on the SDA line while the SCL line is HIGH is defined as a STOP condition. Each data transfer must be terminated by the generation of a STOP (SP) condition.

In order to read multiple bytes, it is necessary to assert the most significant bit of the subaddress field. In other words, SUB(7) must be equal to 1 while SUB(6-0) represents the address of first register to be read.

In the presented communication format MAK is Master Acknowledge and NMAK is No Master Acknowledge.

Default address:

The **SDO/SA0** pins (SDO\_XM/SA0\_XM or SDO\_G/SA0\_G) can be used to modify the least significant bit of the device address. If the SA0 pin is connected to the voltage supply, LSb is '1' (ex. address 0011101b) else if SA0 pad is connected to ground, the LSb value is '0' (ex. address 0011110b).

The slave address is completed with a Read/Write bit. If the bit was '1' (Read), a repeated START (SR) condition will have to be issued after the two sub-address bytes; if the bit is '0' (Write) the master will transmit to the slave with the direction unchanged. *Table 15* and *Table 16* explain how the SAD+Read/Write bit pattern is composed, listing all the possible configurations.

Linear acceleration and magnetic sensor address:

Command	SDO_XM/SA0_XM pin	SAD[6:2]	SAD[1:0]	R/W	SAD+R/W
Read	0	00111	10	1	00111101 (3D)
Write	0	00111	10	0	00111100 (3C)
Read	1	00111	01	1	00111011 (3B)
Write	1	00111	01	0	00111010 (3A)

Table 15. Linear acceleration and magnetic sensor SAD+read/write patterns

Angular rate sensor address:

Command	SAD[6:1]	SAD[0] = SDO_G/SA0_G pin	R/W	SAD+R/W
Read	110101	0	1	11010101 (D5h)
Write	110101	0	0	11010100 (D4h)
Read	110101	1	1	11010111 (D7h)
Write	110101	1	0	11010110 (D6h)



### 6.2 SPI bus interface

The SPI is a bus slave. The SPI allows writing and reading the registers of the device.

The serial interface interacts with the outside world through 4 wires: **CS**, **SPC**, **SDI** and **SDO**.

Figure 12. Read and write protocol



**CS** is the Serial Port Enable and is controlled by the SPI master. It goes low at the start of the transmission and goes back high at the end. **SPC** is the Serial Port Clock and it is controlled by the SPI master. It is stopped high when **CS** is high (no transmission). **SDI** and **SDO** are respectively the Serial Port Data Input and Output. Those lines are driven at the falling edge of **SPC** and should be captured at the rising edge of **SPC**.

Both the Read Register and Write Register commands are completed in 16 clock pulses or in multiples of 8 in case of multiple bytes read/write. Bit duration is the time between two falling edges of **SPC**. The first bit (bit 0) starts at the first falling edge of **SPC** after the falling edge of **CS** while the last bit (bit 15, bit 23, ...) starts at the last falling edge of SPC just before the rising edge of **CS**.

**bit 0**:  $R\overline{W}$  bit. When 0, the data DI(7:0) is written to the device. When 1, the data DO(7:0) from the device is read. In the latter case, the chip will drive **SDO** at the start of bit 8.

*bit 1*: MS bit. When 0, the address remains unchanged in multiple read/write commands. When 1, the address will be auto-incremented in multiple read/write commands.

bit 2-7: address AD(5:0). This is the address field of the indexed register.

*bit 8-15*: data DI(7:0) (write mode). This is the data that will be written to the device (MSb first).

*bit 8-15*: data DO(7:0) (read mode). This is the data that will be read from the device (MSb first).

In multiple read/write commands, further blocks of 8 clock periods will be added. When the  $M\overline{S}$  bit is 0, the address used to read/write data remains the same for every block. When the  $M\overline{S}$  bit is 1, the address used to read/write data is incremented at every block.

The function and the behavior of SDI and SDO remain unchanged.



### 6.2.1 SPI read



The SPI read command is performed with 16 clock pulses. The multiple byte read command is performed by adding blocks of 8 clock pulses to the previous one.

bit 0: READ bit. The value is 1.

*bit 1*:  $M\overline{S}$  bit. When 0, does not increment address; when 1, increments address in multiple reads.

bit 2-7: address AD(5:0). This is the address field of the indexed register.

*bit 8-15*: data DO(7:0) (read mode). This is the data that will be read from the device (MSb first).

bit 16-... : data DO(...-8). Further data in multiple byte reads.



Figure 14. Multiple byte SPI read protocol (2-byte example)



### 6.2.2 SPI write



The SPI Write command is performed with 16 clock pulses. The multiple byte write command is performed by adding blocks of 8 clock pulses to the previous one.

bit 0: WRITE bit. The value is 0.

*bit 1*:  $M\overline{S}$  bit. When 0, does not increment address; when 1, increments address in multiple writes.

bit 2 -7: address AD(5:0). This is the address field of the indexed register.

*bit 8-15*: data DI(7:0) (write mode). This is the data that will be written to the device (MSb first).

bit 16-... : data DI(...-8). Further data in multiple byte writes.



### 6.2.3 SPI read in 3-wire mode

3-wire mode is entered by setting the bit SIM (SPI serial interface mode selection) to '1' in *CTRL\_REG2\_XM (21h)* for the accelerometer and magnetic sensor and in *CTRL\_REG4\_G (23h)* for the gyroscope.





The SPI Read command is performed with 16 clock pulses:

*bit 0*: READ bit. The value is 1.

**bit 1**:  $M\overline{S}$  bit. When 0, does not increment address; when 1, increments address in multiple reads.

bit 2-7: address AD(5:0). This is the address field of the indexed register.

*bit 8-15*: data DO(7:0) (read mode). This is the data that will be read from the device (MSb first).

The multiple read command is also available in 3-wire mode.



# 7 Register mapping

The table given below provides a listing of the 8-bit registers embedded in the device and their respective addresses.

News	Slave	-	Register	address	Default	
Name	Address	Туре	Hex	Binary	Default	
Reserved	Table 16		00-0E			
WHO_AM_I_G	Table 16	r	0F	000 1111	11010100	
Reserved	Table 16		10-1F			
CTRL_REG1_G	Table 16	rw	20	010 0000	00000111	
CTRL_REG2_G	Table 16	rw	21	010 0001	00000000	
CTRL_REG3_G	Table 16	rw	22	010 0010	00000000	
CTRL_REG4_G	Table 16	rw	23	010 0011	00000000	
CTRL_REG5_G	Table 16	rw	24	010 0100	00000000	
REFERENCE_G	Table 16	rw	25	010 0101	00000000	
Reserved	Table 16		26			
STATUS_REG_G	Table 16	r	27	010 0111	output	
OUT_X_L_G	Table 16	r	28	010 1000	output	
OUT_X_H_G	Table 16	r	29	010 1001	output	
OUT_Y_L_G	Table 16	r	2A	010 1010	output	
OUT_Y_H_G	Table 16	r	2B	010 1011	output	
OUT_Z_L_G	Table 16	r	2C	010 1100	output	
OUT_Z_H_G	Table 16	r	2D	010 1101	output	
FIFO_CTRL_REG_G	Table 16	rw	2E	010 1110	00000000	
FIFO_SRC_REG_G	Table 16	r	2F	010 1111	output	
INT1_CFG_G	Table 16	rw	30	011 0000	00000000	
INT1_SRC_G	Table 16	r	31	011 0001	output	
INT1_TSH_XH_G	Table 16	rw	32	011 0010	00000000	
INT1_TSH_XL_G	Table 16	rw	33	011 0011	00000000	
INT1_TSH_YH_G	Table 16	rw	34	011 0100	00000000	
INT1_TSH_YL_G	Table 16	rw	35	011 0101	00000000	
INT1_TSH_ZH_G	Table 16	rw	36	011 0110	00000000	
INT1_TSH_ZL_G	Table 16	rw	37	011 0111	00000000	
INT1_DURATION_G	Table 16	rw	38	011 1000	00000000	
Reserved	Table 15		00-04			

Table '	17.	Register	address	man
Table		Register	uuui 033	map

38/74



Table 17: Register address map (continued)							
Namo	Slave	Tuno	Register	address	Default		
Name	Address	Type	Hex	Binary	Delault		
OUT_TEMP_L_XM	Table 15	r	05	000 0101	output		
OUT_TEMP_H_XM	Table 15	r	06	000 0110	output		
STATUS_REG_M	Table 15	r	07	000 0111	output		
OUT_X_L_M	Table 15	r	08	000 1000	output		
OUT_X_H_M	Table 15	r	09	000 1001	output		
OUT_Y_L_M	Table 15	r	0A	000 1010	output		
OUT_Y_H_M	Table 15	r	0B	000 1011	output		
OUT_Z_L_M	Table 15	r	0C	000 1100	output		
OUT_Z_H_M	Table 15	r	0D	000 1101	output		
Reserved	Table 15		0E	000 1110			
WHO_AM_I_XM	Table 15	r	0F	000 1111	01001001		
Reserved	Table 15		10-11				
INT_CTRL_REG_M	Table 15	rw	12	001 0010	11101000		
INT_SRC_REG_M	Table 15	r	13	001 0011	output		
INT_THS_L_M	Table 15	rw	14	001 0100	00000000		
INT_THS_H_M	Table 15	rw	15	001 0101	00000000		
OFFSET_X_L_M	Table 15	rw	16	001 0110	00000000		
OFFSET_X_H_M	Table 15	rw	17	001 0111	00000000		
OFFSET_Y_L_M	Table 15	rw	18	001 01000	00000000		
OFFSET_Y_H_M	Table 15	rw	19	001 01001	00000000		
OFFSET_Z_L_M	Table 15	rw	1A	001 01010	00000000		
OFFSET_Z_H_M	Table 15	rw	1B	001 01011	00000000		
REFERENCE_X	Table 15	rw	1C	001 01100	00000000		
REFERENCE_Y	Table 15	rw	1D	001 01101	00000000		
REFERENCE_Z	Table 15	rw	1E	001 01110	00000000		
CTRL_REG0_XM	Table 15	rw	1F	001 1111	00000000		
CTRL_REG1_XM	Table 15	rw	20	010 0000	00000111		
CTRL_REG2_XM	Table 15	rw	21	010 0001	00000000		
CTRL_REG3_XM	Table 15	rw	22	010 0010	00000000		
CTRL_REG4_XM	Table 15	rw	23	010 0011	00000000		
CTRL_REG5_XM	Table 15	rw	24	010 0100	00011000		
CTRL_REG6_XM	Table 15	rw	25	010 0101	00100000		
CTRL_REG7_XM	Table 15	rw	26	010 0110	00000001		

Table 17. Register address map (continued)



News	Slave	Turne	Register	Register address		
Name	Address	туре	Hex	Binary	Default	
STATUS_REG_A	Table 15	r	27	010 0111	output	
OUT_X_L_A	Table 15	r	28	010 1000	output	
OUT_X_H_A	Table 15	r	29	010 1001	output	
OUT_Y_L_A	Table 15	r	2A	010 1010	output	
OUT_Y_H_A	Table 15	r	2B	010 1011	output	
OUT_Z_L_A	Table 15	r	2C	010 1100	output	
OUT_Z_H_A	Table 15	r	2D	010 1101	output	
FIFO_CTRL_REG	Table 15	rw	2E	010 1110	00000000	
FIFO_SRC_REG	Table 15	r	2F	010 1111	output	
INT_GEN_1_REG	Table 15	rw	30	011 0000	00000000	
INT_GEN_1_SRC	Table 15	r	31	011 0001	output	
INT_GEN_1_THS	Table 15	rw	32	011 0010	00000000	
INT_GEN_1_DURATION	Table 15	rw	33	011 0011	00000000	
INT_GEN_2_REG	Table 15	rw	34	011 0100	00000000	
INT_GEN_2_SRC	Table 15	r	35	011 0101	output	
INT_GEN_2_THS	Table 15	rw	36	011 0110	00000000	
INT_GEN_2_DURATION	Table 15	rw	37	011 0111	00000000	
CLICK_CFG	Table 15	rw	38	011 1000	00000000	
CLICK_SRC	Table 15	r	39	011 1001	output	
CLICK_THS	Table 15	rw	3A	011 1010	00000000	
TIME_LIMIT	Table 15	rw	3B	011 1011	00000000	
TIME _LATENCY	Table 15	rw	3C	011 1100	00000000	
TIME_WINDOW	Table 15	rw	3D	011 1101	00000000	
Act_THS	Table 15	rw	3E	011 1110	00000000	
Act_DUR	Table 15	rw	3F	011 1111	00000000	

Table 17. Register address map (continued)

Registers marked as *Reserved* must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory-calibrated values. Their content is automatically restored when the device is powered up.



