Development of a GPS-enabled Localization Device

by

Farzad Hamidi

B.Sc., Purdue School of Engineering & Technology, 2011

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Applied Science

in the
School of Mechatronic Systems Engineering
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY

Summer 2015

Approval

Name:	**Farzad Hamidi**
Degree:	**Master of Applied Science (Mechatronic Systems Engineering)**
Title:	***Development of a GPS-enabled Localization Device***
Examining Committee:	**Chair:** Dr. Siamak Arzanpour Associate Professor of Mechatronic Systems Engineering

Mehrdad Moallem
Senior Supervisor
Professor of Mechatronic Systems Engineering

Jiacheng Wang
Supervisor
Associate Professor of Mechatronic Systems Engineering

Kevin Oldknow
Internal Examiner
Senior Lecturer of Mechatronic Systems Engineering

Date Defended/Approved:

Partial Copyright Licence

Abstract

There exists a dichotomy in the design of modem electronic systems: the simultaneous need to be low power and high performance. This arises largely from their usage in battery-operated portable (wearable) platforms. Accordingly, the goal of low-power design for battery-powered electronics is to extend the battery service life while meeting performance requirements. Designers of portable embedded systems therefore focus on power management methods to increased system performance while reducing operating power consumption. Static and dynamic power management policies, memory management schemes, bus encoding techniques, and hardware design tools are needed to meet these often-conflicting design requirements.

The present work was motivated by the need for a low-power device that can be used as an anti-theft alarm system for high-end bikes. The implemented system is a highly low-power object tracking system using GPS and GPRS technologies. The system calculates the position coordinates using GPS technology and sends them to a server through GSM technology. The system can also intelligently detect any motion on the bike and report suspicious activities on the bike. The system operates on very low power and is capable of remaining functional for weeks on a regular battery. The system can switch to an ultra-low power mode in order to extend the battery life for months.

This thesis discusses hardware and software techniques for power management system to design a low-power portable embedded system. Also, the designed power management system for this application is described. In the final chapter, development of the proposed device and implementation of the designed power management methods are defined.

Keywords: Anti-theft System, Low Power Electronics, Embedded Systems, Real-time Tracking, Intelligent Tracking, GPS Technology, GSM/GPRS Technology

Acknowledgements

I would like to express deepest gratitude to my supervisor Dr. Mehrdad Moallem for his full support, expert guidance, understanding and encouragement throughout my study and research. Without his incredible patience and timely wisdom and counsel, my thesis work would have been a frustrating and overwhelming pursuit. In addition, I express my appreciation to Dr. Jiacheng Wang and Dr. Kevin Oldknow for having served on my committee.

My special thanks to my friends and colleagues, especially Jaleh Mirhadi, Yaser M. Roshan, Ali Shagerdmootaab, Mohammad Mehdi Naseri, Rasoul Milasi, Younes Rashidi, Soroush Sefidkar, Farbod Faridi, and Samaneh Khakshour for their continued support and kind suggestions.

Finally, I would like to thank my parents for their unconditional love and support during the past years; I would not have been able to complete this thesis without their continuous love and encouragement.

Table of Contents

List of Tables

List of Figures

List of Acronyms

A

AGPS

Assist GPS 16

ALU

Arithmetic Logic Unit 37

AM

Active mode 43

ASCII

American Standard Code for Information Interchange 29

B

BER

Bit Error Rate 35

C

CMOS

Complementary metal-oxide semiconductor 50

CPU

Central processing unit 37

CR

Carriage Return 29

D

DCE

Data Communications Equipment 33

DGPS

Differential Global Positioning System 31

DOP

Dilution of Precision 31

DSP

Digital Signal Processor 26

E

EDGE

P

PRN

Pseudo Random Noise 13

PSRAM

Pseudostatic RAM 26

R

RAM

Random-access memory 36

RF

Radio Frequency 24

ROM

Read-only memory 36

RTC

Real-time Clock 26

S

SAE

System Architecture Evolution 23

SMS

Short Message Service 2

Short Messege Service 2

SPI

Serial Peripheral Interface 20

T

TI

Texas Instruments 41

TTFF

Time to First Fix 15

TTL

Transistor–transistor logic 55

U

U.S

United States 10

UART

V

Chapter 1.
Introduction

Theft of high-end bicycles is a large and growing problem. For instance, in the city of Toronto 3,139 bikes were stolen in 2011 with only 170 bikes reclaimed by the owner [1]. The number of stolen bikes in Vancouver reaches as high as 2,000 bikes per year [2]. Transportation Alternatives has estimated that upwards of a million bikes get stolen annually in the United States most of which are unreported [3]. The total value of stolen bicycles and their parts is approximated at $350 million a year in the U.S [4]. Similarly, 400,000 bicycles are stolen every year in France [5].

The effects on the bike manufacturing and retail industry are significant. The distribution channels for bikes include manufacturers, retailers and end consumers.

Since the safety of bikes in private and public places is a major concern, having an intelligent anti-theft system can prevent bike theft while also assisting in locating the bike after robbery. The system can help the police track the bike and locate the stolen bike conveniently.

1.1. Overview of the Anti-theft Bike System

The proposed activity involves development of a bike tracking system that senses the motion and coordinates of the bike using GPS and GSM technology, performing as an anti-theft system. The proposed system is an embedded system that continuously monitors the bike's status and location. Moreover, it reports the status and location automatically or on demand. The system is able to notify the user about suspicious activity on the bike or track the bike in real time.

To develop this system a microcontroller unit is interfaced serially to a GSM Modem, GPS receiver, and an accelerometer, as shown in Figure 1.1. The system is battery powered and switching to particular operation modes will extend battery life, significantly.

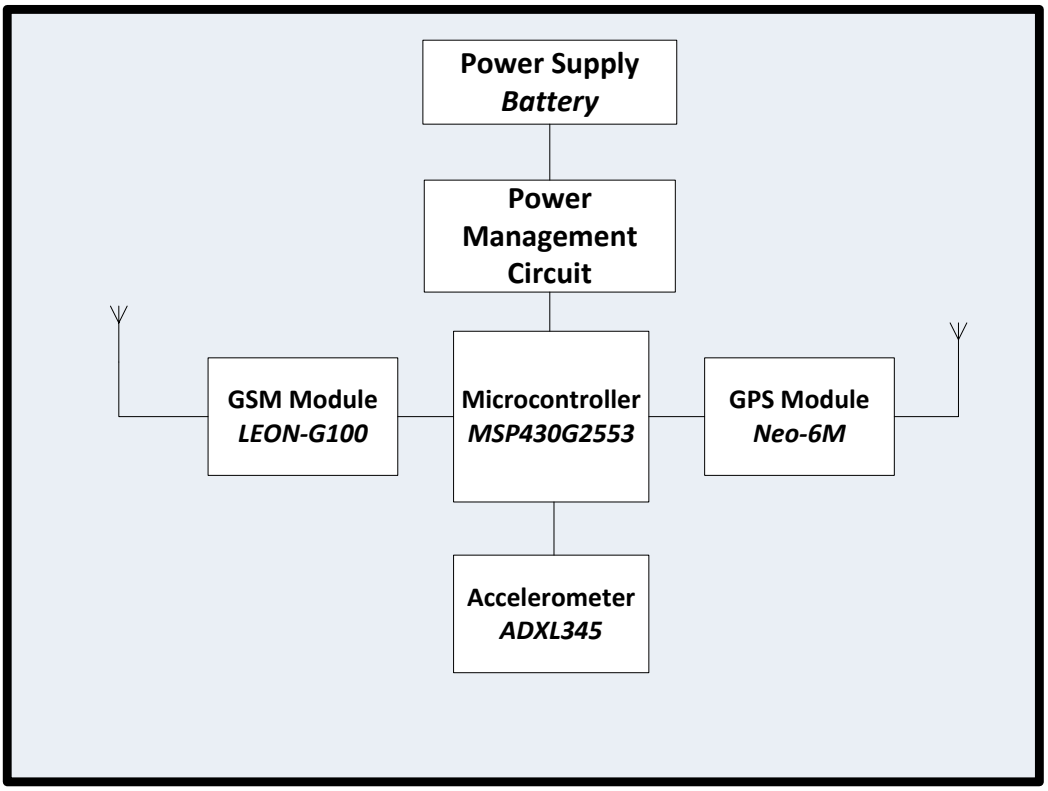


Figure 1-1 System hardware overview

A web-based user interface shows the location on a map. A mobile application is designed to control the features of the system, including tracking the location. The same information could also be sent through SMS to the user's mobile device at the other end from where the position of the bike is demanded. When the request is sent by the user to the system, the system automatically sends a reply with the latitude and longitude of the bike in real time. Figure 1-2, shows how the system interacts with users.

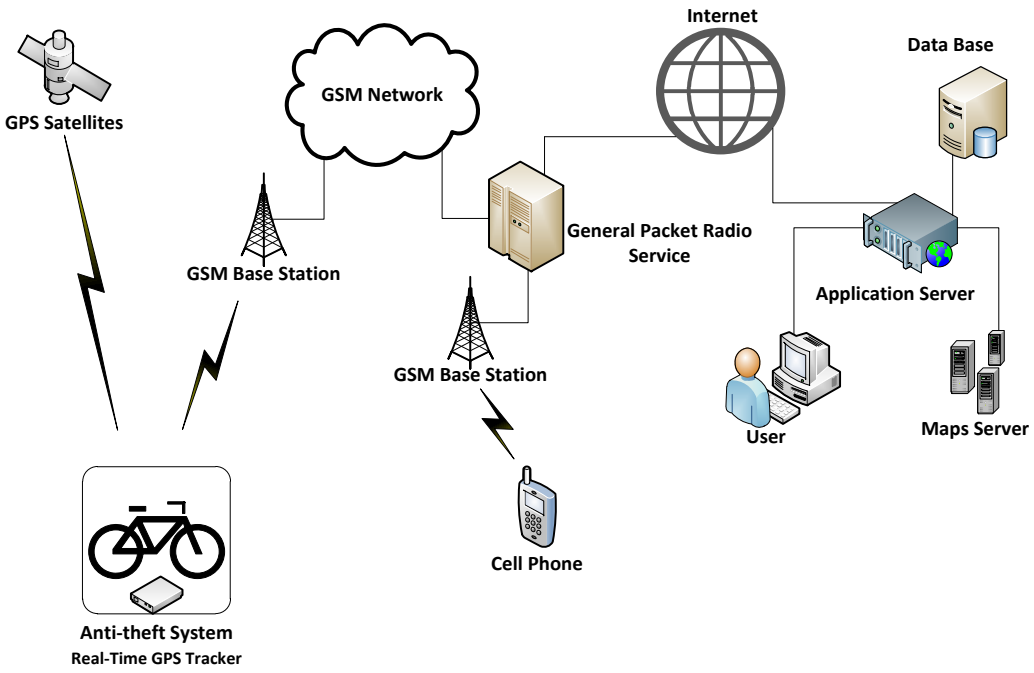


Figure 1-2 System overview

1.2. Limitations and Solutions

By growing the popularity of portable electronic systems, the demand for devices with extensive battery life has become a new challenge for designers. A portable electronic system requires accomplishing minimum power dissipation simultaneously, with acceptable performance. The portable tracking system involves transmission of data, which indeed extends power consumption when compared to other portable systems. The limitation of size and capacity of battery and power-hungry components in this type of system increase complexity. Furthermore, the GPS and GSM technology are considered as slow and high power consuming components. As a matter of fact, the total energy consumed by the GPS and GSM transceivers makes up a large fraction of energy stored in the system's battery.

Power management system is a powerful tool in the hand of engineers used to optimize systems for stamina and power. As a matter of fact, a system without any power management can drain a battery without providing any performance. Choosing the right method of power management on both layers of software and hardware

reduces power consumption significantly. A multiple-component embedded system necessitates advance rules to control and adjust power consumption, in order to efficiently use the available energy.

Position tracking using the GPS technology requires long initialization time, that is highly sensitive to the environment. The long initialization time increases the total consumption of energy and impacts the battery life. Also, the GPS technology is usually non-functional in urban areas where towers may jam satellite signals [6]. These disadvantages make position tracking a major issue in battery-powered systems. Not many of the studies carried out in this field have focused on extending battery life on the GPS tracking devices.

Up to this date, different types of anti-theft bike systems have been introduced to market such as Mobiloc®[1], BikeSpike[2], SPYBIKE[3], and Helios[4]. Although different installation or theft detection methods are used, position tracking is a common feature in all of them. Additionally, other products such as personal trackers carry the same functionality. CHILDTRAC[5], Personal GPS Tracker[6], and Vismo Personal Tracker[7] are examples for personal trackers. As battery life is an important and at the same time weak point of these products, most of them do not provide clear information about their battery life.

BikeSpike claims depending on ride usage, their system may remain functional up to 3-4 weeks, while Helios, SPYBIKE and Moblioc do not provide any information about battery life. CHILDTRAC remains functional from 24 to 72 hours, the Personal GPS Tracker's battery lasts up to 80 hours and finally Vismo Personal Tracker's provides a 10 day battery life. The examples of battery life variation sheds light on the

1 http://www.mobilocgps.com/
2 http://bikespike.com/
3 http://www.integratedtrackers.com/
4 http://ridehelios.com/
5 http://www.childtrac.ca/
6 http://personaltrackerdirect.com/
7 http://www.vismo.com/products/personal-tracker

lack of appropriate power management system used in the portable position tracking devices.

This thesis comprises development of a low-power anti-theft system for bikes with a focus on power optimization for position tracking systems. Moreover, methods of GPS tracking are discussed and solutions to reduce the tracking time are provided.

1.3. Disposition

- **Chapter 1 Introduction**

  The following chapter will give a brief introduction to the Anti-theft Bike System. The criteria and contents of the project will also be introduced.

- **Chapter 2 Accelerometer**

  The method of detecting acceleration force will be discussed along with an introduction of ADXL345.

- **Chapter 3 Positioning Methods**

  The positioning methods chapter will give a survey of different possible positioning methods. The mentioned methods are GPS, AGPS, and Mobile Positioning Systems.

- **Chapter 4 Mobile Communications**

  This chapter will describe different communication links that might be used to establish a connection with the field application. The communication links described here are the ones which are most suitable for this project.

- **Chapter 5 Protocol and Command Language**

  This chapter will discuss the NMEA protocol followed by a few examples of different messages. Also, the AT command for GSM modems with a few examples will be discussed.

- **Chapter 6 Microcontroller**

  The common feature of microcontrollers will be discussed and the TI MSP430 will be introduced.

- **Chapter 7 Optimizing Embedded System Power Consumption for Low-Power Designs**

  This chapter will discuss methods of power management in order to achieve a low-power design.

- **Chapter 8 Design and Development**

  The design of prototype and implementation of the power managements methods will be discussed. Each component will be studied individually in order to reduce power consumption.

Chapter 2.
Accelerometer

An accelerometer is an electromechanical device that can measure acceleration forces. These forces may be static, like the constant force of gravity pulling at your feet, or they can be dynamic, caused by moving or vibrating the accelerometer. Sensing the amount of dynamic acceleration can be utilized to analyze movement of an object. During the rest of this section various accelerometer technologies are described and the one which is chosen in this project is studied in details.

2.1. Technologies

Various technologies are used in the construction of accelerometers. One of them is using the piezoelectric effect. Piezoelectric effect can be described as the microscopic crystal structures generate voltage when they are stressed by accelerative force. Another technology uses the change in capacitance for sensing acceleration. In this method, the capacitance between two microstructures next to each other changes if a force moves one of the structures. This phenomenon is used to detect the acceleration force in some accelerometers by converting the capacitance to voltage. There are other methods to measure acceleration as indicated in Table 1.