# 8 Register description

The device contains a set of registers which are used to control its behavior and to retrieve angular rate data. The register address, consisting of 7 bits, is used to identify them and to write the data through the serial interface.

# 8.1 WHO\_AM\_I\_G (0Fh)

Table 18. WHO_AM_I_G register							
1	1	0	1	0	1	0	0

Device identification register.

# 8.2 CTRL\_REG1\_G (20h)

#### Table 19. CTRL\_REG1\_G register

DR1	DR0	BW1	BW0	PD	Zen	Xen	Yen

#### Table 20. CTRL\_REG1\_G description

DR1-DR0	Output data rate selection. Refer to Table 21			
BW1-BW0	Bandwidth selection. Refer to Table 21			
PD	Power-down mode enable. Default value: 0 (0: power-down mode, 1: normal mode or sleep mode)			
Zen	Z-axis enable. Default value: 1 (0: Z-axis disabled; 1: Z-axis enabled)			
Yen	Y-axis enable. Default value: 1 (0: Y-axis disabled; 1: Y-axis enabled)			
Xen	X-axis enable. Default value: 1 (0: X-axis disabled; 1: X-axis enabled)			

 $\ensuremath{\text{DR[1:0]}}$  is used for ODR selection.  $\ensuremath{\text{BW}}$  [1:0] is used for Bandwidth selection.

In Table 21 all frequencies resulting in combinations of DR / BW bits are given.

#### Table 21. DR and BW configuration setting

DR [1:0]	BW [1:0]	ODR (Hz)	Cutoff
00	00	95	12.5
00	01	95	25
00	10	95	25



DocID024763 Rev 2

41/74

Table 21. DR and BW configuration setting (continued)					
DR [1:0]	BW [1:0]	ODR (Hz)	Cutoff		
00	11	95	25		
01	00	190	12.5		
01	01	190	25		
01	10	190	50		
01	11	190	70		
10	00	380	20		
10	01	380	25		
10	10	380	50		
10	11	380	100		
11	00	760	30		
11	01	760	35		
11	10	760	50		
11	11	760	100		

A combination of PD, Zen, Yen, Xen is used to set device to different modes (power-down / normal / sleep mode) in accordance with Table 22 below.

Table 22. Power mode selection configuration

Mode	PD	Zen	Yen	Xen
Power-down	0	-	-	-
Sleep	1	0	0	0
Normal	1	-	-	-

#### 8.3 CTRL\_REG2\_G (21h)

### Table 23. CTRL\_REG2\_G register

0 <sup>(1)</sup>	0 <sup>(1)</sup>	HPM1	HPM1	HPCF3	HPCF2	HPCF1	HPCF0

1. These bits must be set to '0' to ensure proper operation of the device

### Table 24. CTRL\_REG2\_G description

HPM1-	High-pass filter mode selection. Default value: 00
HPM0	Refer to <i>Table 25</i>
HPCF3-	High-pass filter cutoff frequency selection
HPCF0	Refer to <i>Table 26</i>

DocID024763 Rev 2



LSM9DS0

HPM1	HPM0	High-pass filter mode				
0	0	Normal mode (reset reading HP_RESET_FILTER)				
0	1	Reference signal for filtering				
1	0	Normal mode				
1	1	Autoreset on interrupt event				

Table 25. High-pass filter mode configuration

### Table 26. High-pass filter cutoff frequency configuration (Hz)

HPCF[3:0]	ODR = 95 Hz	ODR = 190 Hz	ODR = 380 Hz	ODR = 760 Hz
0000	7.2	13.5	27	51.4
0001	3.5	7.2	13.5	27
0010	1.8	3.5	7.2	13.5
0011	0.9	1.8	3.5	7.2
0100	0.45	0.9	1.8	3.5
0101	0.18	0.45	0.9	1.8
0110	0.09	0.18	0.45	0.9
0111	0.045	0.09	0.18	0.45
1000	0.018	0.045	0.09	0.18
1001	0.009	0.018	0.045	0.09

# 8.4 CTRL\_REG3\_G (22h)

ſ

		Table	27. CTRL_	REG3_G re	gister		
I1_Int1	I1_Boot	H_Lactive	PP_OD	I2_DRDY	I2_WTM	I2_ORun	I2_Empty

### Table 28. CTRL\_REG3\_G description

I1_Int1	Interrupt enable on INT_G pin. Default value 0. (0: disable; 1: enable)
I1_Boot	Boot status available on INT_G. Default value 0. (0: disable; 1: enable)
H_Lactive	Interrupt active configuration on INT_G. Default value 0. (0: high; 1:low)
PP_OD	Push-pull / Open drain. Default value: 0. (0: push- pull; 1: open drain)
I2_DRDY	Date-ready on DRDY_G. Default value 0. (0: disable; 1: enable)
I2_WTM	FIFO watermark interrupt on DRDY_G. Default value: 0. (0: disable; 1: enable)
I2_ORun	FIFO overrun interrupt on DRDY_G. Default value: 0. (0: disable; 1: enable)
I2_Empty	FIFO empty interrupt on DRDY_G. Default value: 0. (0: disable; 1: enable)



# 8.5 CTRL\_REG4\_G (23h)

Table 29. CTRL_REG4_G register							
BDU	BLE	FS1	FS0	-	ST1	ST0	SIM

...

### Table 30. CTRL\_REG4\_G description

BDU	Block data update. Default value: 0 (0: continuous update; 1: output registers not updated until MSb and LSb read)
BLE	Big/little endian data selection. Default value 0. (0: Data LSb @ lower address; 1: Data MSb @ lower address)
FS1-FS0	Full-scale selection. Default value: 00 (00: 245 dps; 01: 500 dps; 10: 2000 dps; 11: 2000 dps)
ST1-ST0	Self-test enable. Default value: 00 (00: Self-test disabled; Other: See <i>Table 31</i> )
SIM	SPI serial interface mode selection. Default value: 0 (0: 4-wire interface; 1: 3-wire interface).

#### Table 31. Self-test mode configuration

ST1	ST0	Self-test mode
0	0	Normal mode
0	1	Self-test 0 $^{(1)}$ (X positive sign, Y and Z negative sign)
1	0	
1	1	Self-test 1 <sup>(1)</sup> (X negative sign, Y and Z positive sign)

1. DST sign (absolute value in *Table 3*)

# 8.6 CTRL\_REG5\_G (24h)

#### Table 32. CTRL\_REG5\_G register

BOOT FIFO_EN	HPen	INT1_Sel1	INT1_Sel0	Out_Sel1	Out_Sel0
--------------	------	-----------	-----------	----------	----------

### Table 33. CTRL\_REG5\_G description

BOOT	Reboot memory content. Default value: 0 (0: normal mode; 1: reboot memory content)
FIFO_EN	FIFO enable. Default value: 0 (0: FIFO disable; 1: FIFO enable)
HPen	High-pass filter enable. Default value: 0 (0: HPF disabled; 1: HPF enabled) (See <i>Figure 18</i> )
INT1_Sel1-INT1_Sel0	INT1 selection configuration. Default value: 00 (See <i>Figure 18</i> )
Out_Sel1-Out_Sel0	Out selection configuration. Default value: 00 (See <i>Figure 18</i> )





## 8.7 REFERENCE/DATACAPTURE\_G (25h)

Table 34. REFERENCE/DATACAPTURE_G register							
Ref7	Ref6	Ref5	Ref4	Ref3	Ref2	Ref1	Ref0

#### Table 35. REFERENCE/DATACAPTURE\_G description

	<b>–</b> •
Ref 7-Ref0	Reference value for interrupt generation. Default value: 0

# 8.8 STATUS\_REG\_G (27h)

### Table 36. STATUS\_REG\_G register

ZYXOR	ZOR	YOR	XOR	ZYXDA	ZDA	YDA	XDA
-------	-----	-----	-----	-------	-----	-----	-----

### Table 37. STATUS\_REG\_G description

ZYXOR	X, Y, Z -axis data overrun. Default value: 0 (0: no overrun has occurred; 1: new data has overwritten the previous data before it was read)
ZOR	Z-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: new data for the Z-axis has overwritten the previous data)
YOR	Y-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: new data for the Y-axis has overwritten the previous data)
XOR	X-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: new data for the X-axis has overwritten the previous data)



	Table 37. STATUS_REG_G description (continued)
ZYXDA	X, Y, Z -axis new data available. Default value: 0 (0: a new set of data is not yet available; 1: a new set of data is available)
ZDA	Z-axis new data available. Default value: 0 (0: new data for the Z-axis is not yet available; 1: new data for the Z-axis is available)
YDA	Y-axis new data available. Default value: 0 (0: new data for the Y-axis is not yet available;1: new data for the Y-axis is available)
XDA	X-axis new data available. Default value: 0 (0: new data for the X-axis is not yet available; 1: new data for the X-axis is available)

#### Table 27 STATUS DEC C description (continued)

#### OUT\_X\_L\_G (28h), OUT\_X\_H\_G (29h) 8.9

X-axis angular rate data. The value is expressed as two's complement.

#### OUT\_Y\_L\_G (2Ah), OUT\_Y\_H\_G (2Bh) 8.10

Y-axis angular rate data. The value is expressed as two's complement.

#### 8.11 OUT\_Z\_L\_G (2Ch), OUT\_Z\_H\_G (2Dh)

Z-axis angular rate data. The value is expressed as two's complement.

#### FIFO\_CTRL\_REG\_G (2Eh) 8.12

### Table 38. FIFO\_CTRL\_REG G register

					-		
FM2	FM1	FM0	WTM4	WTM3	WTM2	WTM1	WTM0

#### Table 39. FIFO\_CTRL\_REG\_G description

FM2-FM0	FIFO mode selection. Default value: 00 (see Table 40)
WTM4-WTM0	FIFO threshold. Watermark level setting

### Table 40. FIFO mode configuration

FM2	FM1	FM0	FIFO mode
0	0	0	Bypass mode
0	0	1	FIFO mode
0	1	0	Stream mode
0	1	1	Stream-to-FIFO mode
1	0	0	Bypass-to-Stream mode



# 8.13 FIFO\_SRC\_REG\_G (2Fh)

### Table 41. FIFO\_SRC\_REG\_G register

					-		
WTM	OVRN	EMPTY	FSS4	FSS3	FSS2	FSS1	FSS0

### Table 42. FIFO\_SRC\_REG\_G description

WTM	Watermark status. (0: FIFO filling is lower than WTM level; 1: FIFO filling is equal to or higher than WTM level)
OVRN	Overrun bit status. (0: FIFO is not completely filled; 1: FIFO is completely filled)
EMPTY	FIFO empty bit. (0: FIFO not empty; 1: FIFO empty)
FSS4-FSS1	FIFO stored data level

# 8.14 INT1\_CFG\_G (30h)

### Table 43. INT1\_CFG\_G register

AND/OR	LIR	ZHIE	ZLIE	YHIE	YLIE	XHIE	XLIE

### Table 44. INT1\_CFG\_G description

AND/OR	AND/OR combination of interrupt events. Default value: 0 (0: OR combination of interrupt events 1: AND combination of interrupt events
LIR	Latch interrupt request. Default value: 0 (0: interrupt request not latched; 1: interrupt request latched) Cleared by reading INT1_SRC reg.
ZHIE	Enable interrupt generation on Z high event. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured value higher than preset threshold)
ZLIE	Enable interrupt generation on Z low event. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured value lower than preset threshold)
YHIE	Enable interrupt generation on Y high event. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured value higher than preset threshold)
YLIE	Enable interrupt generation on Y low event. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured value lower than preset threshold)



	Table 44. IN TICEG description (continued)					
	Enable interrupt generation on X high event. Default value: 0					
XHIE	(0: disable interrupt request; 1: enable interrupt request on measured value higher than preset threshold)					
XLIE	Enable interrupt generation on X low event. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured value lower than preset threshold)					

### Table 44. INT1\_CFG\_G description (continued)

# 8.15 INT1\_SRC\_G (31h)

Interrupt source register. Read-only register.

		Table	9 45. INT1_	SRC_G reg	jister		
0	IA	ZH	ZL	YH	YL	ХН	XL

### Table 46. INT1\_SRC\_G description

IA	Interrupt active. Default value: 0 (0: no interrupt has been generated; 1: one or more interrupts have been generated)
ZH	Z high. Default value: 0 (0: no interrupt, 1: Z high event has occurred)
ZL	Z low. Default value: 0 (0: no interrupt; 1: Z low event has occurred)
YH	Y high. Default value: 0 (0: no interrupt, 1: Y high event has occurred)
YL	Y low. Default value: 0 (0: no interrupt, 1: Y low event has occurred)
хн	X high. Default value: 0 (0: no interrupt, 1: X high event has occurred)
XL	X low. Default value: 0 (0: no interrupt, 1: X low event has occurred)

Reading at this address clears the INT1\_SRC IA bit (and eventually the interrupt signal on the INT\_G pin) and allows the refresh of data in the INT1\_SRC register if the latched option was chosen.

# 8.16 INT1\_THS\_XH\_G (32h)

#### Table 47. INT1\_THS\_XH\_G register

					-		
-	THSX14	THSX13	THSX12	THSX11	THSX10	THSX9	THSX8

### Table 48. INT1\_THS\_XH\_G description

THSX14 - THSX8	Interrupt threshold. Default value: 000 0000
----------------	--



## 8.17 INT1\_THS\_XL\_G (33h)

Table 49. INT1_THS_XL_G register							
THSX7	THSX6	THSX5	THSX4	THSX3	THSX2	THSX1	THSX0

#### Table 50. INT1\_THS\_XL\_G description

THSX7 - THSX0	Interrupt threshold. Default value: 0000 0000

# 8.18 INT1\_THS\_YH\_G (34h)

### Table 51. INT1\_THS\_YH\_G register

	-	THSY14	THSY13	THSY12	THSY11	THSY10	THSY9	THSY8
--	---	--------	--------	--------	--------	--------	-------	-------

### Table 52. INT1\_THS\_YH\_G description

THSY14 - THSY8	Interrupt threshold. Default value: 000 0000

# 8.19 INT1\_THS\_YL\_G (35h)

### Table 53. INT1\_THS\_YL\_G register

THSR7 THSY6	THSY5	THSY4	THSY3	THSY2	THSY1	THSY0
-------------	-------	-------	-------	-------	-------	-------

#### Table 54. INT1\_THS\_YL\_G description

THSY7 - THSY0	Interrupt threshold. Default value: 0000 0000
---------------	---

# 8.20 INT1\_THS\_ZH\_G (36h)

#### Table 55. INT1\_THS\_ZH\_G register

-	THSZ14	THSZ13	THSZ12	THSZ11	THSZ10	THSZ9	THSZ8
---	--------	--------	--------	--------	--------	-------	-------

### Table 56. INT1\_THS\_ZH\_G description

alue: 000 0000	Interrupt threshold. Default va	THSZ14 - THSZ8	
----------------	---------------------------------	----------------	--



# 8.21 INT1\_THS\_ZL\_G (37h)

	Table 57. INT1_THS_ZL_G register						
THSZ7	THSZ6	THSZ5	THSZ4	THSZ3	THSZ2	THSZ1	THSZ0

#### Table 58. INT1 THS ZL G description

THSZ7 - THSZ0	Interrupt threshold. Default value: 0000 0000

# 8.22 INT1\_DURATION\_G (38h)

#### Table 59. INT1\_DURATION\_G register

					-		
WAIT	D6	D5	D4	D3	D2	D1	D0
				-		-	

#### Table 60. INT1\_DURATION\_G description

WAIT	WAIT enable. Default value: 0 (0: disable; 1: enable)
D6 - D0	Duration value. Default value: 000 0000

The **D6** - **D0** bits set the minimum duration of the interrupt event to be recognized. Duration steps and maximum values depend on the ODR chosen.

The **WAIT** bit has the following definitions:

Wait = '0': the interrupt falls immediately if the signal crosses the selected threshold

Wait = '1': if the signal crosses the selected threshold, the interrupt falls only after the duration has counted the number of samples at the selected data rate, written into the duration counter register.





Figure 20. Wait bit enabled





## 8.23 OUT\_TEMP\_L\_XM (05h), OUT\_TEMP\_H\_XM (06h)

Temperature sensor data.

Refer to Section 4.4: Temperature sensor for details on how to enable and read the temperature sensor output data.

# 8.24 STATUS\_REG\_M (07h)

Table 61. STATUS REG IN registe	able 61	. STATUS	REG	Μ	registe
---------------------------------	---------	----------	-----	---	---------

ZYXMOR	ZMOR	YMOR	XMOR	ZYXMDA	ZMDA	YMDA	XMDA

Table 62. STATUS_	_REG_M description
-------------------	--------------------

ZYXMOR	Magnetic X, Y and Z-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new set of data has overwritten the previous ones).
ZMOR	Z-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new data for the Z-axis has overwritten the previous one)
YMOR	Y-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new data for the Y-axis has overwritten the previous one)
XMOR	X-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new data for the X-axis has overwritten the previous one)
ZYXMDA	X, Y and Z-axis new data available. Default value: 0 (0: a new set of data is not yet available; 1: a new set of data is available).
ZMDA	Z-axis new data available. Default value: 0 (0: a new set of data for the Z-axis is not yet available; 1: a new set of data for the Z-axis is available)
YMDA	Y-axis new data available. Default value: 0 (0: a new set of data for the Y-axis is not yet available; 1: a new set of data for the Y-axis is available)
XMDA	X-axis new data available. Default value: 0 (0: a new set of data for the X-axis is not yet available; 1: a new set of data for the X-axis is available)

# 8.25 OUT\_X\_L\_M (08h), OUT\_X\_H\_M (09h)

X-axis magnetic data.

The value is expressed in 16-bit as two's complement left justified.

# 8.26 OUT\_Y\_L\_M (0Ah), OUT\_Y\_H\_M (0Bh)

Y-axis magnetic data.

The value is expressed in 16-bit as two's complement left justified.



# 8.27 OUT\_Z\_L\_M (0Ch), OUT\_Z\_H\_M (0Dh)

Z-axis magnetic data.

The value is expressed in 16-bit as two's complement left justified.

# 8.28 WHO\_AM\_I\_XM (0Fh)

		Table 63	. WHO_AN	1_I_XM reg	jister		
0	1	0	0	1	0	0	1

Device identification register.

# 8.29 INT\_CTRL\_REG\_M (12h)

Table 64. INT_CTRL_REG_M register							
XMIEN	YMIEN	ZMIEN	PP_OD	IEA	IEL	4D	MIEN

#### Table 65. INT\_CTRL\_REG\_M description

XMIEN	Enable interrupt recognition on X-axis for magnetic data. Default value: 0. (0: disable interrupt recognition;1: enable interrupt recognition)
YMIEN	Enable interrupt recognition on Y-axis for magnetic data. Default value: 0. (0: disable interrupt recognition;1: enable interrupt recognition)
ZMIEN	Enable interrupt recognition on Z-axis for magnetic data. Default value: 0. (0: disable interrupt recognition;1: enable interrupt recognition)
PP_OD	Interrupt pin configuration. Default value: 0. (0: push-pull; 1: open drain)
IEA	Interrupt polarity for both accelerometer and magnetometer. Default value: 0. (0: interrupt active-low; 1: interrupt active-high)
IEL	Latch interrupt request on accelerometer <i>INT_GEN_1_SRC (31h)</i> and <i>INT_GEN_2_SRC (35h)</i> registers, and magnetometer <i>INT_SRC_REG_M (13h)</i> register. Default value: 0. (0: interrupt request not latched; 1: interrupt request latched)
	Once the IEL is set to '1' the interrupt is cleared by reading for the accelerometer the <i>INT_GEN_1_SRC (31h)</i> and <i>INT_GEN_2_SRC (35h)</i> registers, and for the magnetometer the <i>INT_SRC_REG_M (13h)</i> register.
4D	4D enable: 4D detection on acceleration data is enabled when 6D bit in <i>INT_GEN_1_REG (30h)</i> is set to 1.
MIEN	Enable interrupt generation for magnetic data. Default value: 0. (0: disable interrupt generation;1: enable interrupt generation)



# 8.30 INT\_SRC\_REG\_M (13h)

#### Table 66. INT\_SRC\_REG\_M register

M_PTH_X	M_PTH_Y	M_PTH_Z	M_NTH_X	M_NTH_Y	M_NTH_Z	MROI	MINT

#### Table 67. INT\_SRC\_REG\_M description

M_PTH_X	Magnetic value on X-axis exceeds the threshold on the positive side. Default value: 0.
M_PTH_Y	Magnetic value on Y-axis exceeds the threshold on the positive side. Default value: 0.
M_PTH_Z	Magnetic value on Z-axis exceeds the threshold on the positive side. Default value: 0.
M_NTH_X	Magnetic value on X-axis exceeds the threshold on the negative side. Default value: 0.
M_NTH_Y	Magnetic value on Y-axis exceeds the threshold on the negative side. Default value: 0.
M_NTH_Z	Magnetic value on Z-axis exceeds the threshold on the negative side. Default value: 0.
MROI	Internal measurement range overflow on magnetic value. Default value: 0. To enable this feature need to set to 1 MIEN bit in <i>8.29: INT_CTRL_REG_M (12h)</i>
MINT	Magnetic interrupt event. The magnetic field value exceeds the threshold. Default value: 0.

## 8.31 INT\_THS\_L\_M (14h), INT\_THS\_H\_M (15h)

Magnetic interrupt threshold. Default value: 0.

The value is expressed in 16-bit unsigned.

Even if the threshold is expressed in absolute value, the device detects both positive and negative thresholds.

# 8.32 OFFSET\_X\_L\_M (16h), OFFSET\_X\_H\_M (17h)

Magnetic offset for X-axis. Default value: 0.

The value is expressed in 16-bit as two's complement left justified.

# 8.33 OFFSET\_Y\_L\_M (18h), OFFSET\_Y\_H\_M (19h)

Magnetic offset for Z-axis. Default value: 0. The value is expressed in 16-bit as two's complement left justified.

## 8.34 OFFSET\_Z\_L\_M (1Ah), OFFSET\_Z\_H\_M (1Bh)

Magnetic offset for Y-axis. Default value: 0. The value is expressed in 16-bit as two's complement left justified.


## 8.35 REFERENCE\_X (1Ch)

Reference value for high-pass filter for x-axis acceleration data.

## 8.36 REFERENCE\_Y (1Dh)

Reference value for high-pass filter for y-axis acceleration data.

## 8.37 REFERENCE\_Z (1Eh)

Reference value for high-pass filter for z-axis acceleration data.