Sensor Category	Key Technologies
Capacitive	**The metal beam or micromachined feature produces capacitance; the change in capacitance is related to acceleration**
Piezoelectric	**The piezoelectric crystal is mounted to the mass –the voltage output is converted into acceleration**
Piezoresistive	**The beam or micromachined feature whose resistance changes with acceleration**
Hall Effect	**Motion is converted into an electrical signal by sensing of changing magnetic fields**
Magnetoresistive	**The material's resistivity changes in the presence of a magnetic field**
Heat Transfer	**The location of heated mass is tracked during acceleration by sensing temperature**

Table 2-1 Different Types of Accelerometer

Furthermore, there are properties by which accelerometers vary from each other:

- **Number of Axes:** For most activities, two axes are sufficient. Nonetheless, if 3D positioning is required, a 3rd axis accelerometer or two 2 axis ones mounted at right angles are needed.
- **Maximum Swing:** In the case of measuring tilt using earth's gravity, a ±1.5g accelerometer would be adequate. If the accelerometer is used to measure the motion of a car, plane, or robot, then ±2g should give enough flexibility. For a project that experiences very sudden starts or stops, an accelerometer which can deal with ±5g or more may be required [7].
- **Sensitivity:** As a rule, the more sensitivity, the better the sensor. This implies that for a given change in acceleration, there will be a larger change in signal and therefore will be more accuracy.
- **Bandwidth:** For slow-moving tilt-sensing applications, a bandwidth of 50Hz will presumably suffice. For measuring vibration or to control a fast moving machine, a bandwidth of several hundred Hz is required.

2.2. ADXL345

The ADXL345 is a complete 3-axis acceleration measurement system with a selectable measurement range of $\pm 2\,g, \pm 4\,g, \pm 8\,g, or\ \pm 16\,g$. It measures dynamic

acceleration resulting from both motion or shock and static acceleration such as gravity; which allows the device to be used as a tilt sensor. The sensor is a polysilicon surface-micromachined structure built on top of a silicon wafer. Polysilicon springs suspend the structure over the surface of the wafer and provide resistance against forces due to applied acceleration.

Deflection of the structure is measured using differential capacitors that consist of independent fixed plates, as well as plates attached to the moving mass. Acceleration deflects the proof mass and unbalances the differential capacitor, resulting in a sensor output whose amplitude is proportional to the acceleration. Phase-sensitive demodulation is used to determine the magnitude and polarity of the acceleration.

The ADXL345 is well-suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as the dynamic acceleration resulting from motion or shock. Its high resolution ($3.9\ \frac{mg}{LSB}$) enables it to measure inclination changes less than 1.0°.

Low-power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipations. The voltage range of ADXL345 ($2.0\ V\ to\ 3.6\ V$) and its ultra-low current consumption ($23\mu A$) in measurement mode and 0.1μA in standby mode ($V_s = 2.5V$) puts ADXL345 into the set of highly low-power accelerometers.

Chapter 3.
Positioning Methods

3.1. GPS

The development of the U.S military's satellite navigation system began in the early 1970s. The essential components of GPS are the 24 Navstar satellites built by Rockwell International with a weight of 1900 pounds each. The first operational GPS satellite was launched on February 22 in 1978, and the system reached full 24 satellite capability in 1993. The satellite type used in the Global Positioning System today is also called NAVSTAR, which stands for "Navigation Satellite Time and Ranging Global Positioning System" [8]. As of December 2012, there are 32 satellites in the GPS constellation. The additional satellites improve the precision of GPS receiver calculations by providing redundant measurements [9].

3.1.1. GPS Satellites

Today the Global Positioning System consists of 31 satellites orbiting the earth at a height of 20,200 kilometres, as shown in Figure 3-1. These satellites are placed in six different orbital planes, with an inclination of 55 degrees with respect to the equatorial plane. Each satellite orbits the earth every 12 hours in such a formation that every point on the planet will always be in radio contact with at least four satellites. Thus the satellites are moving at a speed of approximately 10,500 $\frac{\mathrm{Km}}{\mathrm{h}}$.

The solar panels of the satellite generate about 800 Watts. The transmit power for the NAVSTAR satellite is 50 Watts or less. Each satellite has an extremely accurate atomic clock that is synchronized to all of the other satellite clocks, as well as to the

ground control stations. Replacement satellites are constantly launched into orbit, as a GPS satellite's life time is from 7.5 to 10 years [10].

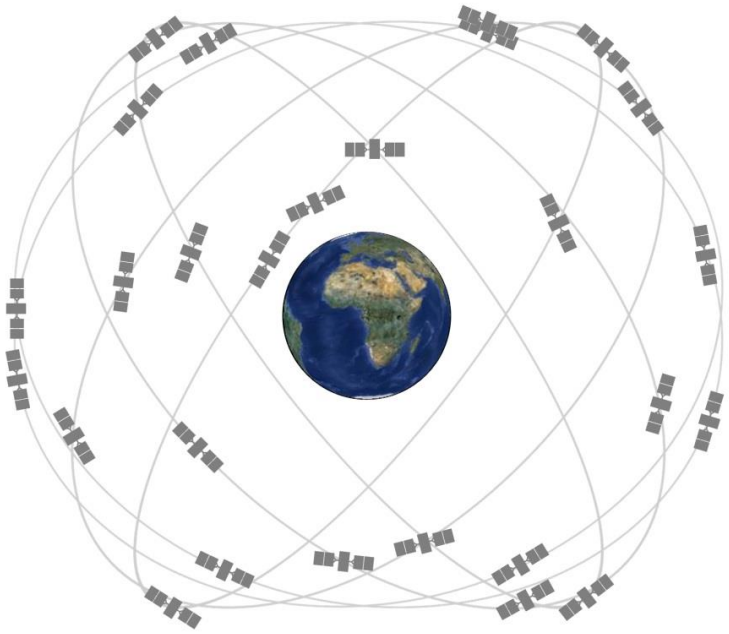


Figure 3-1 Presentation of Satellites Orbits[8]

3.1.2. Satellite Signal

The GPS satellites produce the fundamental L-band frequency of 10.23 MHz. From this fundamental frequency, the L_1 and L_2 signals are created. Multiplying the fundamental frequency by 154 and 120 respectively gives:

$$L_1\ MHz = 1575.42 \qquad (3\text{-}1)$$

$$L_2\ MHz = 2\ 1227.60 \qquad (3\text{-}2)$$

[8] http://www.gps.gov/systems/gps/space/

The satellites transmit a navigation message that essentially is the satellite clock, system time, its orbital elements, various correction data, and health status. The total messages are called almanacs; which contains 1500 bits (with a transmission rate for the message of 50 bits/s) and are divided into five sub frames. The first sub frame contains the GPS week number, a prediction of the user range accuracy, the satellite health, clock correction, the age of the data, and an estimation of the signal group delay. The second and third sub frames contain orbital data (ephemeris data parameters) for the transmitting. The contents of the fourth and fifth sub frames are changed in every message. It can, for example, contain information about the ionosphere (this will be mentioned later), various flags, and almanac data for all 32 satellites in orbit. Each satellite therefore broadcasts the fourth and fifth sub frames. Almanac data of all other satellites in orbit are therefore obtained once the first satellite has been acquired. The receiver must extract the message from each satellite for 18 to 30 seconds to obtain an accurate satellite location from the transmitted message. Thus, in order to collect all of the transmission almanacs, the receiver must demodulate for almost 13 minutes [8] [10].

3.1.3. Technical overview

A short and simplified way to determine a given position:

- The signal is transmitted from a satellite, and contains the time of departure.
- The receiver receives the signal, and the time of arrival is registered.
- The radio waves (the signal) travel at the speed of light ($3*10^8$ m/s), and the distance (D_S) to the satellite can be calculated by following formula [10] [11]:

$$D_s = c.T \qquad (3\text{-}3)$$

$c = \ speed\ of\ light\ \left(\frac{m}{s}\right)$

$T = travel\ time\ for\ signal$

With this information, it is possible to create a hemisphere with its center at the satellite and with the radius D_s. The receiver will be located at the surface of the hemisphere. It is possible to get a 2D position if this is repeated with three satellites; with the use of four satellites, a 3D position is possible.

3.1.3.1 Pseudorange

The fundamental measurement in the Global Positioning System is pseudorange. The receiver receives the pseudorandom noise (PRN) code from a satellite and identifies the satellite, then generates a replica code. The phase by which the replica code must be shifted in the receiver to maintain maximum correlation with the satellite code, multiplied by the speed of light, is approximately equal to the satellite range (Figure 3-2). It is called the pseudorange because the measurement must be corrected by a variety of factors in order to obtain the true range. A simple model for calculating the pseudorange, R, is [10]:

$$R = \theta + \Delta\theta = \theta + c\delta \qquad (3\text{-}4)$$

$$\theta = ||\theta_s - \theta_r|| \qquad (3\text{-}5)$$

$R = pseudorange$

$c = speed\ of\ light$

$\delta = time\ difference$

$\theta = distance\ from\ the\ satelite\ to\ the\ receiver\ (see\ Figure\ 2.1)$

$\theta_r = distance\ from\ the\ geocenter\ to\ the\ receiver\ (see\ Figure\ 2.1)$

$\theta_s = distance\ from\ the\ geocenter\ to\ the\ satelite\ (see\ Figure\ 2.1)$

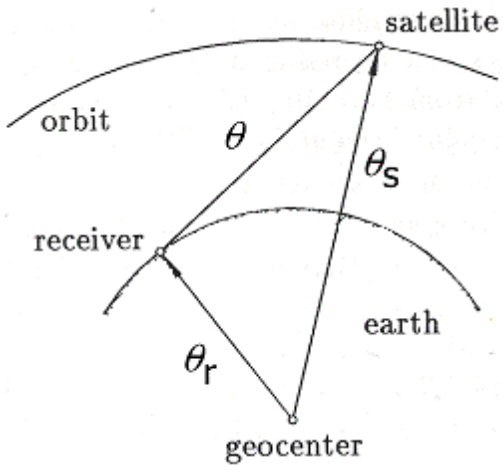


Figure 3-2 Pseudorange Calculation

3.1.3.2 Triangulation

GPS receivers on Earth use GPS satellites as reference points for triangulating a precise position on Earth. Two satellite measurements determine an intersection of two spheres [10]. This intersection will be a circle, and the receiver will be located somewhere on this circle as in Figure 3-3.

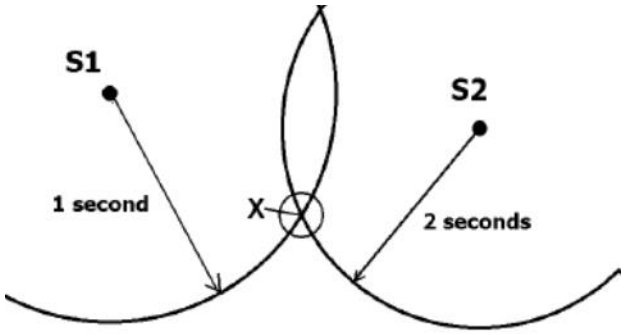


Figure 3-3 The receiver will be located somewhere inside of the black circle created by the intersection between the two spheres

A third satellite's sphere cuts through the circle created by the first two spheres, and so we get two intersection points as shown in Figure 3-4. The receiver is located on one of these two points. These two points are actually enough to determine the position (2D position) of the receiver, because one of the points could not reasonably be

considered as the receiver. With the use of a fourth satellite, one has no problem determining the exact point of the receiver (3D position) [10].

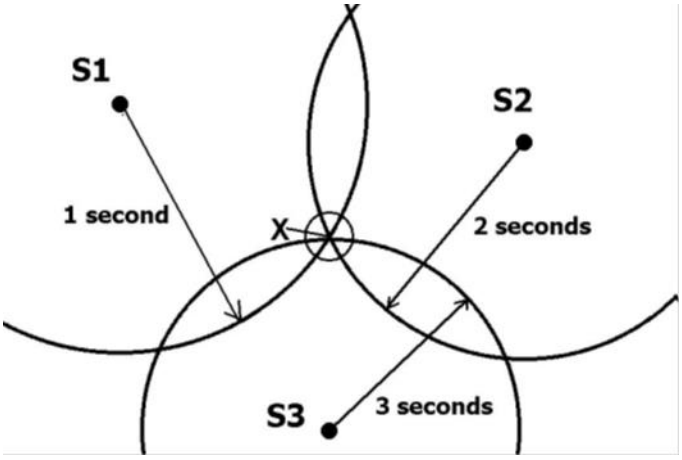


Figure 3-4 A third satellite's sphere intersects with the verge of the circle created by the other two satellites at two points. The receiver is located at one of these points

3.1.4. Technical Limitations

Briefly, the two main issues with GPS system are as follows:

- Time To First Fix
- Poor Signal Condition

3.1.4.1 Time to First Fix

TTFF is the time that the GPS system requires in order to determine its current position. As explained previously, each satellite broadcasts the ephemeris data for 30 seconds periodically. If the GPS system loses the tracking of the data part, it will have to start again in the next 30 second cycle [11].

There are 3 conditions where a GPS system begins to acquire a lock on the position. They are titled as follows [11] [12]:

1. **Factory or Cold Start:** The receiver has no information regarding almanacs data and it can take about 12.5 minutes for full almanacs to be downloaded.

2. **Warm Start:** The receiver has some knowledge of the almanac, but there is no ephemeris data available. Most of the modern GPS systems achieve a position fix in less than 1 minute when the condition is that of a warm start.
3. **Hot Start:** The receiver has the updated the almanacs and ephemeris information and only requires obtaining the time information. Starting on this condition can take up to 20 seconds.

In most conditions, TTFF and positions tracking are time-consuming. It plays a critical role in mobile devices where battery life becomes a big challenge and creates unwanted obstacles. On the other hand dependent on the condition, GPS systems are required to observe satellite signals for up to 13 minutes. Mobile devices' batteries may not be able to handle this activity [13]. In order to overcome this problem, it is required to either increase the capacity of the batteries, which may not be possible, or reduce TTFF as much as possible. Assisted GPS technology is designed to be used in conjunction with GPS to reduce TTFF, and will be discussed in section 3.2.

3.1.4.2 Poor Signal Condition

The GPS receiver has to identify at least three satellites, and receive their orbital position, in order to calculate a position fix. In the event that satellite signals are not available, such as indoor urban environments, where towers have blocked the signals, or in the presence of GPS signal jammers, a GPS receiver cannot receive the signals; as a result, positioning is not possible [14].

3.2. Assisted GPS (AGPS)

Assisted GPS improves the performance of the GPS receiver unit by providing it with data that otherwise would have had to be downloaded from the satellites. Using this technique, the time to first fix can be shortened. Conventional GPS receivers need to search the entire frequency/code space to get the first fix (cold start). Thus, the position can be determined faster with AGPS technology.

AGPS accelerates delivers satellite data such as ephemeris, almanac, accurate time, and satellite status to the GPS receiver over the internet (Figure 3-5). This will shorten the TTFF from minutes to seconds, and reduce the search space size. By using AGPS receiver, it is also possible to detect and demodulate weaker signals than conventional GPS receivers can [11] [15].

The satellite data can be downloaded in two ways:

1. **Offline Assist:** The data can be downloaded from the internet at a convenient time and stored in the memory of application processor. This way, the system requires no connectivity during positioning and can use the data at start up. The valid data is available for 1 to 14 days. The size of these files increase with length of period [15].
2. **Online Assist:** The data has to be downloaded each time positioning is required or at start-up. An internet connection is required when using this service. The data of currently visible satellites can be downloaded from a server and the data validity period is between 2 to 4 hours [15].

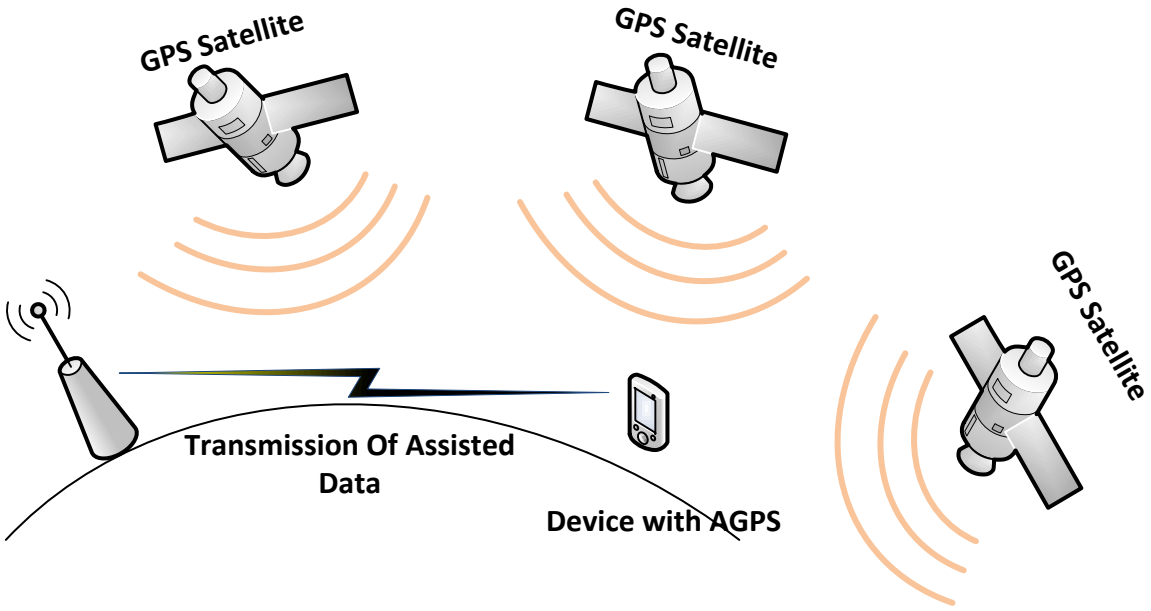


Figure 3-5 Overview of a AGPS System

The online assist requires less time for downloading the data and acquisition performance, but its connectivity requirement during the positioning makes the service unusable in unavailable cellular network conditions.

3.3. Mobile Position System

The position of a GSM mobile phone can be determined using information from the GSM radio network. The accuracy with which the location can be determined depends on the network cell towers, as well as the status of the phone. The best accuracy will be obtained in cities and other places were the sizes of the number of cells are small. Cells covering the countryside are larger and will thus have poorer accuracy. To put this into context, it is not unrealistic to have an accuracy of 10 metres inside cities and 1.5 kilometres in the countryside [14].

3.3.1. CellLocate Technology

This technology uses surrounding mobile network information in conjunction with GPS positioning data to improve positioning. CellLocate technology requires the observer's information regarding the best visible cell station at any specific location. The combination of this information is sent to a server. Then the server can estimate the position on the basis of previous observations from other modules. Figure 3-6 shows the process. The estimated positions are then sent back to the module. Obviously, this system is available in any given position with wireless data network coverage in order to send and receive information [16].

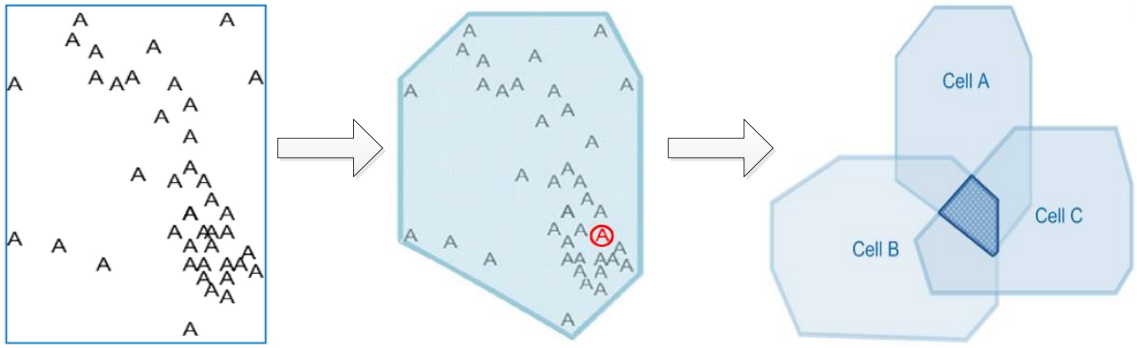


Figure 3-6 CellLocate Technology Positioning Steps

3.3.2. Benefits of CellLocate Technology

Using CellLocate in conjunction with GPS and AGPS has the following benefits:

1. In poor signal condition positioning, it is possible to assist the GPS by providing an approximate fix. In addition, if the GPS satellites are totally blocked, it is possible to provide a fix within a few meters of error. The rate of error is dependent upon previous observations.

2. CellLocate uses a self-learning system in order to improve over time by increasing the density of cellular observations.

3.4. u-blox NEO-6

The NEO-6 module (Figure 3-7) series is a family of stand-alone GPS receivers with a high performance u-blox positioning engine. The receiver offers various connection options, is highly adaptable, and has a low cost. Its compact architecture and low power features makes the NEO-6 module the perfect choice for battery worked gadgets and devices.

Figure 3-7 u-blox NEO-6

The u-blox 6M has 50 positioning channels and engines to provides an acceptable TTFF and accuracy for tracking. The u-blox features A-GPS, CellLocate Technology and UART/SPI/USB which provides acceptable positioning functionalities. Table 3-1 demonstrates NEO-6M's performance [17].

GPS performance

Parameter	Specification			
Receiver type	50 Channels GPS L1 frequency, C/A Code SBAS: WAAS, EGNOS, MSAS			
Time-To-First-Fix[1]		NEO-6G/Q/T	NEO-6M/V	NEO-6P
	Cold Start[2]	26 s	27 s	32 s
	Warm Start[2]	26 s	27 s	32 s
	Hot Start[2]	1 s	1 s	1 s
	Aided Starts[3]	1 s	<3 s	<3 s
Sensitivity[4]		NEO-6G/Q/T	NEO-6M/V	NEO-6P
	Tracking & Navigation	-162 dBm	-161 dBm	-160 dBm
	Reacquisition[5]	-160 dBm	-160 dBm	-160 dBm
	Cold Start (without aiding)	-148 dBm	-147 dBm	-146 dBm
	Hot Start	-157 dBm	-156 dBm	-155 dBm
Maximum Navigation update rate		NEO-6G/Q/M/T	NEO-6P/V	
		5Hz	1 Hz	
Horizontal position accuracy[6]	GPS	2.5 m		
	SBAS	2.0 m		
	SBAS + PPP[7]	< 1 m (2D, R50)[8)]		
	SBAS + PPP[7]	< 2 m (3D, R50)[8]		
Configurable Timepulse frequency range		NEO-6G/Q/M/P/V	NEO-6T	
		0.25 Hz to 1 kHz	0.25 Hz to 10 MHz	
Accuracy for Timepulse signal	RMS	30 ns		
	99%	<60 ns		
	Granularity	21 ns		
	Compensated[9]	15 ns		
Velocity accuracy[6]		0.1m/s		
Heading accuracy[6]		0.5 degrees		
Operational Limits	Dynamics	≤ 4 g		
	Altitude[10]	50,000 m		
	Velocity[10]	500 m/s		

Table 3-1 NEO-6M Performance

Chapter 4.
Mobile Communications

4.1. Introduction

The Global System for Mobile Communications (GSM) family of technologies is the most widely used wireless technology in the world, with more than 6 billion users in more than 219 countries [18].

GSM has begun to be the fastest growing wireless technology in North America. Over the past few years, GSM's market share has grown rapidly. By 2006, the number of GSM subscribers passed 2 billion, and that number reached more than 6 billion in 2012. GSM's share of the market is more than 90 percent [18].

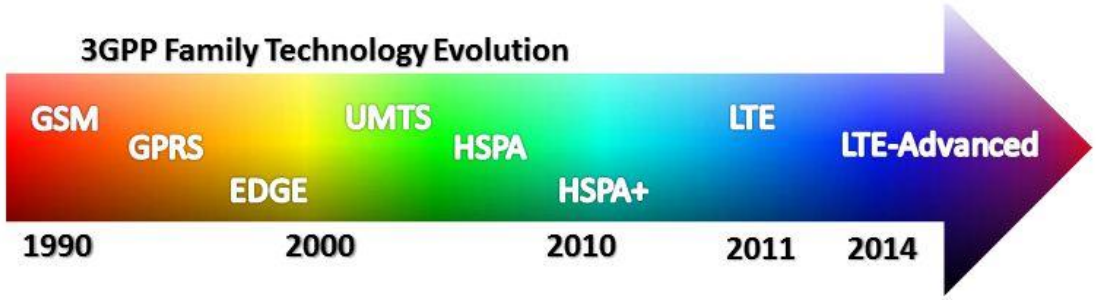


Figure 4-1 GSM Technology Evolution[9]

GSM is the legacy network in the evolution to today's technologies. Figure 4-1 represents this evolution over the past 24 years. GSM has a straightforward, cost-effective migration path to 3G through GPRS, EDGE, and UMTS-HSPA, as well as

[9] http://www.4gamericas.org/index.cfm?fuseaction=page&sectionid=242

beyond 3G via the HSPA Evolution (HSPA+), LTE, and System Architecture Evolution (SAE) initiatives. Each step in the GSM-based movement path influences the network substructure deployed for the previous steps, and is backwards compatible. For example, a UMTS phone can provide voice and data service when connected to a GSM network.

4.2. GPRS

GPRS stands for "General Packet Radio Service", and the technique is packed-switched instead circuit-switched like it is for GSM. This means that the data stream is divided into small packages that are routed through the network. To make this possible, each package contains a destination address. This type of connection between a sender and a receiver is referred to as being "connectionless", unlike the circuit-switched system, where it is referred to as being "dedicated". With a connectionless connection, the user can be “connected” all the time without taking up any bandwidth. Bandwidth is only used when the user sends or receives data. With a dedicated connection, the user uses bandwidth even if not sending or receiving any data. When using a dedicated connection, the user pays for the whole connection time even if no data are sent or received, whereas with GPRS, the user only pays for the amount of data transferred [19] [20].

The transfer rate is higher with GPRS than with GSM. Today’s systems use a maximum transfer rate of 36.2 Kbit/s (9.05 Kbit/s * 4) and 53.6 Kbit/s (13.4 Kbit/s * 4) [21].

With GPRS users do not need a subscription at an ISP for Internet connection. The Internet connection is “built in” with GPRS, and the user gets the Internet connection through the provider for the GPRS service.

4.3. EDGE

EDGE stands for "Enhanced Data rates for GSM Evolution" and the objective of this new technology is to increase data transmission rates and spectrum efficiency, as well as to facilitate new applications and increased capacity for mobile use. This enables data to be sent over a GSM system at speeds up to 473 kbps for 8 timeslots, but this is typically limited to 135 kbps in order to conserve spectrum resources. EDGE is an additional upgrade to the existing GPRS networks by offering a new physical layer, and so EDGE technology is included nearly in all new GSM devices because of the minimal incremental cost of including EDGE in the GSM network [18] [22].

4.4. LEON-G100

LEON-G100 (Figure 4-2) is a quad-band GSM/GPRS wireless module with comprehensive features and the integration of data and voice modules.

Figure 4-2 LEON-G100

4.4.1. Architecture

Figure 4-3 shows the architecture of LEON G100 in block diagrams. LEON G100 consists of three main functional blocks:

- RF

- Baseband
- Power Management

Each of these blocks will be described in detail in the following.

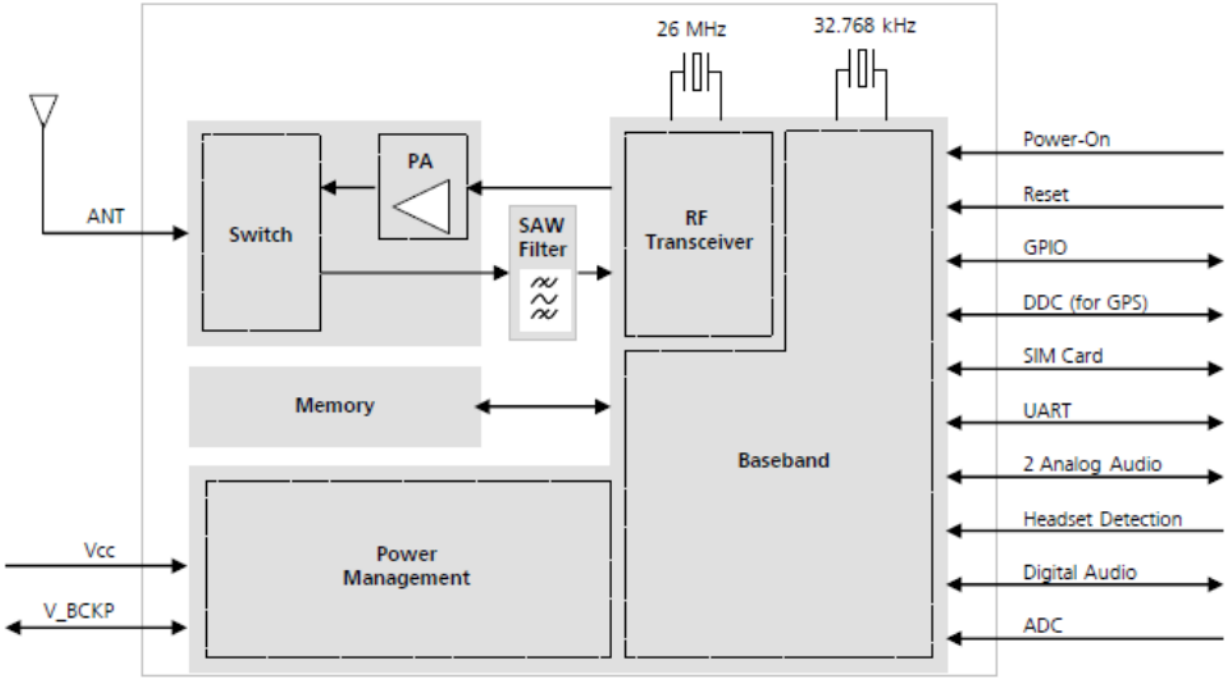


Figure 4-3 LEON G100 Block Diagram

4.4.1.1 RF Block

The RF block is composed of the following main elements:

- RF transceiver (integrated in the GSM/GPRS single chip) performing modulation, up-conversion of the baseband I/Q signals, down-conversion, and demodulation of the RF received signals. The RF transceiver includes:
    - Constant gain direct conversion receiver with integrated LNAs;
    - Highly linear RF quadrature demodulator;
    - Digital Sigma-Delta transmitter modulator;
    - Fractional-N Sigma-Delta RF synthesizer;
    - 3.8 GHz VCO;
    - Digital controlled crystal oscillator.

- Transmit module, which amplifies the signals modulated by the RF transceiver and connects the single antenna input/output pin of the module to the suitable RX/TX path, via its integrated parts:
    - Power amplifier;
    - Antenna switch.
- RX diplexer SAW (band pass) filters
- 26 MHz crystal, connected to the digital controlled crystal oscillator to perform the clock reference in active or connected mode

4.4.1.2 Baseband

The Baseband block is composed of the following main elements:

- Baseband integrated in the GSM/GPRS single chip, including:
    - Microprocessor;
    - DSP (for GSM/GPRS Layer 1 and audio processing);
    - Peripheral blocks (for parallel control of the digital interfaces);
    - Audio analog front-end.
- Memory system in a multi-chip package integrating two devices:
    - NOR flash non-volatile memory;
    - PSRAM volatile memory.
- 32.768 kHz crystal, connected to the oscillator of the RTC to perform the clock reference in idle or power-off mode

4.4.1.3 Power Management

The Power Management block is composed of the following main elements:

- Voltage regulators integrated in the GSM/GPRS single chip for a direct connection to the battery
- Charging control circuitry

4.4.1.4 Operation Modes

The Operation consists of the following modes:

- **Power-Off Mode:** The VCC supply is within normal operating range, but the microprocessor is not operating.
- **Idle-Mode:** The microprocessor runs with 32 kHz as reference oscillator. The module does not accept data signals from an external device.

- **Active-Mode:** The microprocessor runs with 26 MHz as its reference oscillator. The module is ready to accept data signals from an external device.
- **Connected-Mode:** Voice or data calls are enabled. The microprocessor runs with 26 MHz as its reference oscillator. The module is ready to accept data signals from an external device.