## 8.38 CTRL\_REG0\_XM (1Fh)

#### Table 68. CTRL\_REG0\_XM register

BOOT	FIFO_EN	WTM_EN	0 <sup>(1)</sup>	0 <sup>(1)</sup>	HP_Click	HPIS1	HPIS2

1. These bits must be set to '0' for the correct operation of the device

#### Table 69. CTRL\_REG0\_XM description

BOOT	Reboot memory content. Default value: 0 (0: normal mode; 1: reboot memory content)
FIFO_EN	FIFO enable. Default value: 0 (0: FIFO disable; 1: FIFO Enable)
WTM_EN	FIFO programmable watermark enable. Default value: 0 (0: disable; 1: Enable)
HP_Click	High-pass filter enabled for Click function. Default value: 0 (0: filter bypassed; 1: filter enabled)
HPIS1	High-pass filter enabled for interrupt generator 1. Default value: 0 (0: filter bypassed; 1: filter enabled)
HPIS2	High-pass filter enabled for interrupt generator 2. Default value: 0 (0: filter bypassed; 1: filter enabled)

## 8.39 CTRL\_REG1\_XM (20h)

#### Table 70. CTRL\_REG1\_XM register

AODR3	AODR2	AODR1	AODR0	BDU	AZEN	AYEN	AXEN
-------	-------	-------	-------	-----	------	------	------



AODR3- AODR0	Acceleration data rate selection. Default value: 0000 (0000: power-down mode; others: refer to <i>Table 72: Acceleration data rate configura-</i> <i>tion</i> )
BDU	Block data update for acceleration and magnetic data. Default value: 0 (0: continuous update; 1: output registers not updated until MSB and LSB have been read)
AZEN	Acceleration Z-axis enable. Default value: 1 (0: Z-axis disabled; 1: Z-axis enabled)
AYEN	Acceleration Y-axis enable. Default value: 1 (0: Y-axis disabled; 1: Y-axis enabled)
AXEN	Acceleration X-axis enable. Default value: 1 (0: X-axis disabled; 1: X-axis enabled)

Table 71. CTRL\_REG1\_XM description

AODR[3:0] is used to set the power mode and ODR selection. The following table indicates all frequencies resulting from the combination of AODR[3:0].

AODR3	AODR2	AODR1	AODR0	Power mode selection
0	0	0	0	Power-down mode
0	0	0	1	3.125 Hz
0	0	1	0	6.25 Hz
0	0	1	1	12.5 Hz
0	1	0	0	25 Hz
0	1	0	1	50 Hz
0	1	1	0	100 Hz
0	1	1	1	200 Hz
1	0	0	0	400 Hz
1	0	0	1	800 Hz
1	0	1	0	1600 Hz

Table 72. Acceleration data rate configuration

## 8.40 CTRL\_REG2\_XM (21h)

Table 73. CTRL_REG2_XM register								
ABW1	ABW0	AFS2	AFS1	AFS0	AST1	AST0	SIM	



ABW[1:0]	Accelerometer anti-alias filter bandwidth. Default value: 00
	Refer to Table 75: Acceleration anti-alias filter bandwidth
AFS[2:0]	Acceleration full-scale selection. Default value: 000
	Refer to Table 76: Acceleration full-scale selection
AST[1:0]	Acceleration self-test enable. Default value: 00
	(00: self-test disabled; see Table 77: Self-test mode configuration)
SIM	SPI Serial Interface Mode selection. Default value: 0
	(0: 4-wire interface; 1: 3-wire interface)

#### Table 74, CTRL REG2 XM description

#### Table 75. Acceleration anti-alias filter bandwidth

ABW1	ABW0	Anti-alias filter bandwidth			
0	0	773 Hz			
0	1	194 Hz			
1	0	362 Hz			
1	1	50 Hz			

#### Table 76. Acceleration full-scale selection

AFS2	AFS1	AFS0	Acceleration full scale
0	0	0	±2 g
0	0	1	±4 g
0	1	0	±6 g
0	1	1	±8 g
1	0	0	±16 g

#### Table 77. Self-test mode configuration

AST1	AST0	Self-test mode
0	0	Normal mode
0	1	Positive sign self-test
1	0	Negative sign self-test
1	1	Not allowed

## 8.41 CTRL\_REG3\_XM (22h)

# Table 78. CTRL\_REG3\_XM register P1\_BOOT P1\_TAP P1\_INT1 P1\_INT2 P1\_INTM P1\_DRDYA P1\_DRDYM P1\_EMPTY



DocID024763 Rev 2

57/74

P1_BOOT	Boot on INT1_XM pin enable. Default value: 0 (0: disable; 1: enable)
P1_TAP	Tap generator interrupt on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_INT1	Inertial interrupt generator 1 on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_INT2	Inertial interrupt generator 2 on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_INTM	Magnetic interrupt generator on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_DRDYA	Accelerometer data-ready signal on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_DRDYM	Magnetometer data-ready signal on INT1_XM pin. Default value: 0 (0: disable; 1: enable)
P1_EMPTY	FIFO empty indication on INT1_XM pin. Default value: 0 (0: disable; 1: enable)

Table 79. CTRL\_REG3\_XM description

## 8.42 CTRL\_REG4\_XM (23h)

#### Table 80. CTRL\_REG4\_XM register

P2_TAP	P2_INT1	P2_INT2	P2_INTM	P2_DRDYA	P2_DRDYM	P2_Overrun	P2_WTM
--------	---------	---------	---------	----------	----------	------------	--------

#### Table 81. CTRL\_REG4\_XM description

P2_TAP	Tap generator interrupt on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_INT1	Inertial interrupt generator 1 on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_INT2	Inertial interrupt generator 2 on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_INTM	Magnetic interrupt generator on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_DRDYA	Accelerometer data-ready signal on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_DRDYM	Magnetometer data-ready signal on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_Overrun	FIFO overrun interrupt on INT2_XM pin. Default value: 0 (0: disable; 1: enable)
P2_WTM	FIFO watermark interrupt on INT2_XM pin. Default value: 0 (0: disable; 1: enable)



## 8.43 CTRL\_REG5\_XM (24h)

#### Table 82. CTRL\_REG5\_XM register

					-		
TEMP_EN	M_RES1	M_RES0	M_ODR2	M_ODR1	M_ODR0	LIR2	LIR1

#### Table 83. CTRL\_REG5\_XM description

TEMP_EN	Temperature sensor enable. Default value: 0 (0: temperature sensor disabled; 1: temperature sensor enabled)
M_RES[1:0]	Magnetic resolution selection. Default value: 00 (00: low resolution, 11: high resolution)
M_ODR[2:0]	Magnetic data rate selection. Default value: 110 Refer to <i>Table 84: Magnetic data rate configuration</i>
LIR2	Latch interrupt request on INT2_SRC register, with INT2_SRC register cleared by reading INT2_SRC itself. Default value: 0. (0: interrupt request not latched; 1: interrupt request latched)
LIR1	Latch interrupt request on INT1_SRC register, with INT1_SRC register cleared by reading INT1_SRC itself. Default value: 0. (0: interrupt request not latched; 1: interrupt request latched)

#### Table 84. Magnetic data rate configuration

M_ODR2	M_ODR1	M_ODR0	Power mode selection
0	0	0	3.125 Hz
0	0	1	6.25 Hz
0	1	0	12.5 Hz
0	1	1	25 Hz
1	0	0	50 Hz
1	0	1	100 Hz <sup>(1)</sup>
1	1	0	Reserved
1	1	1	Reserved

 Available only for accelerometer ODR > 50 Hz or accelerometer in power-down mode (refer to Table 72, AODR setting).

## 8.44 CTRL\_REG6\_XM (25h)

#### Table 85. CTRL\_REG6\_XM register

$0^{(1)}$ MFS1 MFS0 $0^{(1)}$ $0^{(1)}$ $0^{(1)}$ $0^{(1)}$ $0^{(1)}$
---

1. These bits must be set to '0' for the correct operation of the device



#### Table 86. CTRL\_REG6\_XM description

MFS1-	Magnetic full-scale selection. Default value: 01
MFS0	Refer to Table 87: Magnetic full-scale selection

#### Table 87. Magnetic full-scale selection

MFS1	MFS0	Magnetic full scale
0	0	± 2 gauss
0	1	± 4 gauss
1	0	± 8 gauss
1	1	± 12 gauss

## 8.45 CTRL\_REG7\_XM (26h)

## Table 88. CTRL\_REG7\_XM register

|--|

1. These bits must be set to '0' for the correct operation of the device

#### Table 89. CTRL\_REG7\_XM description

AHPM1- AHPM0	High-pass filter mode selection for acceleration data. Default value: 00 Refer to <i>Table 90: High-pass filter mode selection</i>
AFDS	Filtered acceleration data selection. Default value: 0 (0: internal filter bypassed; 1: data from internal filter sent to output register and FIFO)
MLP	Magnetic data low-power mode. Default value: 0 If this bit is '1' the MODR is set to 3.125 Hz independently from the MODR settings. Once the bit is set to '0' the magnetic data rate is configured by MODR bits in <i>CTRL_REG5_XM (24h)</i> register.
MD1-MD0	Magnetic sensor mode selection. Default 10 Refer to <i>Table 91: Magnetic sensor mode selection</i>

#### Table 90. High-pass filter mode selection

AHPM1	AHPM0	High-pass filter mode
0	0	Normal mode (resets x, y and z-axis reading <i>REFERENCE_X (1Ch)</i> , <i>REFERENCE_Y (1Dh)</i> and <i>REFERENCE_Y (1Dh)</i> registers respectively)
0	1	Reference signal for filtering
1	0	Normal mode
1	1	Autoreset on interrupt event



LSM9DS0

MD1-0	MD1-0	Magnetic sensor mode			
0	0	Continuous-conversion mode			
0	1	Single-conversion mode			
1	0	Power-down mode			
1	1	Power-down mode			

Table 91. Magnetic sensor mode selection

## 8.46 STATUS\_REG\_A (27h)

#### Table 92. STATUS\_REG\_A register

ZYXAOR	ZAOR	YAOR	XAOR	ZYXADA	ZADA	YADA	XADA
--------	------	------	------	--------	------	------	------

#### Table 93. STATUS\_REG\_A description

ZYXAOR	Acceleration X-, Y- and Z-axis data overrun. Default value: 0
	(0: no overrun has occurred; 1: a new set of data has overwritten the previous one)
ZAOR	Acceleration Z-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new set of data for the Z-axis has overwritten the pre- vious one)
YAOR	Acceleration Y-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new set of data for the Y-axis has overwritten the pre- vious one)
XAOR	Acceleration X-axis data overrun. Default value: 0 (0: no overrun has occurred; 1: a new set of data for the X-axis has overwritten the pre- vious one)
ZYXADA	Acceleration X-, Y- and Z-axis new value available. Default value: 0 (0: a new set of data is not yet available; 1: a new set of data is available)
ZADA	Acceleration Z-axis new value available. Default value: 0 (0: a new set of data for the Z-axis is not yet available; 1: a new set of data for the Z-axis is available)
YADA	Acceleration Y-axis new value available. Default value: 0 (0: a new set of data for the Y-axis is not yet available; 1: a new set of data for the Y-axis is available)
XADA	Acceleration X-axis new value available. Default value: 0 (0: a new set of data for the X-axis is not yet available; 1: a new set of data for the X- axis is available)

## 8.47 OUT\_X\_L\_A (28h), OUT\_X\_H\_A (29h)

X-axis acceleration data.

The value is expressed in 16 bit as two's complement left justified.



## 8.48 OUT\_Y\_L\_A (2Ah), OUT\_Y\_H\_A (2Bh)

Y-axis acceleration data.

The value is expressed in 16-bit as two's complement left justified.

## 8.49 OUT\_Z\_L\_A (2Ch), OUT\_Z\_H\_A (2Dh)

Z-axis acceleration data.

The value is expressed in 16-bit as two's complement left justified.

## 8.50 FIFO\_CTRL\_REG (2Eh)

#### Table 94. FIFO\_CTRL\_REG register

FM2 FM1 FM0	FTH4 F	FTH3 F	TH2	FTH1	FTH0
-------------	--------	--------	-----	------	------

#### Table 95. FIFO\_CTRL\_REG description

FM2-FM0	FIFO mode selection. Default value: 000
FTH4-FTH0	FIFO watermark level. Default value: 00000

#### Table 96. FIFO mode configuration

FM2	FM1	FM0	FIFO mode			
0	0	0	Bypass mode			
0	0	1	FIFO mode			
0	1	0	Stream mode			
0	1	1	Stream-to-FIFO mode			
1	0	0	Bypass-to-Stream mode			

Interrupt generator 2 can change the FIFO mode.

## 8.51 FIFO\_SRC\_REG (2Fh)

#### Table 97. FIFO\_SRC\_REG register

WTM	OVRN	EMPTY	FSS4	FSS3	FSS2	FSS1	FSS0
-----	------	-------	------	------	------	------	------

#### Table 98. FIFO\_SRC\_REG description

WTM	Watermark status.
	WTM bit is set to '1' when FIFO content exceeds watermark level.
OVRN	FIFO Overrun status.
	OVRN bit is set to '1' when FIFO buffer is full.



Table 98. FIFO\_SRC\_REG description (continued)

EMPTY	Empty status. EMPTY bit is set to '1' when all FIFO samples have been read and FIFO is empty.
FSS4-FSS0	FIFO stored data level. FSS4-FSS0 bits contain the current number of unread FIFO levels.

## 8.52 INT\_GEN\_1\_REG (30h)

This register contains the settings for the inertial interrupt generator 1.

Table 99	. INT	GEN	1 REG	register
	_			

AOI	6D	ZHIE/ ZUPE	ZLIE/ ZDOWNE	YHIE/ YUPE	YLIE/ YDOWNE	XHIE/ XUPE	XLIE/ XDOWNE
-----	----	---------------	-----------------	---------------	-----------------	---------------	-----------------

#### Table 100. INT\_GEN\_1\_REG description

AOI	And/Or combination of Interrupt events. Default value: 0. Refer to <i>Table 101: Interrupt mode</i>
6D	6-direction detection function enabled. Default value: 0. Refer to <i>Table 101: Interrupt mode</i>
ZHIE/ ZUPE	Enable interrupt generation on Z high event or on direction recognition. Default value: 0 (0: disable interrupt request;1: enable interrupt request)
ZLIE/ ZDOWNE	Enable interrupt generation on Z low event or on direction recognition. Default value: 0 (0: disable interrupt request;1: enable interrupt request)
YHIE/ YUPE	Enable interrupt generation on Y high event or on direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
YLIE/ YDOWNE	Enable interrupt generation on Y low event or on direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
XHIE/ XUPE	Enable interrupt generation on X high event or on direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
XLIE/XDOWNE	Enable interrupt generation on X low event or on direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)

#### Table 101. Interrupt mode

AOI	6D	Interrupt mode				
0	0	OR combination of interrupt events				
0	1	direction movement recognition				
1	0	AND combination of interrupt events				
1	1	6-direction position recognition				



The difference between AOI-6D = '01' and AOI-6D = '11' is as follows:

AOI-6D = '01' is movement recognition. An interrupt is generated when the orientation moves from an unknown zone to a known zone. The interrupt signal stays for a duration ODR.

AOI-6D = '11' is direction recognition. An interrupt is generated when the orientation is inside a known zone. The interrupt signal stays until the orientation is inside the zone.

## 8.53 INT\_GEN\_1\_SRC (31h)

This register contains the status for the inertial interrupt generator 1.

Table 102. INT_GEN_1_SRC register										
0	0 IA ZH ZL YH YL XH XL									

IA	Interrupt Status. Default value: 0 (0: no interrupt has been generated; 1: one or more interrupts have been generated)
ZH	Z high. Default value: 0 (0: no interrupt, 1: Z high event has occurred)
ZL	Z low. Default value: 0 (0: no interrupt; 1: Z low event has occurred)
YH	Y high. Default value: 0 (0: no interrupt, 1: Y high event has occurred)
YL	Y low. Default value: 0 (0: no interrupt, 1: Y low event has occurred)
хн	X high. Default value: 0 (0: no interrupt, 1: X high event has occurred)
XL	X low. Default value: 0 (0: no interrupt, 1: X low event has occurred)

#### Table 103. INT GEN 1 SRC description

Reading at this address clears the *INT\_GEN\_1\_SRC (31h)* IA bit (and the interrupt signal on the corresponding interrupt pin) and allows the refreshment of data in the *INT\_GEN\_1\_SRC (31h)* register if the latched option was chosen.

## 8.54 INT\_GEN\_1\_THS (32h)

Table 104. INT1_THS register								
0	THS6	THS5	THS4	THS3	THS2	THS1	THS0	
Table 105. INT1_THS description								

THS6 - THS0	Interrupt 1 threshold. Default value: 000 0000



## 8.55 INT\_GEN\_1\_DURATION (33h)

Table 106. INT1_DURATION register										
0	D6	D5	D4	D3	D2	D1	D0			

#### Table 107. INT1 DURATION description

	— — — — — — — — — — — — — — — — — — — —
D6 - D0	Duration value. Default value: 000 0000

The **D6** - **D0** bits set the minimum duration of the Interrupt 1 event to be recognized. Duration steps and maximum values depend on the ODR chosen.

## 8.56 INT\_GEN\_2\_REG (34h)

This register contains the settings for the inertial interrupt generator 2.

#### Table 108. INT\_GEN\_2\_REG register

					J		
AOI	6D	ZHIE/	ZLIE/	YHIE/	YLIE/	XHIE/	XLIE/
		ZUPE	ZDOWNE	YUPE	YDOWNE	XUPE	XDOWNE

#### Table 109. INT\_GEN\_2\_REG description

AOI	And/Or combination of Interrupt events. Default value: 0. Refer to Table 109: INT_GEN_2_REG description
6D	6 direction detection function enabled. Default value: 0. Refer to Table 109: INT_GEN_2_REG description
ZHIE/ ZUPE	Enable interrupt generation on Z high event or on direction recognition. Default value: 0 (0: disable interrupt request;1: enable interrupt request)
ZLIE/ ZDOWNE	Enable interrupt generation on Z low event or on direction recognition. Default value: 0 (0: disable interrupt request;1: enable interrupt request)
YHIE/ YUPE	Enable interrupt generation on Y high event or on direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
YLIE/ YDOWNE	Enable interrupt generation on Y low event or on Direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
XHIE/ XUPE	Enable interrupt generation on X high event or on Direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)
XLIE/XDOWNE	Enable interrupt generation on X low event or on Direction recognition. Default value: 0 (0: disable interrupt request; 1: enable interrupt request.)



#### Table 110. Interrupt mode

AOI	6D	Interrupt mode
0	0	OR combination of interrupt events
0	1	6-direction movement recognition
1	0	AND combination of interrupt events
1	1	6-direction position recognition

The difference between AOI-6D = '01' and AOI-6D = '11' is as follows:

AOI-6D = '01' is movement recognition. An interrupt is generated when the orientation moves from an unknown zone to a known zone. The interrupt signal stays for a duration ODR.

AOI-6D = '11' is direction recognition. An interrupt is generated when the orientation is inside a known zone. The interrupt signal stays until the orientation is inside the zone.

## 8.57 INT\_GEN\_2\_SRC (35h)

This register contains the status for the inertial interrupt generator 2.

Table 111.	INT	GEN	2	SRC	reaister

|--|

#### Table 112. INT\_GEN\_2\_SRC description

IA	Interrupt status. Default value: 0 (0: no interrupt has been generated; 1: one or more interrupts have been generated)
ZH	Z high. Default value: 0 (0: no interrupt, 1: Z high event has occurred)
ZL	Z low. Default value: 0 (0: no interrupt; 1: Z low event has occurred)
YH	Y high. Default value: 0 (0: no interrupt, 1: Y high event has occurred)
YL	Y low. Default value: 0 (0: no interrupt, 1: Y low event has occurred)
хн	X high. Default value: 0 (0: no interrupt, 1: X high event has occurred)
XL	X low. Default value: 0 (0: no interrupt, 1: X low event has occurred)

Reading at this address clears the *INT\_GEN\_2\_SRC (35h)* IA bit (and the interrupt signal on the corresponding interrupt pin) and allows the refreshment of data in the *INT\_GEN\_2\_SRC (35h)* register if the latched option was chosen.



## 8.58 INT\_GEN\_2\_THS (36h)

Table 113. INT_GEN_2_THS register										
0	THS6	THS5	THS4	THS3	THS2	THS1	THS0			

#### Table 114. INT\_GEN\_2\_THS description

THS6 - THS0	Interrupt 1 threshold. Default value: 000 0000

## 8.59 INT\_GEN\_2\_DURATION (37h)

Table 115	INT	GFN	2	DURATION register	
			~	DUINATION TEGISLET	

0	D6	D5	D4	D3	D2	D1	D0
---	----	----	----	----	----	----	----

#### Table 116. INT\_GEN\_2\_DURATION description

D6 - D0	Duration value. Default value: 000 0000

The **D6** - **D0** bits set the minimum duration of the Interrupt 2 event to be recognized. Duration steps and maximum values depend on the ODR chosen.

## 8.60 CLICK\_CFG (38h)

#### Table 117. CLICK\_CFG register

ZD ZS YD YS XD XS				ZD	ZS	YD	YS	XD	XS
-------------------	--	--	--	----	----	----	----	----	----

#### Table 118. CLICK\_CFG description

ZD	Enable interrupt double-click on Z-axis. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)
ZS	Enable interrupt single-click on Z-axis. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)
YD	Enable interrupt double-click on Y-axis. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)
YS	Enable interrupt single-click on Y-axis. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)



Table 118. CLICK_CFG description
Enable interrupt double-click on X-axis. Default value: 0
(0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)
Enable interrupt single-click on X-axis. Default value: 0 (0: disable interrupt request; 1: enable interrupt request on measured accel. value higher than preset threshold)

#### 8.61 CLICK\_SRC (39h)

#### Table 119. CLICK\_SRC register

	IA	DClick	SClick	Sign	Z	Y	х
--	----	--------	--------	------	---	---	---

#### Table 120. CLICK\_SRC description

IA	Interrupt active. Default value: 0 (0: no interrupt has been generated; 1: one or more interrupts have been generated)
DClick	Double-click enable. Default value: 0 (0: double-click detection disabled, 1: double-click detection enabled)
SClick	Single-click enable. Default value: 0 (0: single-click detection disabled, 1: single-click detection enabled)
Sign	Click sign. 0: positive detection, 1: negative detection
Z	Z click detection. Default value: 0 (0: no interrupt, 1: Z high event has occurred)
Y	Y click detection. Default value: 0 (0: no interrupt, 1: Y high event has occurred)
X	X click detection. Default value: 0 (0: no interrupt, 1: X high event has occurred)

#### CLICK\_THS (3Ah) 8.62

#### Table 121. CLICK\_THS register

- Ths6 Ths5 Ths4 Ths3 Ths2 Ths1 Ths	)
-------------------------------------	---

#### Table 122. CLICK\_SRC description

Ths6-Ths0	Click threshold. Default value: 000 0000



## 8.63 TIME\_LIMIT (3Bh)

		Table	e 123. TIME	LIMIT reg	ister		
-	TLI6	TLI5	TLI4	TLI3	TLI2	TLI1	TLI0

#### Table 124. TIME LIMIT description

	—	•
TLI7-TLI0	Click time limit. Default value: 000 0000	

## 8.64 TIME\_LATENCY (3Ch)

		Table 1	25. TIME_I	LATENCY r	egister		
TLA7	TLA6	TLA5	TLA4	TLA3	TLA2	TLA1	TLA0

#### Table 126. TIME\_LATENCY description

	•
TLA7-TLA0	Click time latency. Default value: 0000 0000

## 8.65 TIME WINDOW (3Dh)

#### Table 127. TIME\_WINDOW register

TW7 TW6 TW5 TW4 TW3 TW2 TW1 TW0
---------------------------------

#### Table 128. TIME\_WINDOW description

TW7-TW0	Click time window
---------	-------------------

## 8.66 Act\_THS (3Eh)

#### Table 129. TIME\_WINDOW register

	Acth6	Acth5	Acth4	Acth3	Acth2	Acth1	Acth0
--	-------	-------	-------	-------	-------	-------	-------

#### Table 130. TIME\_WINDOW description

Acth[6:0]	Sleep-to-Wake, Return-to-Sleep activation threshold
	1 LSb = 16 mg



## 8.67 Act\_DUR (3Fh)

ActD7

Table 131. Act_DUR register						
ActD6	ActD5	ActD4	ActD3	ActD2	ActD1	ActD0

#### Table 132. Act\_DUR description

ActD[7:0]	Sleep-to-Wake, Return-to-Sleep duration
	DUR = (Act_DUR + 1)*8/ODR



## 9 Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK<sup>®</sup> packages, depending on their level of environmental compliance. ECOPACK<sup>®</sup> specifications, grade definitions and product status are available at: www.st.com. ECOPACK is an ST trademark.