Chapter 5.
Protocol and Command Language

5.1. NMEA Protocol

NMEA 0183 v4.00 is an industry standard from the National Marine Electronics Association which defines an electrical interface and a data protocol for communications between marine instrumentation. The standard defines electrical signal requirements, data transmission protocol, timing, and specific sentence formats. The National Marine Electronics Association is an association of manufacturers, dealers, educational institutions, distributors, and others that are interested in peripheral marine electronics communication between marine instrumentation. Today, the standard is also used together with non-marine instrumentation: for example, to navigate on roads with a GPS receiver.

5.1.1. NMEA Electrical Interface

NMEA 0183 devices are designed as either listeners or talkers (though some devices can be both). NMEA allows for a single talker and several listeners on one circuit. The devices use an asynchronous serial interface to communicate. Most of the GPS modules today use a serial bus to communicate. However, depending on the application, there are GPS modules that benefit SPI or I2C bus.

5.1.2. NMEA General Sentence Format

All data transmitted with NMEA is done so in the form of sentences. Only printable ASCII (ASCII –American Standard Code for Information Interchange) characters are allowed, with the exception of CR (Carriage return) and LF (Line feed).

The sentences always start with a $ sign and end with <CR><LF>. A sentence may contain up to 80 characters excluding $, CR and LF. There are three basic kinds of sentences: proprietary-, talker-, and query sentences. The structure of a NMEA protocol message is shown in Figure 5-1.

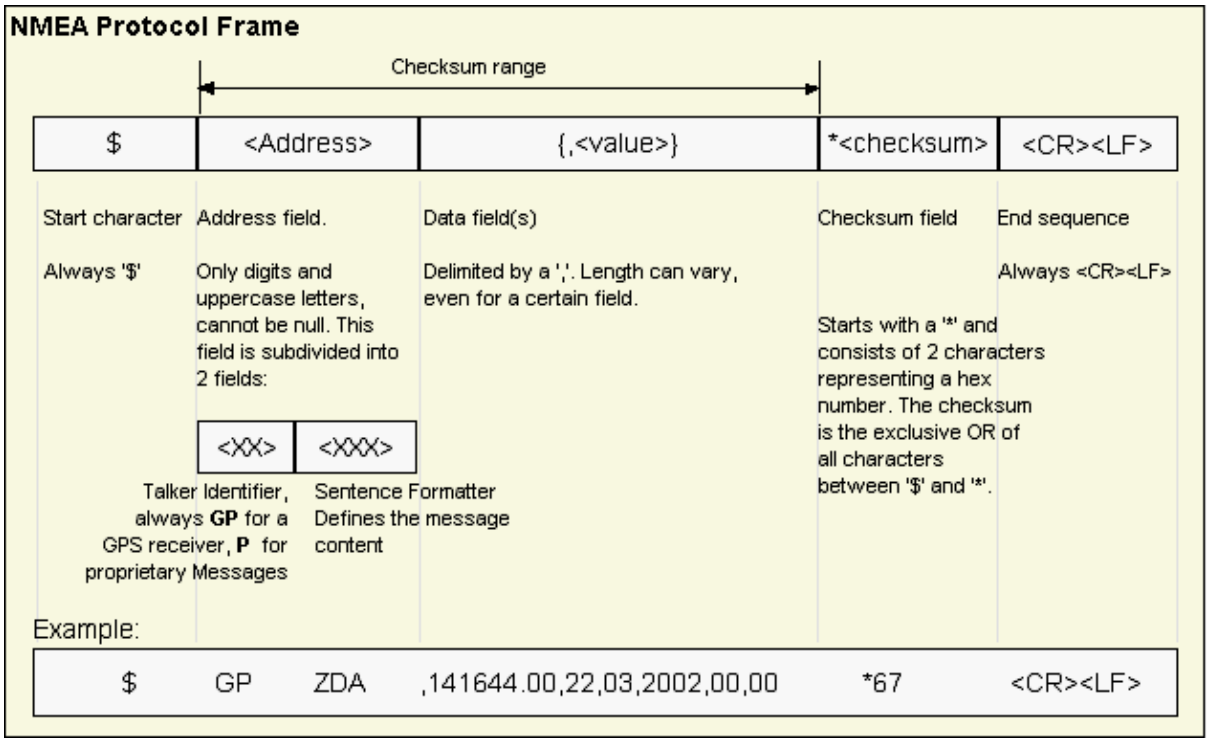


Figure 5-1 NMEA Protocol Message Format

5.1.3. Talker Sentence

A talker sentence starts with a $ followed by two characters (tt) which are the talker identifier. The next three characters are the sentence identifier (sss), followed by a number of data fields (d1, d2… dn) separated by commas. If data for a field are not available, the commas are still sent, but with no spaces between them. An optional checksum follows after the data fields, and then the sentence is ended with CR and LF. The general format fora talker sentence is:

*$ttsss,d1,d2, ...,dn,*hh,<CR><LF>*

5.1.4. Proprietary Sentences

The NMEA standard allows manufactures to define their own sentences. These sentences are called proprietary sentences, and start with $P followed by a three letter manufacturer ID. This is followed by the data that the manufacture decides to include, following the general format of the standard sentences.

5.1.5. Query Sentence

A query sentence is a sentence used by the listener device to request a specific sentence from a talker device. The first two characters after the $ are the talker identifier (tt) of the requester; the next two characters are the talker identifier (ll) of the listener. The fifth character is always a Q, which defines that the message is a query. The next field contains three characters which are the sentence identifier (sss) for the requested sentence. The general format for a query sentence is:

$ttllQ,sss,<CR><LF>

5.1.6. Talker Identifiers

Talker identifiers give information about who the talker is, and are specified in the first one or two characters in the sentence, i.e. $--. One example of talker identifiers is:

$GPGLL Global Positioning System (GPS)

5.1.7. GGA Global Positioning System Fix Data

The GGA message is a base message that is supported by all GPS receivers. The information inside of this message contains the time, position, fixation related data, etc., for a GPS receiver. The position is given in the datum format WGS84, and the time is given in Greenwich Time. The fixation data are an indication of whether there is a fixation available or not. It also indicates if a GPS or a DGPS fixation is available. The complete contents and structure of the message are shown below:

*$--GGA,hhmmss.ss,llll.ll,a,yyyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh*

	1	2	3	4	5	6	7	8	9	10	11	12	13	12	13
$--GGA	hhmmss.ss	llll.ll	a	yyyyy.yy	a	x	xx	x.x	x.x	M	x.x	M	X.X	xxxx	hh

1. Time (UTC)
2. Latitude
3. N or S (North or South) Indicator
4. Longitude
5. E or W (East or West) Indicator
6. GPS Quality Indicator,
    - fix not available,
    - GPS fix,
    - Differential GPS fix
7. Number of satellites in use, 00 – 12
8. Horizontal Dilution of Precision (DOP)
9. Antenna Altitude above/below mean-sea-level (geoid)
10. Units of antenna altitude, metres
11. Geoidal separation, the difference between the WGS-84 earth ellipsoid and mean-sea-level geoid), "-" means mean-sea-level below ellipsoid
12. Units of geoidal separation, metres
13. Age of differential GPS data, time in seconds since last SC104 type 1 or 9 update, null field when DGPS is not used
14. Differential reference station ID, 0000-1023
15. Checksum

5.1.8. GSV Satellites in View

The GSV message gives the information of all of the GPS satellites in view. The information inside of this message contains data that might be used when analyzing and

debugging a system. It can, for example, be handy if there are no fixations available. The provided data are satellite identification, elevation- and azimuth-angle, and signal-to-noise ratio for each satellite in view. This information can give valuable information as to why there is no fixation available. The complete contents and structure of the message are shown below:

*$--GLL,x,x,x,x,x,x,,...*hh*

	1	2	3	4	5	6
$--GLL	X	X	X	X	X	x

1. Latitude
2. N or S (North or South)
3. Longitude
4. E or W (East or West) Indicator
5. Time (UTC)
6. Status (Data valid or not valid)
7. Checksum

5.2. AT Commands

AT Commands is a series of commands to control modems such as a GSM modem to do specified functions.

5.2.1. Command Description

The AT commands configure and enable the wireless modes' functionalities according to their datasheet. DCE (Data Communications Equipment) interface can operate in these individual modes:

- **Command mode:** the DCE waits for AT command instructions. Any characters sent to the DCE are interpreted as commands for the DCE to execute.

- **Data mode:** the DCE transfers data after having sent the "CONNECT" string; any character sent to the DCE is intended to be transmitted to the remote party.
- **Online command mode:** the DCE is communicating with a remote party, but treats signals from the DTE on transmit line as command lines, and sends responses to the DTE on receive line.

5.2.2. Command Line

Commands start with the prefix AT and end with <CR><LF>. The AT commands typically have the following generic syntax:

"AT" <command_name><string><S3_charactor>.

Where:

- <command_name>: command name string; it can have a "+" character as the prefix
- <string>: string consisting of the value parameters following the syntax provided in this manual
- <S3_charactor> command line termination character (<CR>).

5.2.3. AT Commands Mode

5.2.3.1 Action Command

An Action command forces the DCE to transmit information or execute a specific action for the command. A typical use of this command mode is to provide the factory-programmed settings of the DCE like manufacturer name, firmware version, etc.

For example: *ATD123456;* Makes a voice call.

5.2.3.2 Set Command

A Set command is performed to set the preferred settings for any specific command. The Set command is the only way to set the preferred settings in the DCE. It is possible to store the current settings in the profile and to then retrieve them in another connection.

Example: *AT+COPS=0,0*;

Forces an attempt to select and register the GSM/UMTS network operator.

5.2.3.3 Read Command

A Read command provides the current values of the command parameters. It is used to find out the last configuration of the parameters of the command.

For example: *AT+COPS?*

5.2.3.4 Test Command

A Test command provides a complete list of the values supported by each parameter of the command.

For example:

AT+CSQ=?

returns the list of supported Received Signal Strength Indications and in GSM RAT indicates the BER (Bit Error Rate).

For a specific set of commands, the information can be provided directly by the DCE when a specific event happens. This type of command is an unsolicited (or intermediate) result and can be enabled through a set command.

Chapter 6.
Microcontroller

6.1. Introduction

As the brain of the application, the microcontroller typically consumes the most power and has the most control over the system. A microcontroller is a single chip containing a CPU, memory (RAM and ROM), input /output ports, timers, and serial ports. The primary use of a microcontroller is to control a machine or system using a fixed program stored in the ROM and this program does not change over the life time of the system.

Several languages may be used for programming microcontrollers. On one hand, C stands as a high-level language which is easy to understand for humans, and on the other hand, it can usually be compiled into efficient machine code that microcontrollers can process. These features makes C a convenient programming language for microcontrollers.

In this project, C has been used, which is a very common optimal for small microcontrollers.

6.2. Texas Instruments MSP430G2553

The Texas Instrument MSP430G family of ultra-low power microcontrollers consists of several devices featuring different set of peripherals that are targeted for various applications. The MSP430 microcontroller is an extremely versatile platform that

supports many applications. The MSP430 uses both 16 bits wide buses for data and address. The registers in the CPU are also 16 bits.

It is geared mostly towards low energy and less intensive applications that operate with batteries, so processing capabilities and memory, among other things, are limited. With its ultra-low power consumption and peripherals, it enables the designing engineer to meet the goals of many projects. More than a few features make the MSP430 appropriate for low-power and portable applications:

- Its tiny size with a large number of registers makes it efficient.
- The low-power mode configuration is simple, and the mode is controllable via dedicated registers. An interrupt can awaken the CPU, which will go back into low-power after that.
- Depending on the size of operations and the level of activity, multiple types of power modes are specified.
- There are several choices for clocks. A 32 KHz frequency crystal can be used as the clock source, reducing the power consumption significantly. Similarly, a Digitally Controlled Oscillator (DCO) clocks the CPU internally and provides a restart time of less than 1 µs for the MCU.
- The various peripherals that can operate independently from the CPU.

6.2.1. Clock Generator

A number of types of oscillator can be used as the clock source. The two most commons are as follows:

- **Crystal:** provides an accurate and stable frequency for microcontrollers in the range of a few MHz to 32 KHz. One of the disadvantages of this type is the power consumption, particularly at high frequencies.
- **RC:** provides poor accuracy and stability, but it is cheap and quickly accessible.

The TI MSP430G2553 benefits from three internal clocks. They cover the need for low-power, high frequency for more performance, and accurate frequency. The supported clocks for MSP430G2553:

- **Master Clock or MCLK:** Some peripherals and the CPU use it.
- **Subsystem Master Clock or SMCLK:** Is used by peripherals.

- **Auxiliary Clock or ACLK:** Is used by peripherals.

Peripherals can be configured to be run either by SMCLK or ACLK. ACLK runs at a very low frequency and comes from a 32 KHz crystal. Therefore, it consumes low current compared to other clock sources.

6.2.2. Interrupts

MSP430 uses vectored interrupts, and each of the Interrupt Subroutines have its own vector. The address of the ISR is stored in a vector table at a dedicated memory location. This way, it is not necessary to search for the source of the interrupts in most cases. In some other microcontrollers, such as the microchip PIC 16, there is only one vector for all interrupts, which requires checking flags in order to find the appropriate interrupt source.

Interrupts are regularly utilized for different applications such as:

- Executing urgent tasks that have a high priority.
- Saving power for handling slow inputs.
- Triggering for waking the CPU from sleep mode.

Interrupts are categorized as either maskables or non-maskables. Maskable interrupts are effective if the dedicated bit is set. Otherwise, they will be ignored. Non-maskable interrupts cannot be disabled, because the instant watchdog timer's interrupt is not maskable.

If more than one interrupt triggers, then the interrupt that has the higher priority will be processed. A higher priority means a higher address. The addresses are fixed in memory and are not editable.

If the microcontroller is on the sleep or low-power mode, it must return to active mode in order to process the interrupt handler. This feature will be discussed in following chapters.

6.2.3. Power Modes of Operation

The MSP430 was outlined from the beginning for low power. These factors contribute to its operation modes. The MSP430 has the advantage of five power operation modes. The operation modes have three different benefits:

- Ultra low-power
- Speed in processing data
- Minimization of individual peripheral current consumption

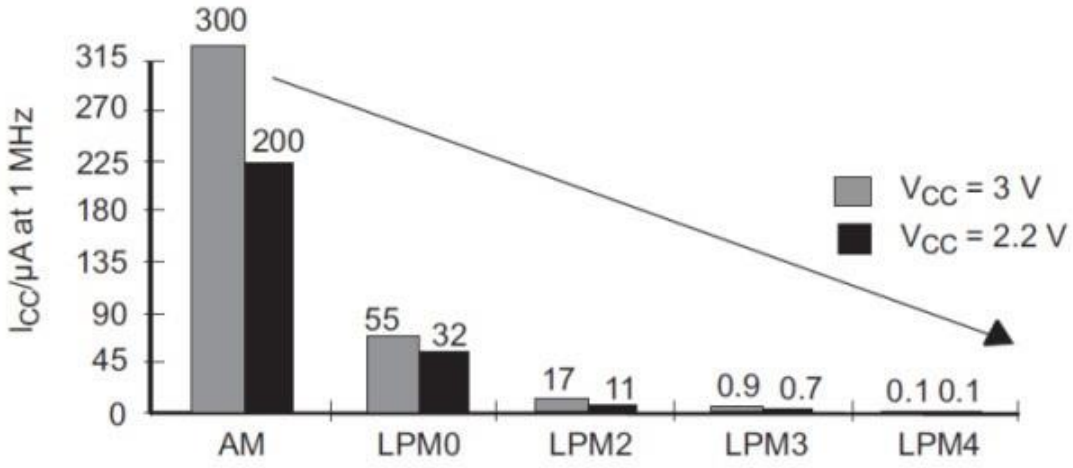


Figure 6-1 MSP430G2553 Operation Modes [23]

Figure 6-2 demonstrates the typical current consumption of each low-power mode (LPM) and active mode (AM) for different voltage sources.

1. **LPM0:** The CPU and MCLK are disabled, and the SMCLK and ACLK remain active. This operation mode is used when some peripherals need a high frequency clock from SMCLK and DCO.

2. **LPM1, LMP2:** These two modes are described briefly in Table 6-1.