Databook				
Symbol	Min.	Тур.	Max.	Note
A			1.070	
A1	0.000	-	0.050	4
b		0.200		7
D		4.000		6
D2		1.750		
E		4.000		6
E2		1.750		
e1		0.500		
e2		2.500		
L		0.350		7
L1	-	0.100	-	
L2	-	0.100	-	
N		24		5
R1	-	0.080	-	

Table 133. LGA 4x4x1 mm 24-lead mechanical data (see note 1 and 2)

Symbol	Tolerance of Form and Position
	Databook
D/E	0.15
Notes	1 and 2
REF	-

Δ

Note:

- 1. Dimensioning and tolerancing schemes conform to ASME Y14.5M-1994.
- 2. All dimensions are in millimeters.
- 3. The "Pin 1 Indicator" is identified on top and/or bottom surfaces of the package.
- 4. A1 is defined as the distance from the seating plane to the land.
- 5. "N" is the maximum number of terminal positions for the specified body size.
- 6. The tolerance of the typical value is specified in table "Tolerance of Form and Position".
- 7. Dimensions "b" and "L" are specified: For solder mask defined: at terminal plating surface For non-solder mask defined: at solder mask opening



DocID024763 Rev 2

71/74



Figure 21. LGA 4x4x1 mm 24-lead outline



# 10 Revision history

Date	Revision	Changes
24-Jun-2013	1	Initial release
05-Aug-2013	2	Updated LA_So in <i>Table 3</i> Updated <i>Figure 4, Figure 5,</i> and <i>Table 7</i> Updated <i>Section 5.1</i> Updated <i>Section 9: Package information</i> Minor textual updates throughout <i>Section 8: Register description</i>

#### Table 134. Document revision history



#### Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

ST PRODUCTS ARE NOT AUTHORIZED FOR USE IN WEAPONS. NOR ARE ST PRODUCTS DESIGNED OR AUTHORIZED FOR USE IN: (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS; (B) AERONAUTIC APPLICATIONS; (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER'S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR "AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL" INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

#### © 2013 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan -Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com



October 2005

## **BSS138** N-Channel Logic Level Enhancement Mode Field Effect Transistor

#### **General Description**

These N-Channel enhancement mode field effect transistors are produced using Fairchild's proprietary, high cell density, DMOS technology. These products have been designed to minimize on-state resistance while provide rugged, reliable, and fast switching performance. These products are particularly suited for low voltage, low current applications such as small servo motor control, power MOSFET gate drivers, and other switching applications.

#### Features

• 0.22 A, 50 V.  $R_{DS(ON)}$  = 3.5 $\Omega$  @ V<sub>GS</sub> = 10 V  $R_{DS(ON)}$  = 6.0 $\Omega$  @ V<sub>GS</sub> = 4.5 V

- High density cell design for extremely low R<sub>DS(ON)</sub>
- Rugged and Reliable
- Compact industry standard SOT-23 surface mount package





## Absolute Maximum Ratings T<sub>A=25°C</sub> unless otherwise noted

Symbol		Paramete	r	Ratings	Units
V <sub>DSS</sub>	Drain-Source Voltage			50	V
V <sub>GSS</sub>	Gate-Source Voltage			±20	V
lo	Drain Current	– Continuous	(Note 1)	0.22	A
		- Pulsed		0.88	
<b>&gt;</b> <sub>D</sub>	Maximum Po	wer Dissipation	(Note 1)	0.36	W
	Derate Above	e 25°C		2.8	mW/°0
Γ <sub>J</sub> , T <sub>STG</sub>	Operating and Storage Junction Temperature Range			-55 to +150	°C
Г,	Maximum Lead Temperature for Soldering Purposes, 1/16" from Case for 10 Seconds			300	°C
	Fulposes, 1/		00001103		
Therma	I Characte	eristics			I
Therma	<b>Characte</b> Thermal Resi	eristics stance, Junction-to-	-Ambient (Note 1)	350	°C/W
Therma R <sub>θJA</sub> Packag Device	I Characte Thermal Resi e Marking Marking	eristics stance, Junction-to- and Orderir Device	Ambient (Note 1)	350 Tape width	C/W

©2005 Fairchild Semiconductor Corporation

BSS138 Rev C(W)

**BSS138** 

Symbol	Parameter	Test Conditions	Min	Тур	Мах	Units
Off Char	acteristics	I				
BV <sub>DSS</sub>	Drain-Source Breakdown Voltage	$V_{GS} = 0 V$ , $I_D = 250 \mu A$	50			V
<u>ΔBVdss</u> ΔTj	Breakdown Voltage Temperature Coefficient	$I_D$ = 250 µA,Referenced to 25°C		72		mV/°C
DSS	Zero Gate Voltage Drain Current	$V_{\text{DS}} = 50 \text{ V}, \qquad V_{\text{GS}} = 0 \text{ V}$			0.5	μA
		$V_{DS} = 50 \text{ V}, V_{GS} = 0 \text{ V} \text{ T}_{J} = 125^{\circ}\text{C}$			5	μA
		$V_{DS} = 30 \text{ V},  V_{GS} = 0 \text{ V}$			100	nA
GSS	Gate-Body Leakage.	$V_{GS}=\pm 20~V, ~~V_{DS}=0~V$			±100	nA
On Chara	acteristics (Note 2)					
V <sub>GS(th)</sub>	Gate Threshold Voltage	$V_{DS} = V_{GS}$ , $I_D = 1 \text{ mA}$	0.8	1.3	1.5	V
ΔV <sub>GS(th)</sub> ΔT <sub>J</sub>	Gate Threshold Voltage Temperature Coefficient	$I_D = 1$ mA,Referenced to $25^{\circ}C$		-2		mV/°C
R <sub>DS(on)</sub>	Static Drain–Source	$V_{GS} = 10 \text{ V}, \qquad I_D = 0.22 \text{ A}$		0.7	3.5	Ω
	On-Resistance	$V_{GS} = 4.5 V$ , $I_D = 0.22 A$		1.0	6.0	
		$V_{GS} = 10 \text{ V}, \text{ I}_{D} = 0.22 \text{ A}, \text{ T}_{J} = 125^{\circ}\text{C}$		1.1	5.8	
D(on)	On-State Drain Current	$V_{GS} = 10 V$ , $V_{DS} = 5 V$	0.2	0.5		A
FS	Forward Transconductance	$V_{DS} = 10V, I_D = 0.22 \text{ A}$	0.12	0.5		5
Dynamic	Characteristics					
Ciss	Input Capacitance	$V_{DS} = 25 V$ , $V_{GS} = 0 V$ ,		27		pF
Coss	Output Capacitance	f = 1.0 MHz		13		pF
Crss	Reverse Transfer Capacitance			6		pF
R <sub>G</sub>	Gate Resistance	$V_{\text{GS}} = 15 \text{ mV},  f = 1.0 \text{ MHz}$		9		Ω
Switchin	g Characteristics (Note 2)					
d(on)	Turn–On Delay Time	$V_{DD} = 30 \text{ V}, \qquad I_D = 0.29 \text{ A},$		2.5	5	ns
r	Turn–On Rise Time	$V_{GS} = 10 \text{ V}, \qquad R_{GEN} = 6 \Omega$		9	18	ns
d(off)	Turn-Off Delay Time			20	36	ns
f	Turn–Off Fall Time			7	14	ns
Qg	Total Gate Charge	$V_{DS} = 25 \text{ V}, \qquad I_D = 0.22 \text{ A},$		1.7	2.4	nC
Q <sub>gs</sub>	Gate-Source Charge	V <sub>GS</sub> = 10 V		0.1		nC
Q <sub>gd</sub>	Gate–Drain Charge			0.4		nC
Drain-So	ource Diode Characteristics	and Maximum Ratings				
s	Maximum Continuous Drain-Source	e Diode Forward Current			0.22	А
V <sub>SD</sub>	Drain–Source Diode Forward Voltage	$V_{GS}=0~V, \qquad  I_S=0.44~A(\text{Note 2})$		0.8	1.4	V
otes: R <sub>0JA</sub> is the su the drain pins	Voltage um of the junction-to-case and case-to-ambient the s. $R_{BUC}$ is guaranteed by design while $R_{BCA}$ is dete	rmal resistance where the case thermal reference	e is defined	d as the so	blder moun	v ting surfac

Scale 1 : 1 on letter size paper

2. Pulse Test: Pulse Width  $\leq 300~\mu\text{s},~\text{Duty}~\text{Cycle} \leq 2.0\%$ 

**BSS138** 

BSS138 Rev C(W)



**BSS138** 

BSS138 Rev C(W)



**BSS138** 

BSS138 Rev C(W)

TRADEMARKS					
The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.					
ACExTMFASTActiveArrayTMFASTBottomlessTMFPSTBuild it NowTMFRFECoolFETTMGlobaCROSSVOLTTMGTOTDOMETMHiSelEcoSPARKTMI2CTME²CMOSTM $i-LoTT$ EnSignaTMImplieFACTTMIntellillFACT Quiet SeriesTMAcross the board. ArourThe Power Franchise®Programmable Active D	<sup>®</sup> <sup>™</sup> <sup>™</sup> <sup>™</sup> <sup>™</sup> alOptoisolator <sup>™</sup> <sup>™</sup> C <sup>™</sup> <sup>™</sup> edDisconnect <sup>™</sup> MAX <sup>™</sup> MAX <sup>™</sup>	ISOPLANAR <sup>TM</sup> LittleFET <sup>TM</sup> MICROCOUPLER <sup>TM</sup> MicroFET <sup>TM</sup> MICROWIRE <sup>TM</sup> MSX <sup>TM</sup> MSXPro <sup>TM</sup> OCX <sup>TM</sup> OCXPro <sup>TM</sup> OCXPro <sup>TM</sup> OPTOLOGIC <sup>®</sup> OPTOPLANAR <sup>TM</sup> PACMAN <sup>TM</sup> POP <sup>TM</sup>	PowerSaver <sup>™</sup> PowerTrench <sup>®</sup> QFET <sup>®</sup> QS <sup>™</sup> QT Optoelectronics <sup>™</sup> Quiet Series <sup>™</sup> RapidConfigure <sup>™</sup> RapidConnect <sup>™</sup> µSerDes <sup>™</sup> ScalarPump <sup>™</sup> SILENT SWITCHER <sup>®</sup> SMART START <sup>™</sup> SPM <sup>™</sup> Stealth <sup>™</sup> SuperFET <sup>™</sup>	SuperSOT <sup>™</sup> -6 SuperSOT <sup>™</sup> -8 SyncFET <sup>™</sup> TinyLogic <sup>®</sup> TINYOPTO <sup>™</sup> TruTranslation <sup>™</sup> UHC <sup>™</sup> UltraFET <sup>®</sup> UniFET <sup>™</sup> VCX <sup>™</sup> Wire <sup>™</sup>	

#### DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

#### LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user. 2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

#### PRODUCT STATUS DEFINITIONS

#### **Definition of Terms**

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
Obsolete	Not In Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.
		Rev. 117





1-800-383-7323 USA/CAN www.energizer.com



## Specifications



**ENERGIZER LA522** 

Advanced Lithium











Classification: Chemical System: Designation: Nominal Voltage: Operating Temp: Storage Temp: Max Discharge : Safety Features:

Typical Weight: Typical Volume: Jacket: Terminal: Shelf Life: Typical Li Content: UL/UN Listed: Lithium 9V Lithium-Manganese Dioxide (Li/MnO<sub>2</sub>) ANSI-1604LC 9.0 volts -40°C to 60°C (-40°F to 140°F) -40°C to 60°C (-40°F to 140°F) 1000 mA continuous Positive Temperature Coefficient Switch (PTC) Burst Proof Venting Holes 33.9 grams (1.2 oz.) 21.4 cubic centimeters (1.3 cubic inch) Plastic Label Miniature Snap 10 Years 1.35 grams (0.048 oz.) MH47482





Continuous Discharge to 5.4 volts 1000 750 500 250 0 200 500 1000 Discharge (mA)

# 500 mA Continuous Discharge



#### **Important Notice**

8

This datasheet contains typical information specific to products manufactured at the time of its publication. ©Energizer Holdings, Inc. - Contents herein do not constitute a warranty. 130

### Unity System Requirements

1- Generally content developed with Unity can run pretty much everywhere. How well it runs is dependent on the complexity of your project. More detailed requirements:

- Desktop:
  - o OS: Windows XP+, Mac OS X 10.6+, Ubuntu 10.10+, SteamOS+
  - Graphics card: DX9 (shader model 2.0) capabilities; generally everything made since 2004 should work.
  - CPU: SSE2 instruction set support.
  - Web player supports IE, Chrome, Firefox, Safari and others.
- iOS: requires iOS 4.3 or later.
- Android: OS 2.3.1 or later; ARMv7 (Cortex) CPU; OpenGL ES 2.0 or later.
- Blackberry: OS 10 or later.