3. **LPM3:** The CPU, MCLK, SMCLK and DCO are disabled. In this mode, only the ACLK remains active. This mode usually being used when a peripheral such as timer requires a clock. For example, when the timer is configured to awaken the device frequently or at after a specific amount of time.

4. **LPM4:** The CPU and all clocks are disabled. On this mode, only an external signal can trigger the interrupt and weaken the device. The current consumption in this mode is in the lowest level.

LPM4 and LPM3 have been used the most in this project. The device spent most of time in LPM4 in order to reduce the power consumption to the lowest possible level. More details will be discussed in the Chapter 8.

Table 6-1 Shows the CPU and clocks status for each mode in brief.

SCG1	SCG0	OSCOFF	CPUOFF	Mode	CPU and Clocks Status
0	0	0	0	Active	CPU is active, all enabled clocks are active
0	0	0	1	LPM0	CPU, MCLK are disabled, SMCLK, ACLK are active
0	1	0	1	LPM1	CPU, MCLK are disabled. DCO and DC generator are disabled if the DCO is not used for SMCLK. ACLK is active.
1	0	0	1	LPM2	CPU, MCLK, SMCLK, DCO are disabled. DC generator remains enabled. ACLK is active.
1	1	0	1	LPM3	CPU, MCLK, SMCLK, DCO are disabled. DC generator disabled. ACLK is active.
1	1	1	1	LPM4	CPU and all clocks disabled

Table 6-1 MSP430G2553 Operation Modes Configurations

It is possible to divide all of the operation modes into the two groups of Active and Sleep mode. The device is either in any of the low-power modes or in active mode. While the device is in active mode, it is described as *waking* and *sleeping* when it is in low-power mode. The arrangement of four bits: SCG0, SCG1, CPUOFF, and OSCOFF in the status register state the selection of low-power modes. In active mode, all of these bits are clear, and the device runs with its highest level of current consumption.

Figure 6-3 demonstrates the sequence of operation modes in the MSP430. Interrupts activate the trigger switch in order to awaken the device, so that it can perform each task. Before the device moves to active mode, it enables the MCLK in order to handle the interrupt. This process runs in the hardware level, and so does not require any involvement from the programmer. After handling the interrupt, the device goes back

to sleep mode and disables all of the clocks. Finally, it disables the MCLK to end the ISR. The device remains in the low-power mode until the next interrupt.

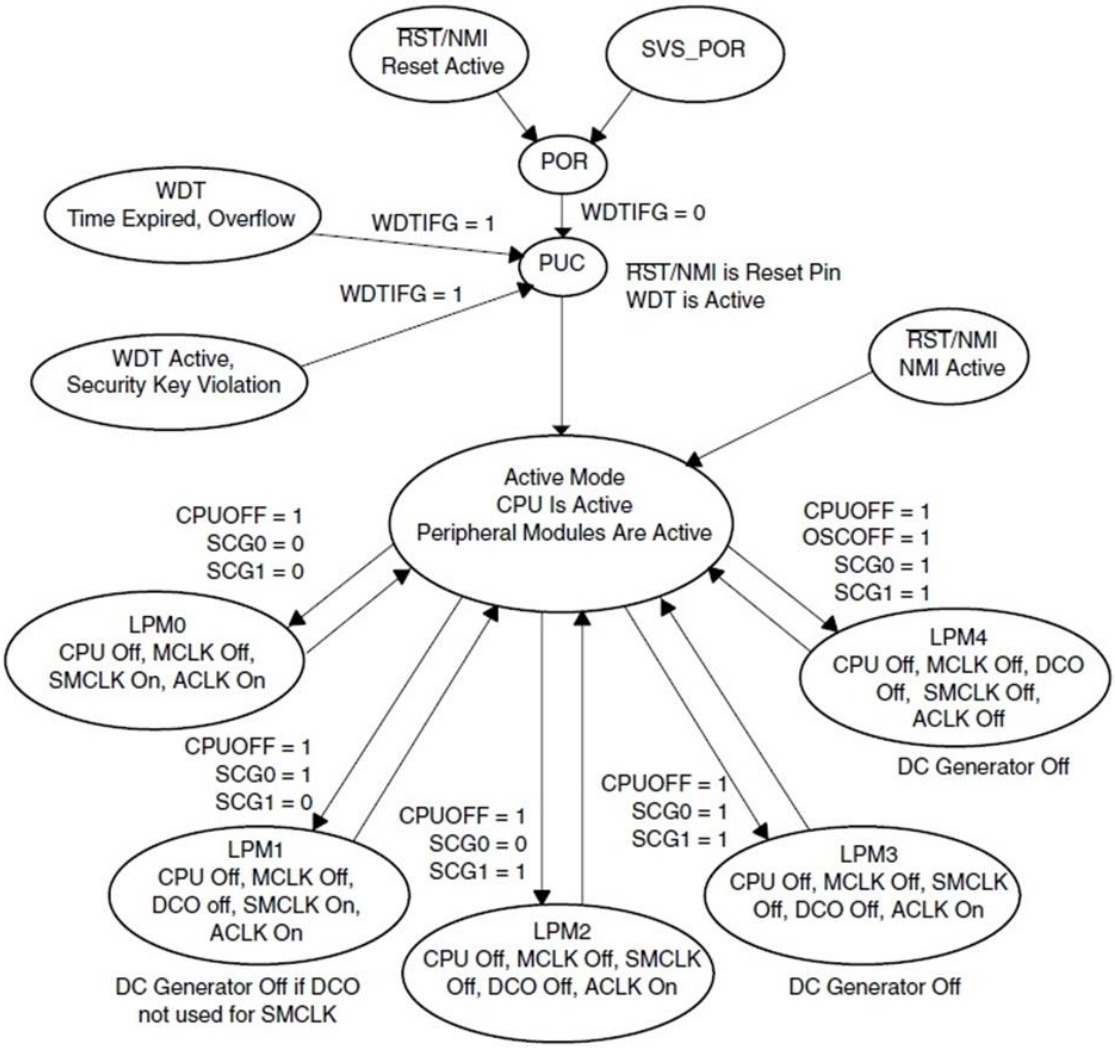


Figure 6-2 Sequence of Operation Modes in MSP430G2553 [23]

6.2.4. TI MSP430's Communication Peripheral

Generally, independent from the type of communication, the MSP430 can transfer a single bit at a time; which is called Serial Communication. The communication types that the MSP430 can handle are:

- Serial Peripheral Interface (SPI)
- Inter-integrated Circuit bus (I2C)
- Asynchronous Serial Communication

Most the MSP430 carry hardware modules to handle them.

6.2.4.1 Universal Serial Communication Interface

The USCI modules handle all type of communication. There are two channels, A and B. They are two separate peripherals but they share a few register and interrupt vectors.

- **Asynchronous Channel A:** Is capable of detecting the baud rate of the signal. It can also handle SPI.
- **Synchronous Channel B:** Handles both the SPI and I2C.

Chapter 7.
Optimizing Embedded System Power Consumption for Low-power Designs

7.1. Introduction

System design is the implementation of a desired performance or service while achieving predefined goals and satisfying existing constraints. These goals and constraints vary significantly from application to application. The outcome of an optimal embedded architecture demands savvy choices of processors, integrated circuit, peripherals, and communication links. As with all designs, it is important for the designer of a low-power embedded system to consider trade-offs between power consumption and other factors such as cost, size, and complexity. While some low-power techniques can be used with no cost to the system, others may require trade-offs.

Low-power perspectives should be used in all structures of a system. Therefore, setting and keeping power requirements have to be considered in both hardware and software design. Although power-efficient design of peripherals is beyond the scope of this project, the elements, components, and electrical specification of a system need to be studied for power optimization.

The hardware design has a noticeably large contribution to power optimization. Reducing the power consumption of all of the constituents of the base hardware can minimize its power dissipation. The peripheral devices tend to consume a significant fraction of the total power budget. Also, without doubt, achieving a low-power design is not possible if one underestimates the influence of software and a controller program. Essential techniques can be used to reduce the data rate and switching activity on the communication links between the hardware platform and these peripherals. Additionally,

dynamic power management policies that aim turning off the peripherals during their idle periods help significantly to reduce power dissipation. These techniques have the responsibility of controlling the unit's behaviour and communications. Also, the techniques have the responsibility of preventing unnecessary running or processing times. Choices for the software implementation affect the power efficiency of the base hardware and peripheral devices.

In this chapter, the techniques of optimizing power consumption will be discussed. These techniques show how it is possible to minimize the power consumption in hardware level, algorithm level, and data-flow level. This will also include the method for implementing algorithms into software for low level devices such as the TI MSP430.

7.2. Power Consumption

Power consumption is one of the most important aspects in the product lifecycle of an embedded hardware device, and is extremely noticeable in tiny devices that use battery as the power source. This type of device should be capable of assuring certain minimum usage/idle times between recharges. Therefore, battery-powered devices require having power optimized design in order to extend the battery life while meeting the performance requirements. Conjointly, the average and peak power consumption have an impact on the battery life [24], packaging costs, and reliability. Consequently, the importance of power consumption is evident, and has to be considered even when designing non-portable devices.

Figure 7-1 shows the total current consumption of a system in a period of time. Total power consumption consists of two types of power [25]:

- Dynamic power consumption
- Static power consumption

Based on these values, the total power is calculated as:

$$P_{total} = P_{Dynamic} + P_{Static} \qquad (7\text{-}1)$$

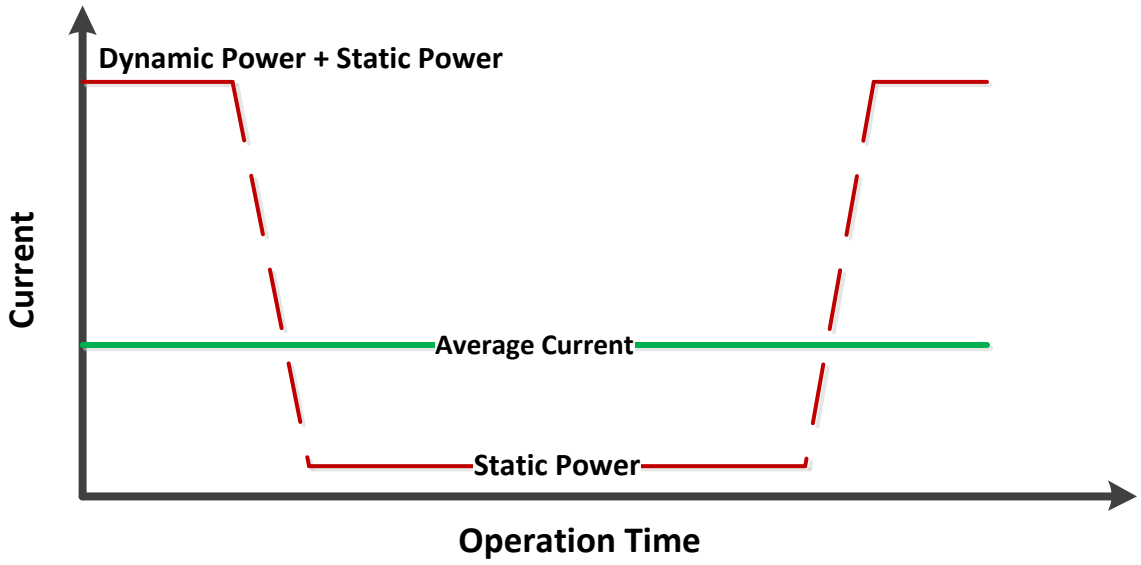


Figure 7-1 System's Current Consumption

7.2.1. Static Power Consumption

Static power consumption, or static leakage consumption, exists in any device. The static power consumptions occur independently of system's activities and encompasses all of the power required to maintain proper system operation while the code is not actively running. This typically includes bias currents for analog circuits, low-power timekeeping oscillators, and leakage current. Static power is a major concern for battery-based systems, which spend significant portions of their application lifetime in Sleep mode [26].

One option to reduce system-level power consumption is to use low static power devices. Analog circuits, such as voltage regulators and Brown-out Resets (BOR), require a certain amount of bias current in order to maintain acceptable accuracy, as temperature and voltage vary. In order to offset the current consumption of many of these modules, the best techniques utilize the flexibility built into the MCU to enable and disable analog blocks as necessary, or they utilize lower power and lower accuracy modes. [25].

7.2.2. Dynamic Power Consumption

Dynamic power consumption refers to the power consumed by the device actively using the processor or microcontroller, peripheral, memory, clocks, and etc. Dynamic power increases by using more components of the system, more cores, more memories, higher clock rates or the speed of switching. This includes the power loss in switching the CMOS circuits and the bias currents for the active analog circuits of the device, such as A/Ds or oscillators.

The dynamic power can be calculated as follows:

$$P = V^2.f.C \qquad (7\text{-}2)$$

where V is the system voltage, f is the frequency, and C is the load capacitance.
Equation (7-2) exposes key factors about dynamic power consumption.

Voltage (V) is the most significant factor in dynamic power consumption because it is squared; frequency (f) is another important factor contributing to dynamic power. As a consequence, adjusting voltage and frequency has a significant impact on power consumption. However, adjusting them on an embedded system required superior consideration since each system has its own boundaries and limitation.

The only control the system's designer has over the internal load capacitance (C) is the ability to enable or disable the circuit's components, such as microcontroller. On the other hand, the designer has control over the external load capacitance of a signal that is routed to an I/O port. For example, the designer can eliminate power losses by not keeping these capacitances larger than the internal capacitance.

It is important to determine the maximum dynamic power consumption in order to properly design the system power supply. For example, many batteries perform differently depending on the rate of current drawn from the battery, and it is important to know what current levels the system's battery is capable of handling [24].

7.3. Power Optimization

To create an effective power-optimized system, it is important to study what causes power-consumption and where to focus power minimization efforts. There are four main factors that play important roles in power consumption of a device:

- Software
- Process technology
- Frequency
- Voltage

The software is basically the algorithm of the device. Power optimization strategies, where the dynamic power is implementable in code, can reduce the power consumption, which will be discussed later in Section 7.3.

The process technology is another constraint of embedded design. Smaller process technology leads to smaller transistors technology. Smaller transistors consume less dynamic. On the other hand, the smaller process technology is, the larger the leakage becomes, as in Figure 7-2. For instance, choosing a microcontroller with a larger process technology will reduce leakage, but with the trade-off of having higher dynamic current consumption.

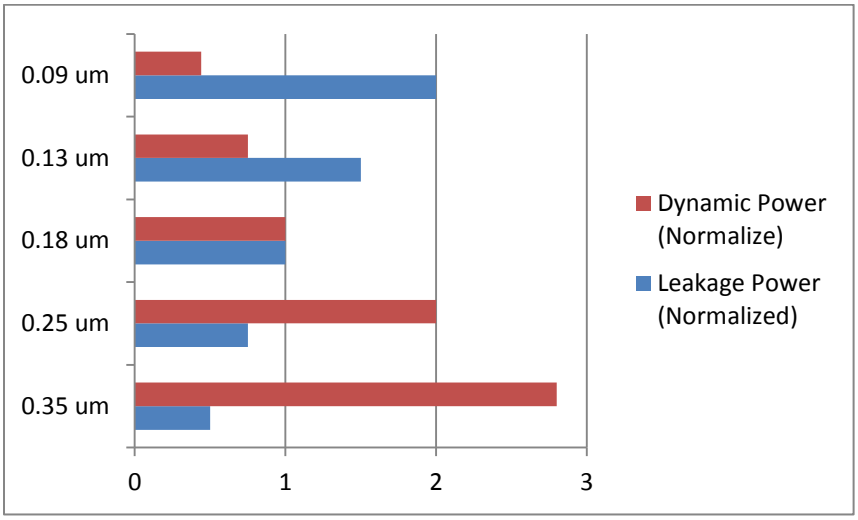


Figure 7-2 Comparison of process technology power consumption [27]

Higher frequency is an advantage, and will provide more speed in processing. Power is a function of voltage multiplied by current ($P = V * I$), where current is a direct result of the clock rate. The higher frequency, along with higher voltage, means more power consumption and also power dissipation.

7.3.1. Power Optimization Methods

Generally, power optimization can be categorized into three types, as follows:

1. Hardware Optimization and Customization Methods
2. Software Optimization
3. Communication Optimization

Hardware optimization refers to the method of clock control and optimization of power consumption using available features in hardware. Software or algorithmic optimization refers to the method of reducing preventable and unnecessary data processing. Communication optimization focuses on buses and peripherals, where data transmissions utilize with considering the power consumption. This can be accomplished by taking advantage of applicable feature in peripherals.

This chapter contains a demonstration of the various effective techniques and methods for hardware and software optimization. In the next chapter, the implementation of these methods will be discussed.

The communication optimization is out of the scope of this thesis, and will not be discussed in detail.

7.3.1.1 Hardware Optimization Methods

All of the methods in this section are discussed with the specific application kept in mind. It is important to think about how the power consumption profile varies based on the application. For example, a music player may work continuously on active mode while playing music, but a security system may spend most of its time in idle mode, and a trigger from its sensors will put the system in active mode. These two applications have very different power profiles.

However, all of these applications have used either gating or scaling, or combination of these, for their power and clock control. The most common methods for hardware optimization are as follows:

1. Power gating
2. Clock gating
3. Voltage scaling
4. Frequency scaling

Any of the abovementioned methods are controllable via software and hardware. However, they are considered hardware optimization methods, since the hardware of the application is required to have adequate features that give these methods their advantage.

7.3.1.2 Power Gating

In order to eliminate the static leakage, a circuit or component will be cut off from its power supply while it is not in use. This can be achieved by using a current switch. Also, often DC/DC converters are provided with an input pin to disable or enable them,

as shown in Figure 7-3. This method is used on a smaller scale inside microcontrollers to control peripherals. Some peripherals may be unnecessary for certain applications and it is possible to disable them by clearing or setting their flags.

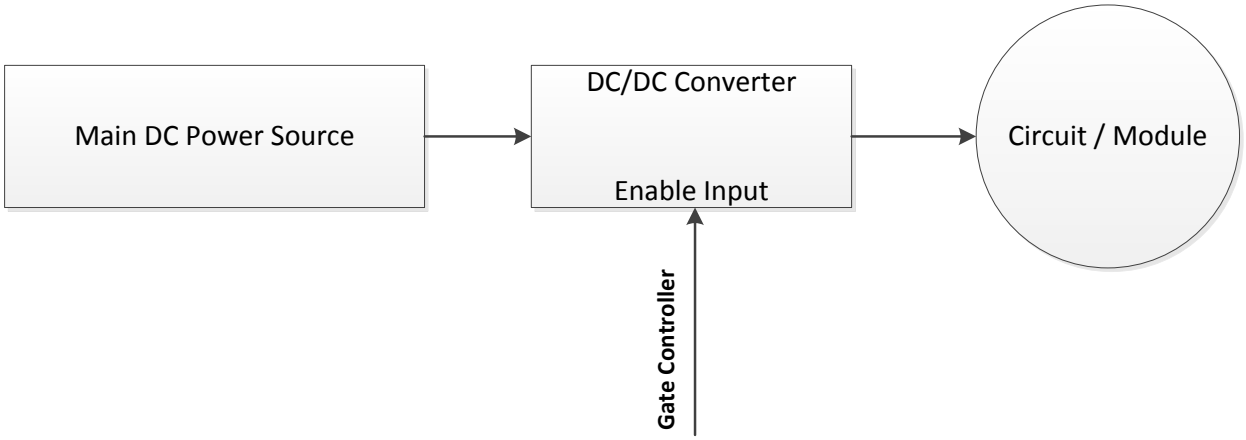


Figure 7-3 Power Gating Method Implementation

7.3.1.3 Clock Gating

Most of the new generation of microcontrollers have multiple clock sources for different proposes. This reduces the power consumption by cutting off unnecessary clock sources for different modes, or enabling them for a certain amount of time in order to handle a task. Besides, clock gating enables the programmer to cut dynamic power through the use of a few instructions.

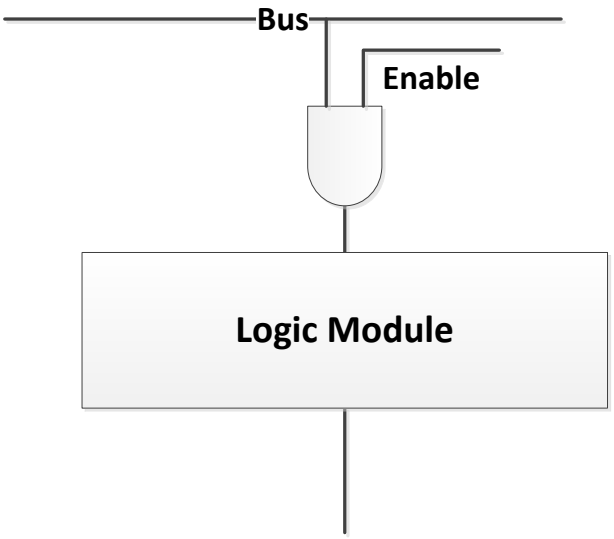


Figure 7-4 Clock Gating Method Implementation

One possible way of implementing clock gating is shown in Figure 7-4. The clock to the register files at the inputs of an unused module is turned on or off using an extra AND gate controlled by an Enable signal. This signal is either introduced explicitly by the system, or generated automatically by the clock synthesis tools. Take for instance the case of a simple microprocessor. Given an instruction loaded in the instruction register (IR), the decoding logic determines which data path units are needed for its execution, and subsequently set their Enable signals to 1.

7.3.1.4 Static Voltage Scaling

As mentioned before, the supply voltage has a direct effect on the total power consumption, and likewise the static power consumption. It is very common in embedded designs to consider a supply voltage source for high power consumers, and a Transistor–transistor logic level voltage (TTL) for low-level power consumers. For example, a DC/DC convertor may have a 15V voltage source for MOSFET drivers and a 3V voltage source for its controller, which may contain a microcontroller.

Sometimes it is possible to split a system into multiple divisions with different supply voltages. Although they may have their supply voltage rates in common, the total power consumption of the divisions with lower voltage rates will decrease [28]. For

instance, in a given embedded device there are units that are functional in 2 to 3.6 volts, and other units' supply voltages are in the range of 3-5 volts. By considering a 2 volt source and 3.3 volt source, there will be a significant decrease in the power consumption. This technique is more noticeable in battery-powered devices that consist of components that are meant to be functional for a large amount of time. Figure 7-5 shows this technique in a block diagram representation.

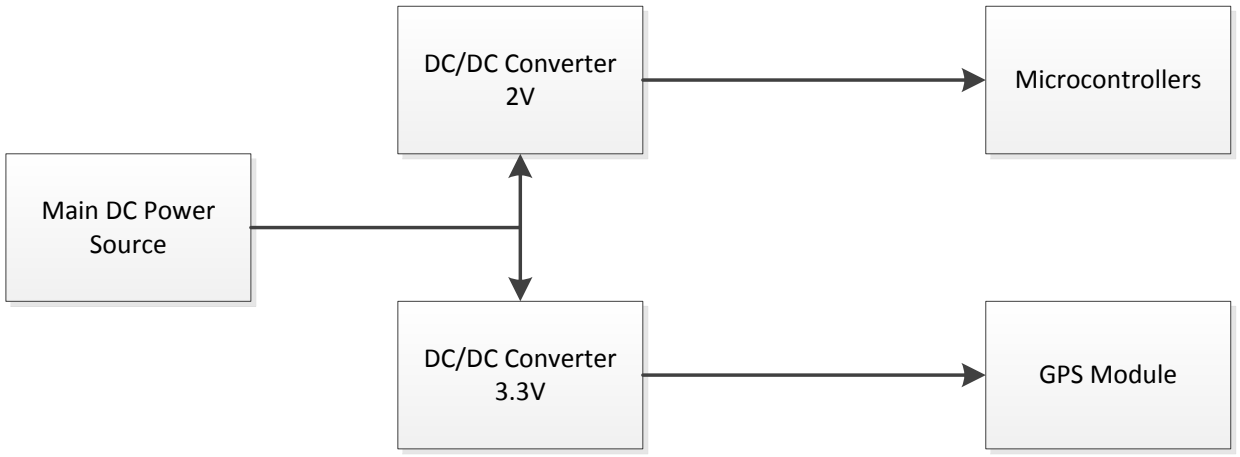


Figure 7-5 Static Voltage Scaling Method Implementation

7.3.1.5 Dynamic Voltage Scaling

Dynamic voltage scaling may apply during the run time of a system in order to reduce power dissipation while the system does not require extra power and stays in idle or sleep mode [28] [29] [30]. Dynamic voltage scaling may use only one power supply for most of the components of a system, and depending on demand, the system increases the level of power [31]. In order to implement this technique, a variable voltage source may be used. Figure 7-6 presents a dynamic voltage scaling power supply.

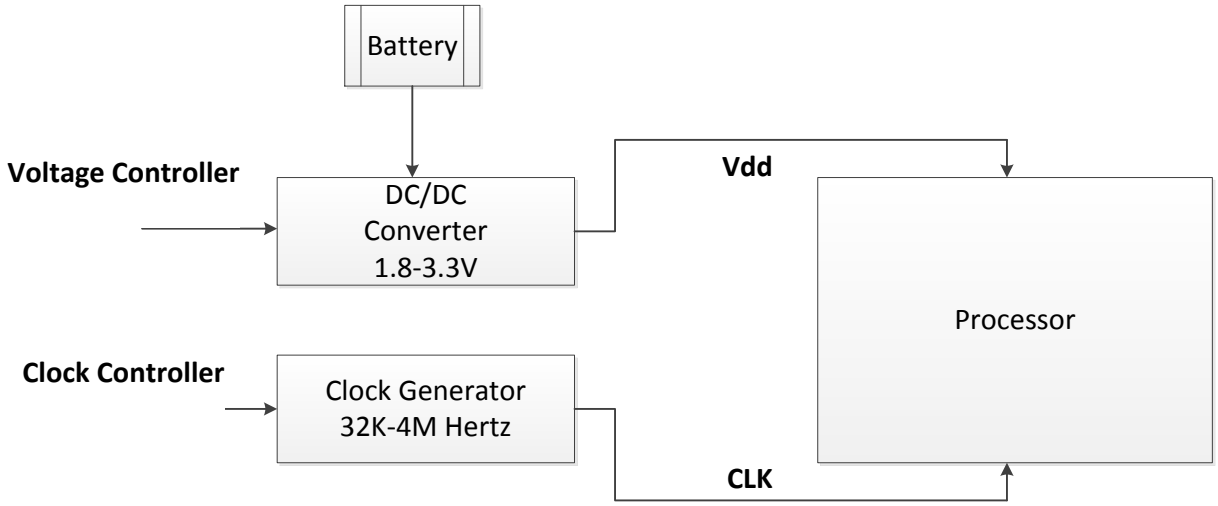


Figure 7-6 Dynamic Voltage Scaling Method Implementation

Hence, the frequency of a clock generator has to change simultaneously with the supply voltage in order to make sure that the circuit can work properly.

Since the static voltage scaling cannot switch to another supply voltage level, it is not efficient for the system where processor may require different power levels for different workloads. Dynamic voltage scaling gives more flexibility to the system for switching power levels for components on demand.

7.3.1.6 Frequency Scaling

Frequency is usually adjusted by the designer in order to control power consumption. Martin [32] studied the effect of frequency scaling on battery lifetime, developing a system for identifying the CPU frequency at which the most computation could be performed using a single battery charge. Flinn [33] conducted a similar study of the Itsy pocket computer, using external power management and off-line evaluation. Micro-benchmarks were used to study the effect of frequency scaling on the processor's performance and power consumption.

The optimal operation frequency for a system is determined by a combination of factors [29] [30] [34]:

- Communication or sampling speed requirements

- Processing performance

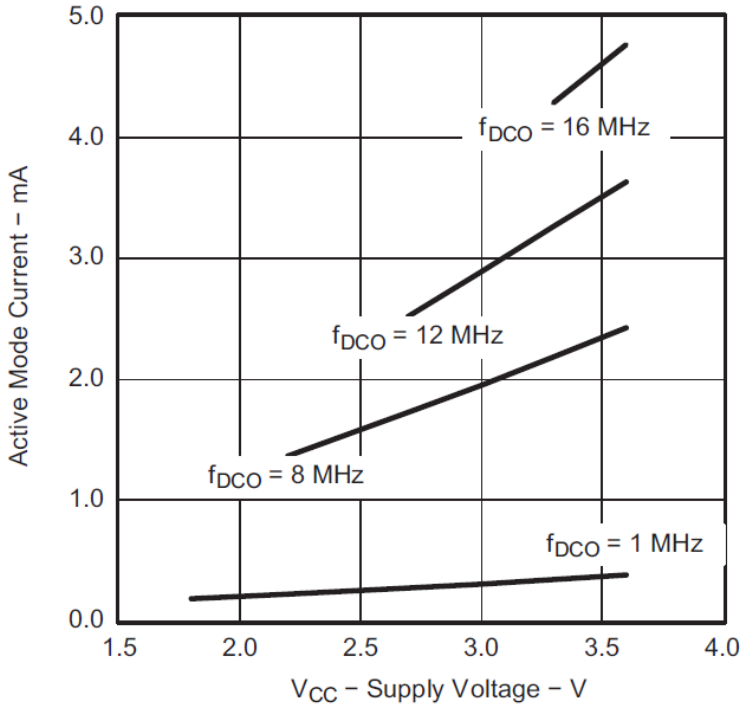


Figure 7-7 Current consumption for different voltage levels and frequency in TI MSP4302553

Figure 7-7 shows current consumption for different voltage levels in TI MSP4302553. As the power equation (7-2) indicates, lower frequencies will result in a lower dynamic current, and it is important to keep in mind that execution speed is also a factor in power consumption. Clock control is available in many processors, which allows the changing of the values of various PLLs in run time. In some cases, it may be optimal to run at a higher frequency and finish an operation more quickly in order to allow the system to return to sleep for minimal power use. In practice, running a processor at a higher frequency allows for more “free” cycles, which can be used to hold the device in a low-power/sleep mode. Also, considering that at low frequencies, the dynamic switching current may no longer dominate system power consumption. Instead, the static power consumption used in biasing the analog circuits on the MCU will dominate.

7.4. Software Optimization

Software design has a critical impact on low-power system design. Selecting the best low-power devices, and optimizing hardware for low power, can easily be wasted if the proper techniques are not utilized when writing system software.

The following sections will focus on the methods of reducing power consumption in software.

7.4.1. Event-Driven System

A very effective technique to operate the system is operating on sleep or idle mode in order to minimize power and run the system on active mode for short time events to handle a task or interrupt. Ideally, the system's behaviors are defined into multiple states, and transitions between the states are dependent on events. Using this method gives the designer an opportunity to reduce power consumption and optimize the system for each state [35].

To design an asynchronous system, it is essential to have a complete profile of the system's power consumption and execution time. Having the entirety of the information helps the designer to balance the trade-offs of power consumption and performance, in order to reduce the total energy consumed in each state. This can be achieved by controlling the frequency and voltage of a circuit using the methods mentioned in the Hardware Optimization section [36]. A combination of these methods in two levels of software and hardware gives an extensional ability to tune the system.