Yan Lin <wildseven@gmail.com>

Sat, Sep 6, 2014 at 5:43 PM

## Question for a design project

11 messages

**Yan Lin** <wildseven@gmail.com> To: Ray Stedronsky <ladder106@gmail.com>

Hey Ray,

I'm working on a design project for school and we're essentially creating a dark device that can reverse-track your movement so you can find your way back to where you started. Navigation is calculated from predetermined street locations to find the shortest route. The device isn't reliant on GPS because it doesn't work well in instances where a direct line of sight is obscured by rubble or an underground environment. An example of where this is applicable is search and rescue in a collapsed building and you want to find your way back to your original exit quickly and reliably.

That said, I was wondering if you knew and might be able to share how this is typically done in an unfamiliar or damaged environment. Our guess is in the case where there's a fire, firefighters dragging the hose in and following it back seems reliable, but I'm not sure if that's actually how how it's executed in the real world in other regards.

Don't worry about it if you're extremely busy. Just thought I'd ask in case you knew off hand.

Hope all is well with you and Carole! Say hi to her from me.

Yan

Ray Stedronsky <ladder106@gmail.com> To: Yan Lin <wildseven@gmail.com> Sat, Sep 6, 2014 at 6:50 PM

Yan,

Happy to help. I'll write up a description for how the FD does searches in buildings. Hope that will help.

Would it be better if I just called you or would a written thing be best.

I'll do it this evening if your not short on time.

R

[Quoted text hidden]

Ray Stedronsky Davis, CA

Yan Lin <wildseven@gmail.com> To: Ray Stedronsky <ladder106@gmail.com> Sat, Sep 6, 2014 at 7:04 PM

Ray,

A phonecall would be great so you don't have to type it all up. I'll be on campus working on this for a few hours so feel free to call whenever– I have a few days before I need to hand this in. This information is going to be

really useful for my group members and myself in doing the initial pitch. I think you have my number but in the event that you don't, it's 530.574.4779.

My partners and I really appreciate your assistance in this matter.

Yan

**Ray Stedronsky** <ladder106@gmail.com> To: Yan Lin <wildseven@gmail.com> Sat, Sep 6, 2014 at 11:20 PM

Yan,

Got involved with other stuff...sorry. Since it's now 2230, I'll write it up. Feel free to call with questions or for clarification.

Some basic conditions and principles:

Fires make black smoke. Most of the stuff in living areas today is plastic and burns like any hydrocarbon.

Heated gasses and smoke rise to the ceiling in a room and fill the room from top to bottom.

Depending on how much the fire is contained within the room or building and how much "clean" airflow is possible into the fire area, there MAY be a clear or semi-clear area a foot or so off the floor where some visibility is possible. The "clean" air is cooler and denser to it's always down low.

Most searches are done by the crew of a Truck or Ladder Company in a Fire Dept of any size. Hoselines are NOT involved in a search because they take too long to deploy slowing the search process and wearing you out dragging 300 lbs of hose and water through the area.

Flashlights are only about 10% effective in smoke.

Searches are done while crawling to get below the heat either on hands and knees or on your stomach like a sniper.

You are wearing a breathing apparatus mask which limits your field of vision, heavy coat and pants and heavy gloves that impair your tactical sense. You are carrying tools that will allow you to breach a door or wall if necessary.

Searches are generally most effective with two-man teams but sometimes as many as four are used for searches in a large open space like a warehouse.

OK, so now the How-To:

11/29/2014

The basic search method involves the walls. You start at the front door (for example) and decide to keep the wall on your right hand side. This is called a "right hand search" for obvious reasons. The other way would be to start on the left for a "left hand" search.

Start at your front door for example (although any entry point will work). Walk or crawl along always keeping the wall on your right side. If you imagine yourself doing this you'll see that eventually you will arrive back at the front door.

To speed the process up under fire conditions a team will travel crawling together with the second man just behind and off to the side of the first. So, the first man has the wall on his right, the second will have the left leg of the first man on his right and be behind and slightly to the left covering more floor area. (a left hand search will be the opposite).

You're trying to cover as much ground as fast as possible because it's hot and any humans inside don't have much survival time left.

You notice things like floor coverings. Carpet means living room, bedroom, hallways. Tile/linoleum means bathrooms, kitchens etc.

You will also note that doors swing open a specific way when you are inside a living area. Bedroom doors swing inward and this can be felt when crawling down the hallway. Inward swinging doors are set deeper into the door frame and this is easy to feel. Outward swinging doors in living areas are generally closets, heat/ac cupboards, and also the door leading into the garage in ranch style houses with attached garages.

This is critical information because once a bedroom is located, the team splits. One man stays by the doorway while the other enters the bedroom to do a quick search of that area. Most bedrooms are cluttered so while you are still doing a "right-hand" search, you're running into dressers, piles of clothes, bicycles, beds (that have to be searched themselves both on top making sure you don't miss a bunk bed and underneath for hiding kids) and lots of other things. It's easy to get spun around in a bedroom so it's the job of the guy at the door to keep track of time and also to answer your yells if you get lost. You can generally follow the sound back to him since bedrooms aren't that large.

Also, bedrooms are nice because they have a door and at least one window. This means that if fire conditions go to shit, you both can crawl inside the bedroom, shut the door (that will keep the fire out for at least a minute or two), break the window letting fresh air inside and improving visibility and giving you a way to bail out to the exterior.

Once one bedroom is done, you go out, continue the right-hand path and find another. Generally you trade jobs with the searcher of the previous room taking the doorway job on

the next one to share the workload.

Knowing the area you are working in and the general way buildings are constructed is a major part of the firefighter's job. For example. If you're working in the suburbs with mainly tract/ranch style housing, it's obvious when entering the front door which way (right or left) most of the living spaces or bedrooms are located. If you're downtown or in an area with lots of apartments, you learn over time that certain floorplans are used over and over. For example, some downtown apartments are layed out with the bedrooms to the left on the right side of the hallway and on the right on the left side...etc, etc with lots of variations. This is just one of the reasons that experience is passed down from older to newer firefighters .

## OK...now "Large Area" searches:

This involves a 200 ft length of rope with 4, 20 ft lengths of individual rope. The rope is marked with a series of knots and metal rings about the size of a silver dollar. The knot/ring sequence gives you your distance down the rope every 50 ft. Remember that you generally can't see well or at all so visual cues don't work.

The search is done with 2 to 4 firefighters and would be done in warehouses, supermarkets, etc.

The rope is anchored to a strong point when you make entry and the rope bag is carried with the first man inside. He picks the direction. The second man counts the knots/rings and calls out distance. The others just follow along on the rope.

Using auditory clues, you can usually tell when you are in a large open space. At this point the individual ropes can be deployed. Remember this is usually done by "feel" only. The lead man stops and calls for a "radius search". The second man clips his 20 ft rope onto the main line and goes out 20 ft. The third man follows that 20 ft. line, then clips his 20 ft on and goes to the end....etc. Once the line or lines are taut, the one, two or three guys then start moving in a circle. The man with the rope bag is the center and it's his job to tell the team when the circle is complete (otherwise they just keep going).

As you can imagine, this looks, diagramatically, like a large lollipop on a map.

Variations on this would be used in a grocery store-like area with large isles where the lead man would stop at an intersection and the short ropes could be deployed to the sides.

Other variations are used depending on your creativity and how well drilled your team is.

So, that's the short version of only one of the fun things you get to do as a fireman.

You guys should all try this at home or at someone's apartment. It's can be done by putting a hoodie on backwards and the hood over your face. But the best way is to cover the inside of a pair of lab or shop goggles with a layer of wax paper - then turn the lights out and use a flashlight. The very best learning experience comes by having someone video the guy doing the search so he can watch his actions later.

Hope that explains it well enough. Call with questions and tell me exactly what you guys are trying to do. Once you come up with something practical I can likely tell you if you're gonna be rich or that your idea sucks.

R

[Quoted text hidden] --Ray Stedronsky Davis, CA

Yan Lin <wildseven@gmail.com> Sun, Sep 7, 2014 at 12:27 AM To: Andre Julien <a href="mailto:sandre-julien@gmail.com">a sisharks23@gmail.com</a>, alexander.s.meadows@gmail.com

[Quoted text hidden]

Yan Lin <wildseven@gmail.com> To: Ray Stedronsky <ladder106@gmail.com> Sun, Sep 7, 2014 at 12:33 PM

Ray,

This is extremely useful information. Thank you so much for the time you took to write it all up, as well as the information.

I'll give a short rundown of the idea we have. We've done a bunch of research regarding GPS, and we realized one of the faults is the system is that you can't get a signal in indoor environments, subterranean locations, and areas with thick foliage canopies. The way a few companies have done indoor positioning is by using Wi-Fi, indoor GPS repeaters, etc. This doesn't really help when you have a mass of different implementations for indoor navigation, and you need to quickly extract people from a building safely.

To address the issue, and potentially apply it to search and rescue teams, we're looking at using inertial measurement units, and exploiting acceleration, gyroscopic, pitch, yaw, and roll data to generate positional data from a given starting point. The idea is to be able to start from an entrance of a building, begin recording data, and be able to create a "bread crumb" type of trail back to your origin. Right now we just want to do a proof-of-concept, but the idea is sort of like this:

https://www.youtube.com/watch?v=6ijArKE8vKU

Initially we're looking to just generate positional coordinates in a 3D space by double-integrating acceleration data, but apparently this is difficult due to the amount of noise devices create. You need to create a Kalman filter to create a linear approximation of the data, so it's smooth enough so when you integrate the data, your noise doesn't generate issues in velocity, as well as position.

From my understanding, the visual component is not applicable due to smoke in a burning building, but as a

#### Gmail - Question for a design project

result we think we can adapt it into a auditory cue that repeatedly tells you where you originally entered the room/building/etc when you want to exit. The initial concept has a lot of applications. The target product here is a dark device that can provide self-referential positional data.

Sorry for the block of text. Any feedback would be fantastic, but we're not in dire need of it- we likely won't have to tackle the actual application portion until early next year. We're just doing the preliminary proof of concept right now. Thanks once again for the detailed account of the work you do!

Yan [Quoted text hidden]

Ray Stedronsky <ladder106@gmail.com> To: Yan Lin <wildseven@gmail.com> Sun, Sep 7, 2014 at 12:43 PM

If you guys could get something like this to work and get it small enough (like iphone sized) to carry, you could sell the thing to many manufacturers of firefighting gear.

Integrating it into the breathing apparatus mask as a HUD would be excellent but however it's done it would be a life-saving device. One of the highest line-of-duty deaths in the FD is getting lost and disoriented.

If there's anything I can do to help, let me know. We have mazes already set up to train recruits in searching, breathing apparatus use, etc. Using those might be a good test site for your idea.

I'd be very interested in helping out.

R [Quoted text hidden]

Yan Lin <wildseven@gmail.com> To: Ray Stedronsky <ladder106@gmail.com> Sun, Sep 7, 2014 at 1:06 PM

This was actually the engineering journal article that gave me the idea for contacting you about this topic specifically.

http://spectrum.ieee.org/static/how-new-indoor-navigation-systems-will-protect-emergency-responders

An IMU is extremely lightweight, and has low battery consumption. I'll give you more details as we progress. [Quoted text hidden]

**Ray Stedronsky** <ladder106@gmail.com> To: Yan Lin <wildseven@gmail.com> Sun, Sep 7, 2014 at 1:24 PM

That whole system seems to be based on finding guys who are already lost.

It's kinda like the Stop-Drop and Roll they used to tell us to teach school kids. I NEVER liked that idea at all. It's basic assumption is that you are ALREADY on fire when the major important point is NOT catching on fire and surviving in the first place. We used to teach them to crawl and find doors instead.