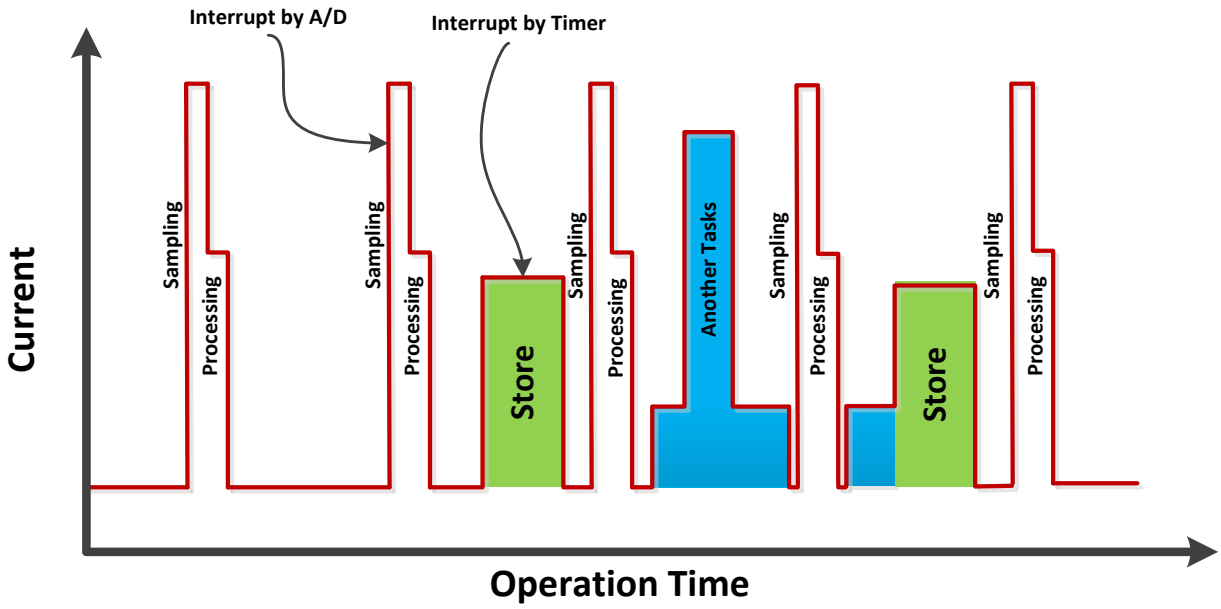


Figure 7-8 Power Consumption on Conditional Code Execution

7.4.2. Interrupts vs. Polling

Wait loops drain power significantly, and designer should consider avoiding a polling-based system [37]. In a polling-based system, the processor may work on active mode while waiting for an event. In an interrupt-based system, the consumed power and time can be utilized to handle another job. Besides, an interrupt-based system allows the system to use power-saving modes, such as Sleep and Idle, much more frequently in the place of waiting for an event with polling.

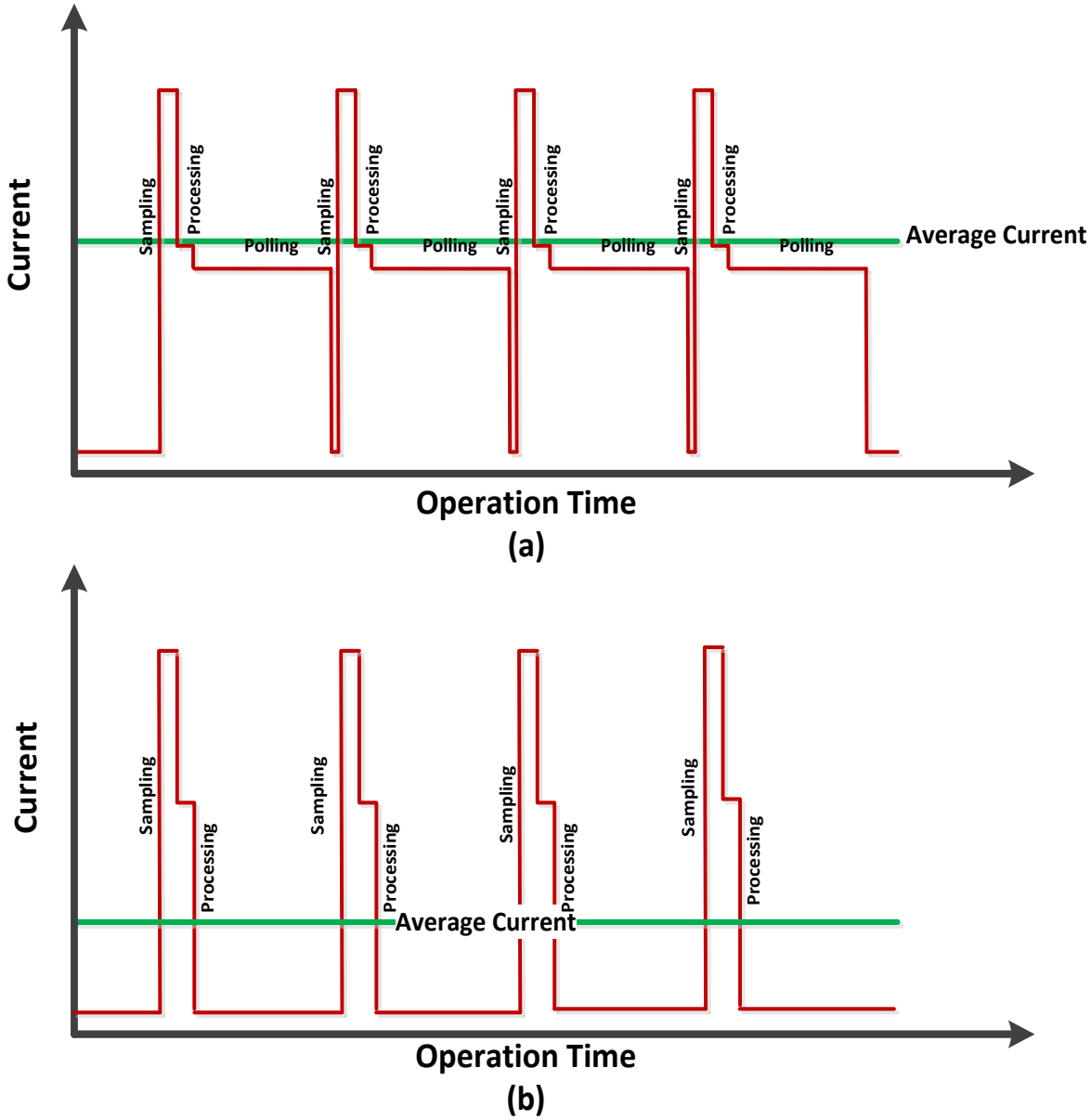


Figure 7-9 Current Consumption of (a) a Polling-based System, (b) an Interrupt-based System

Figure 7-9 demonstrates the current consumption of a polling-based system and an interrupt-based system. In the polling-based system, sensors sample periodically, and the system keeps running actively until a pre-required condition satisfies. On the other hand, the interrupt-based system's sensors sample periodically, and after each sampling, the system goes into sleep mode to reduce power consumption. This application does not require a significant amount of processing time for each sample. It

can wait in idle mode for the next sampling time. Instead of a polling-based method, an interrupt-based method can be used so that the system can use power-saving modes.

Chapter 8.
Design and Development

8.1. Introduction

An embedded system consists of a combination of hardware and software components to form a computational engine that will perform a specific function. The hardware provides the performance. The software provides a majority of the features and flexibility in the system.

In this chapter, the development of system's hardware and power management system will be discussed. Then the implementation of algorithms and operations flowcharts will be explained in details.

8.2. Design of Hardware and Power Management System

The hardware and power management system consists of multiple chipsets and components, along with their electrical specifications. Figure 8-1 demonstrates a big picture perspective of the system's hardware.

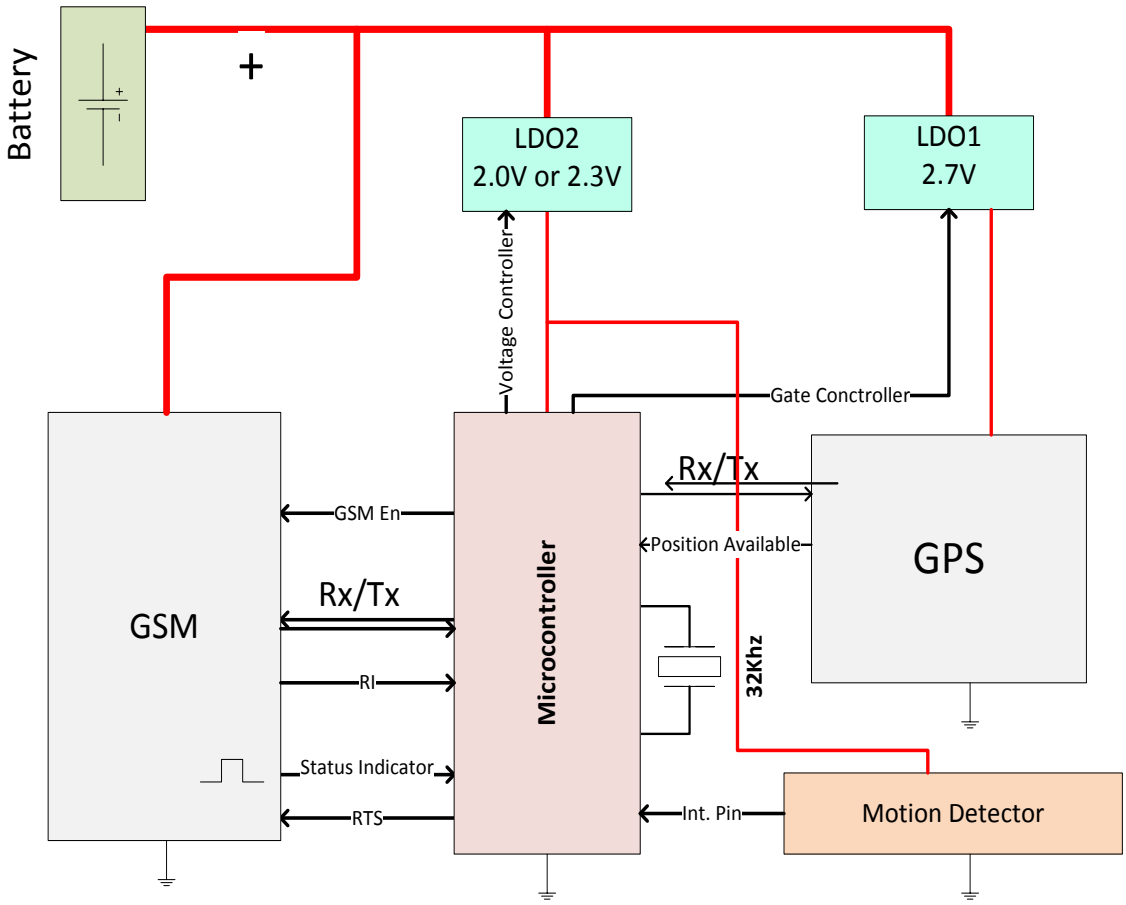


Figure 8-1 Anti-theft System Components

When attempting to prolong battery life, one is required to focus on ways to shut down unused circuitry, ensuring that all quiescent currents and leakage paths are minimized, so the power supply efficiency is maximized. To implement that, two linear drop-out voltage (LDO1 and LDO2) regulators have been used to configure voltage scaling and shunt down unused circuitry. The LDO1 is a fixed voltage output regulator which supplies the GPS receiver and LDO2 is an adjustable voltage regulator which supplies the Microcontroller and accelerometer. The microcontroller controls both LDOs which is discussed in Section 8.2.4.

In this section, each component will be discussed individually, followed by an explanation of the implementation of Power Management's hardware.

8.2.1. Accelerometer - ADXL345

The ADXL345 is used to detect motions on the bike. The ADXL345 communicates with the microcontroller using SPI protocol. SDO, SDA, and SCL are used for SPI communications. Chip Select (CS) pin is required to be kept low during communication.

As mentioned earlier, the ADXL345 is a programmable chipset, and this feature provides an option to configure “Int. Pin”. The Int. Pin is used as a trigger to inform the system of activities on the system.

The supply voltage range of ADXL345 is 2.0V to 3.6V which is a proper voltage ranges for a low-power design. The accelerometer is connected to the LDO2, since the minimum input voltage is 2.0 V. The accelerometer provides a trigger that causes an I/O interrupt in the MCU. This function helps the system to switch to ultra-low power mode, and will be explained in the Section 8.4.

8.2.2. GSM Modem – LEON-G100

The GSM modem is the most power-consuming component in the system. As mentioned before, the modem’s cut off voltage is 3.8 V, and the maximum input voltage is 5V. As a matter of fact, the modem’s minimum input voltage is the highest compared with other components. The modem power is supplied directly from the system’s battery. The GSM_EN pin is configured to switch off the modem when it is applicable, to reduce the power consumption. A Tantalum capacitor (C12 from Figure 8-2) is used at the VCC of LEON-G100 to handle module’s current burst.

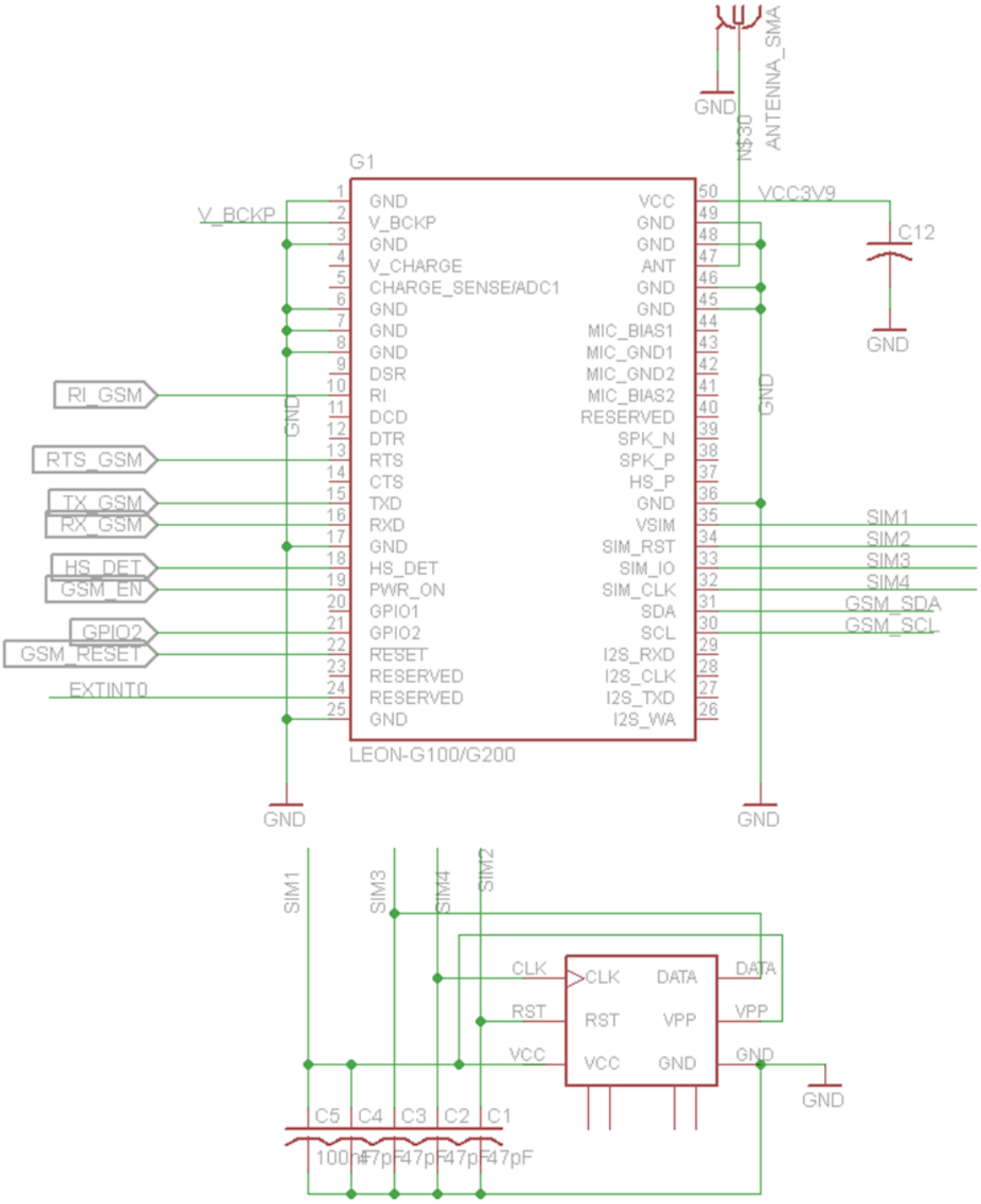


Figure 8-2 LEON-G100 Hardware Setup

SIM1, SIM2, SIM3, and SIM4 are connected to a Cell SIM Card connector.