So, I like your idea much better. Hey...let's NOT get lost in the first place, or if we do, we find our own way back.

If you could combine the IMU with a HALO-like tracking device that would send out a sonar-type signal and give you a "snapshot" of the walls and furniture in your environment and update every one or two seconds like HALO, you would have the perfect set-up.

In all cases doing this in a "hand-held" type box is just marginally OK. We use thermal imaging cameras (TICs) they are hand-held, and too large. If you bring this device right up to your facepiece in your mask you can see it. Add to the fact that we already have our hands full and need both of them to drag some 300 lb. fat-ass out of the building, a HUD type device that integrates with the breathing apparatus mask is by far the best way to use electronics.

R

[Quoted text hidden]

Yan Lin <wildseven@gmail.com> To: Ray Stedronsky <ladder106@gmail.com> Sun, Sep 7, 2014 at 1:33 PM

There's another group in the class that's doing machine vision, and attempts to read a room using multiple cameras to create a 3D map I think. The SONAR idea is really awesome, but the one concern I have is by using a signal that you need to recapture to form a snapshot of the walls and furniture, you have to have several sensors that operate at a frequency that your transducers can reliably read the data back from assuming the environment isn't somehow distorting your signals. Those are just concerns I suppose may happen, but I thought along the same lines. I Google Image searched firefighter breathing apparatuses and you have a lot of real estate to work with on those visors. I'm sort of surprised it isn't already implemented at this point. [Quoted text hidden]

**Ray Stedronsky** <ladder106@gmail.com> To: Yan Lin <wildseven@gmail.com> Sun, Sep 7, 2014 at 2:39 PM

It's a matter of cost and ruggedness.

The stuff gets hot, wet, mashed, pounded on, twisted, caught on wires, nails and bend metal.

That plus it has to go on in less than a minute. No time to be doing the "plug and play" thing.

Also exposed wiring sucks. Gets ripped out in a short time. It's like giving stuff to a bunch of very strong 4 year olds.

Then there's cost. The breathing app. is very expensive as is.

But you're right you have space to carry units on the backpack and lots of clear mask to work a HUD into. Would be nice to get all that stuff plus your radio in one package. [Quoted text hidden]
B. Resume

# **Andre Julien**

andrejulien@csus.edu

# **EDUCATION**

*In progress:* **BS, Computer Engineering**; CSU Sacramento; GPA 3.59; Dec 2015 **AS Mathematics and Physical Science**; American River College **Programming Certificate**; American River College

#### **Courses:**

Advanced Computer Design Computer Hardware Design Java Language Programing Advanced Logic Design C Programing Electronics 1 Computer Interfacing Data Structures and Algorithm Analysis Probability and Random Signals

# **PROJECTS**

#### **Dual Stage Facial Detection with Proximity Activation Lock**

As part of a three-person team implemented a dual stage lock by interfacing a web camera, the Raspberry Pi, and the Parallax microcontroller.

# **Dark Inertial Mapping**

As part of a four person team implementing passive emergency personnel path mapping, to facilitate incident command.

### **MIPS** Pipeline Data path

As part of a two-person team implemented a six stage, sixteen bit instruction, MIPS pipeline data path using the Synopsys tool. Implemented basic integer arithmetic, floating point multiplication, register forwarding and hazard detection.

# **KNOWLEDGE AND SKILLS**

Computer Languages: C • Perl • Verilog • X86 Assembly • Java • Turbo Pascal • Visual Basic • HTML

Systems: Linux • Unix • Mac OSX • Windows

Software Applications: Quartus • XIIinx ISE • VIM • Multisim • PSpice • GDB • JGrasp • MS Office

Tools: Synopsys, Digilent Analog Discovery • Oscilloscope • Function Generator

### Communication/Organization/Leadership:

- Extensive experience managing business workflow
- 6+ years experience managing multidisciplined technical teams while coordinating with other essential departments
- Bilingual: English/French

# WORK EXPERIENCE

Hardware Validation Engineering Intern	Intel Corporation	9/14-Current
Service Manager	PCP Motorsports	8/04-8/11
Jr Systems Administrator	PCP Motorsports	1/06-1/11
PDI Technician/Service Writer	PCP Motorsports	1/97-3/04
Package Handler	<b>United Parcel Service</b>	1/96-1/01

# ACTIVITIES AND ACCOMPLISHMENTS

Dean's Honor List

• Member, IEEE

- Tau Beta Pi
- · Hold US, Canadian, and French Citizenship

11740 Barbwire Ct, Elk Grove, CA 95624

**OBJECTIVE:** A career in the Computer Engineering industry

#### **EDUCATION:**

In Progress: BS, Computer Engineering, CSU Sacramento, May 2015 (3.2 Sacramento State GPA) Computer Engineering, UC Davis April 2010

### **RELATED COURSES:**

Senior Design Project \* Computer Hardware Design Electronics I Advanced Computer Organization CMOS and VLSI Introduction to Microprocessors

\* Current Semester

# KNOWLEDGE AND SKILLS

## Senior Project:

Indoor Pedestrian Mapping Utilizing Dead-Reckoning. In a 4-person team, conducted research regarding the technological needs of Search and Rescue organizations. Devised a device and accompanying software to improve the coordination efforts of both individual search personnel, as well as search coordinators.

#### Computer Languages:

C(++), Verilog, Python, MIPS, X86 Assembly, HTML

#### Systems:

Windows, OS X, Linux, Unix

#### Software Applications:

Cadence PSPICE, Cadence Virtuoso, MultiSIM, Microsoft Office, VIM

#### **PROJECT EXPERIENCE:**

### MIPS Pipelined Datapath

As part of a 2-person team, developed a 6-stage pipelined datapath, using the Synopsys tool. Implemented basic integer arithmetic as well as floating point multiplication, register forwarding, and hazard detection in a 16-bit instruction format.

#### **CMOS level design for Flash ADC**

Worked as part of a 2-person team to design both the gate-level and layout-levels for the decode logic 4-bit Flash Analog-to-Digital Converter using bubble suppress logic. Overall design was implemented using the Cadence Virtuoso tool using a 0.18um process.

#### MIDI Arduino interface, I2C Programmer

As part of a 3-person group, tasked with the implementation of a slave device using I2C for a MIDI-Arduino interface that received file structure data, and displayed it to an 8x12 LCD.

#### WORK EXPERIENCE:

#### Instructional Student Assistant California State University Sacramento

Provided assistance to laboratory instructor and students in the 117L Network Analysis laboratory for Spring 2014. Answered conceptual queries, provided troubleshooting for PSPICE and circuit builds, and graded student lab reports.

### IS Support Desk Analyst

Specialty Care

9/11 - 3/12

2/14 - 5/14

Facilitated the transition of services from former company during acquisition while maintaining previous responsibilities.

Supported end-users with IT-related problems. Established and maintained a routine preventative

#### **IT Specialist**

#### Active Diagnostics

12/09 - 9/11

ACTIVITIES AND ACCOMPLISHMENTS:

maintenance system for electronic medical equipment

Dean's Honor Roll Spring 2014

#### 530.574.4779

yanhlin@gmail.com

142

# **Alexander Meadows**

Folsom, CA, 95630

### **EDUCATION**

BS, Electrical/Electronics Engineering, CSU Sacramento; 3.927; May 2015 AS, Multidisciplinary Math and Science, Folsom Lake College; 4.0; May 2012 EIT Certified

### **Related Courses**

Device Physics *	Applied Electromagnetics	Network Analysis Lab
Advanced Analog Circuits Lab*	CMOS & VLSI	Feedback Systems
Signals & Systems	Modern Communication Systems	Differential Equations
Electronics II	Probability and Random Signals	Senior Project*

\* In progress as of Spring 2015

## SKILLS

### **Problem Solving:**

Methodically search for the source of a problem, applying both research and creative thinking when an unknown problem arises. An example of this is getting a 32 bit driver to work on a 64 bit machine by using a virtual machine.

#### Design:

Knowledge of designing circuits to fit practical needs. Including troubleshooting and debugging. Experience using FPGAs and Microprocessors. Designed a CPU in Verilog and a three bit adder chip in both P-Spice and L-edit.

#### **Project Management:**

Including delegation based on skill and specialty, economic and time consideration. Motivation and communication with project members, and leadership through example. An example of this is leading the Bayside Folsom Church tech team as their production manager, improving setup and tear-down efficiency by 30 minutes.

#### Computer Skills:

P-Spice, Multisim, Quartus II, ModleSim, L-Edit, XSI, Putty, Notepad++, Rhino, GitHub, and VM ware. Known Languages include: Basic levels of C, Verilog, JavaScript, and Spin.

### WORK EXPERIENCE

Production Manager	Bayside Church, Folsom Campus	6/14 to present
Student Researcher	CSUS Smart Grid	11/13 - 8/14
Lab Assistant	CSUS EEE department	1/14 - 5/14
Learning Assistant	CSUS MSTI	8/13 - 5/14

### PROJECTS

### Paid Research: Smart Energy Device Test Plan

Determined the effectiveness of smart energy devices for the California Energy Commission. With a team of 9 researched the available devices, developed a test plan, and tested the devices. My role in this project was to gather information concerning the capabilities of the devices and how to install them.

# School Project: Air G

Working with another engineering student, we designed, built and programmed a glove based air guitar using flex sensors, an accelerometer, and an Arduino microprocessor. It included user definable notes, programmable tones, volume control, and variable tone length based off strumming speed.

### School Project: Three Bit Adder

My partner and I designed a three bit added chip using only NAND and transmission gates. This design process included calculating the how much each gate was driving. We then laid out this design in L-edit for a .5 micron process, taking into consideration parasitic capacitance and cross talk.

# **ACCOMPLISHMENTS and ACTIVITIES**

- Tau Beta Pi Engineering Honors Society
- Deans Honor Roll
- Delta Epsilon Iota Honor Society
- Campus Impact Award

Working 10 hours per week, while carrying 14 units per semester and maintaining a 3.9 GPA

### Alexander.s.meadows@gmail.com

Joey Youngblood 2501 Hurley Way Apt 53 Sacramento, California 95825 (925) 858-0771 jyoungblood@csus.edu

# Work Experience:

LSD APPS:

Lead Android Engineer

- Worked on developing an Android app that would be used by health and fitness professionals.
- Built API for interfacing mobile device with mysql server
- \_ Created an engine in php for use by internal programmers

## **IRT Service Desk at CSUS:**

Faculty Staff Technology Consultant/

- Student Faculty and Staff Technology Consultant Sacramento, CA Taking phone calls and walk-in appointments with users who needed help with personal and company
  - machines.
  - Creating documentation and provisioned users for specific services.

**MESA:** 

Student Mentor/Tutor	Sacramento, CA
- Mentoring incoming freshmen.	
- Tutoring lower division computer engineering and mathematics courses.	
Togo's:	2007 - 2010
Food Prep/Cashier	Livermore, CA

Food Prep/Cashier

Managed restaurant-wide closing responsibilities and customer service issues.

Provided quality customer service and prepared food.

### **Schooling:**

**California State University Sacramento** Major: Computer Engineering

Projected Graduation: Spring 2015

2014 - current

2010 - current

2010 - 2014

Minor: Math

GPA: 2.83

# **Livermore High School**

Several college level courses: Stats, Econ, English, Computer Science GPA: 3.0

# Technical:

- Languages: C/C++(7 years), Visual Basic (7 years), Java (5 years), C#/ JavaScript/Python (3 years) •
- Web Design: HTML 5 (4 years), PHP (2 year), JQuery (2 year), MySQL (2 year) •
- IDE: Jgrasp, Eclipse, MS Visual Studio •
- Game Engine: Unity (3 years), Project Anarchy (2 year), Unreal Engine (2 year),
- OS: Android, Windows, Mac OS X, Ubuntu

# **Relevant Course Work:**

- Numerical Analysis & Applied • Mathematics
- **Differential Equations**
- Computer Hardware Design
- Advanced Logic Design

- **Mechanical Physics** ٠
- **Electrical Physics**
- Artificial Intelligence
- ٠ Computer Networks & Internets