In order to have full control over power consumption of the modem, several ports are assigned. Theses ports are as follows:

- **RTS** is used to drive the modem into the sleep or active mode.
- **PWR_ON** is used to switch off/on the modem.
- **GSM_RESET** is used to monitor the modems power status and to apply an asynchronous reset to the entire module.
- **GPIO2** is programmed as a network indicator to monitor the modem's network activities.
- **RI** is used as an incoming notification indicator for a call, text message, or packet data.

Communication between the GSM modem and the MCU can be controlled with RI and RTS, which will be discussed in Section 8.3.1.4. The modem can operate in sleep or active mode. Figures 8-3 and 8-4 compare the current consumptions of the modem in active and sleep modes.

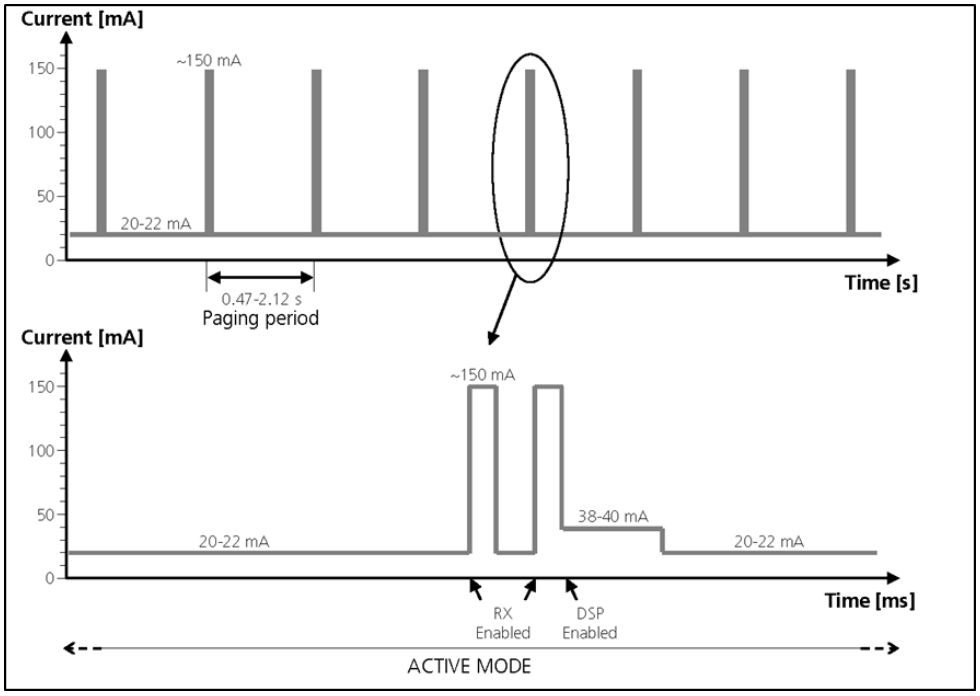


Figure 8-3 the GSM modem active mode current consumption

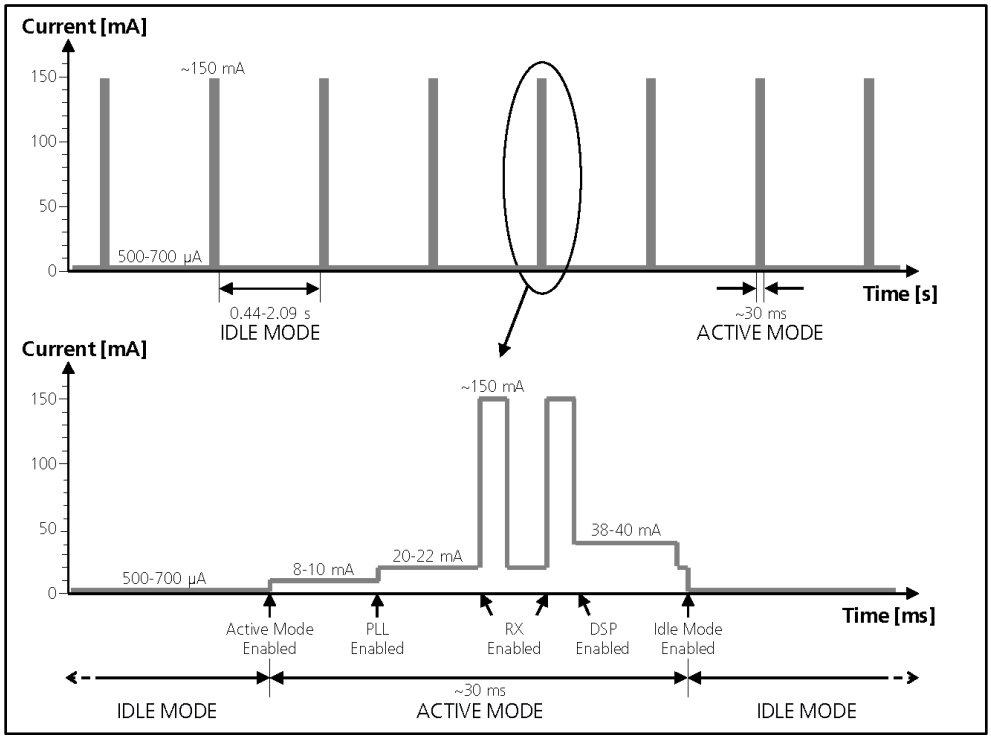


Figure 8-4 GSM Modem Sleep Mode Current Consumption

8.2.3. GPS Receiver – Neo-6M

The Neo-6M's supply voltage is in the range of 2.7 to 3.6. The receiver has the responsibility of positioning and it is the second most power-consuming component in the system. In order to reduce the power consumption of the receiver, Power Gating and Static Voltage Scaling methods are used to increase the battery life. Since the voltage of the battery is more than the receiver's minimum input voltage, adding a low-dropout (LDO1) between the input-voltage source and the GPS receiver reduces the total power consumption (Figure 8-5). The LDO1 also provides an input (the Gate Controller) in order to disable and enable the receiver, which helps to eliminate the power leakage when the receiver is not required. Driving the Gate Controller over 1.2 V turns on the regulator. Driving it below 0.4V will switch the regulator into shutdown mode, which disables the GPS receiver and eliminates the receiver's power leakage.

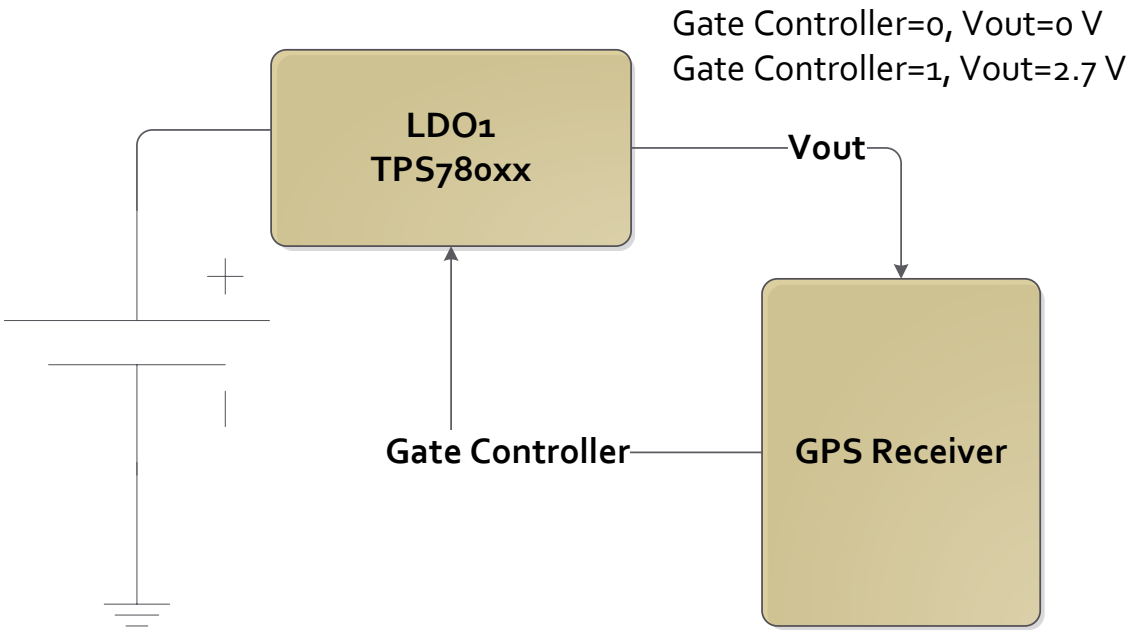


Figure 8-5 GPS Receiver Power Controller

The Neo-6M communicates with the MCU by UART serial communication by using RX_GPS and TX_GPS buses as in Figure 8-6. It calculates the satellites status, along with time and geographic coordinates. Then, the data is sent to microcontroller.

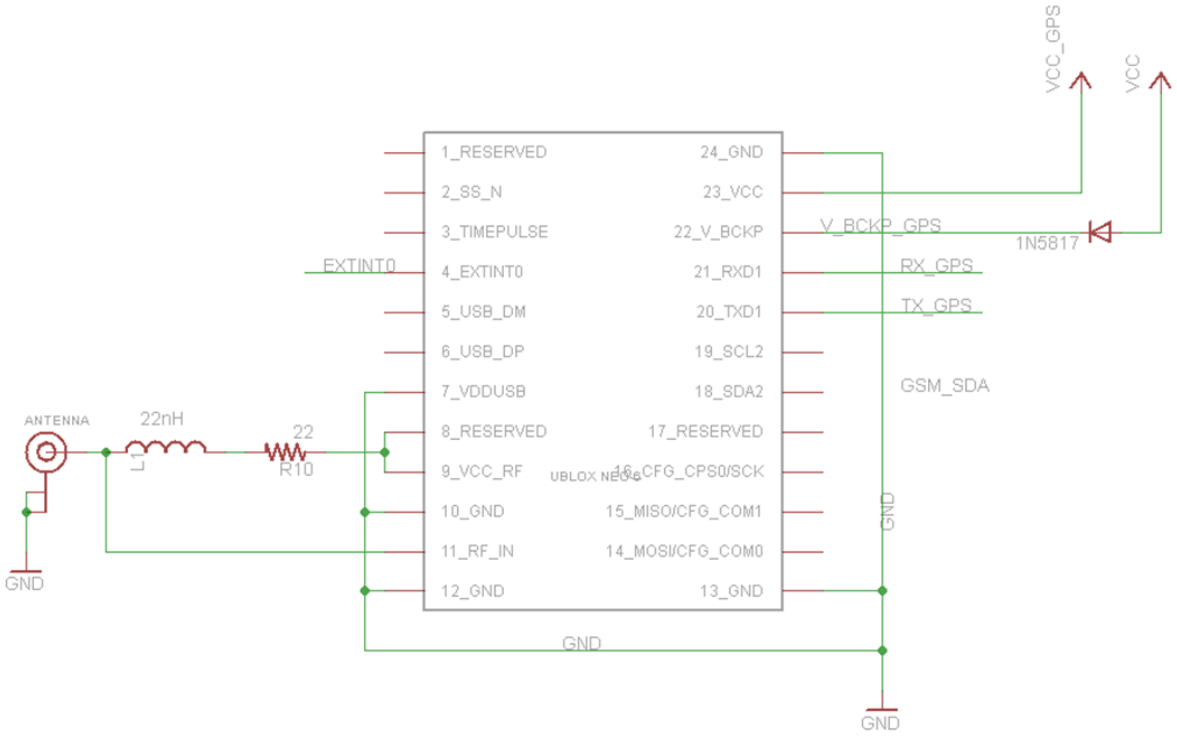


Figure 8-6 Neo-6M Circuit Interface

8.2.4. Microcontroller – TI MSP430G2553

The TI MSP430 microcontroller is used to monitor, control, manage, and respond to activity on the device. The microcontroller also controls the power management system.

Also, the TI MSP430 has the responsibility to communicate with the components and execute the system's algorithms and strategies. The algorithms are programmed into the microcontroller.

8.2.4.1 Supply Power

The TI MSP430 operates with an input voltage of between 1.8 V and 3.6 V. The wide voltage range provides many choices for low-power operation. The VCC_CTRL pin controls the voltage switching, as shown is Figure 8-7.

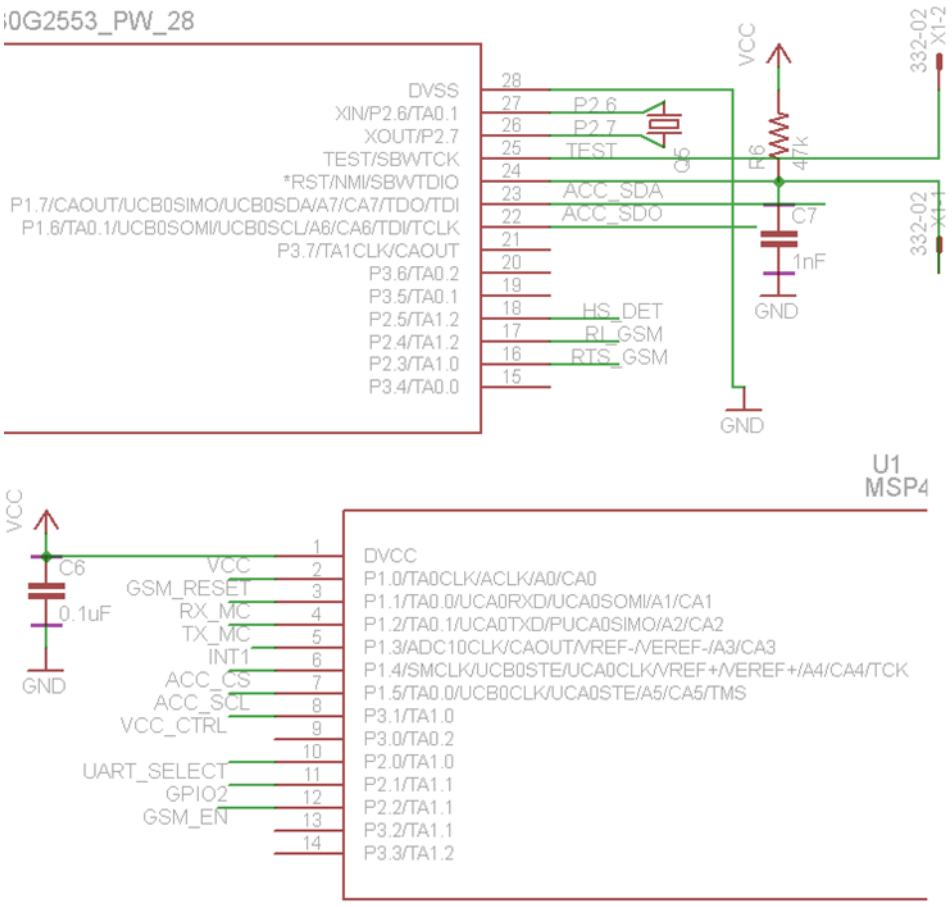


Figure 8-7 MSP430G2553 Circuit Interface

8.2.4.2 Frequency

The TI MSP430 provides different clock source options that can be used to implement Frequency Scaling in order to achieve an optimal operation frequency. The optimal frequency can be defined for two operation modes: sleep and active.

The TI MSP430 in this design uses an external 32 KHz oscillator (Q5) as the Auxiliary Clock Source for individual peripheral modules (figure 8-7). Using 32 KHz, it provides a very low-power option for the MCU that can be used in sleep mode. The Master Clock (MCLK) is configured to operate from the on-chip 8 MHz digitally controlled oscillator (DCO). In active mode, the DCO is used as the clock source. Thus, it requires more power in order to benefit the higher frequency oscillator. Therefore, by considering

the Voltage Scaling method, an LDO (LDO2) with two fixed outputs is in charge of providing the two voltage levels. The LDO2 provides 2.0 V and 2.3 V for this purpose. The 2.0 V is used during the sleep mode, and 2.3 V is used during the active mode. The microcontroller controls voltage levels switching by driving the Voltage Controller pin, as shown as Figure 8-8.

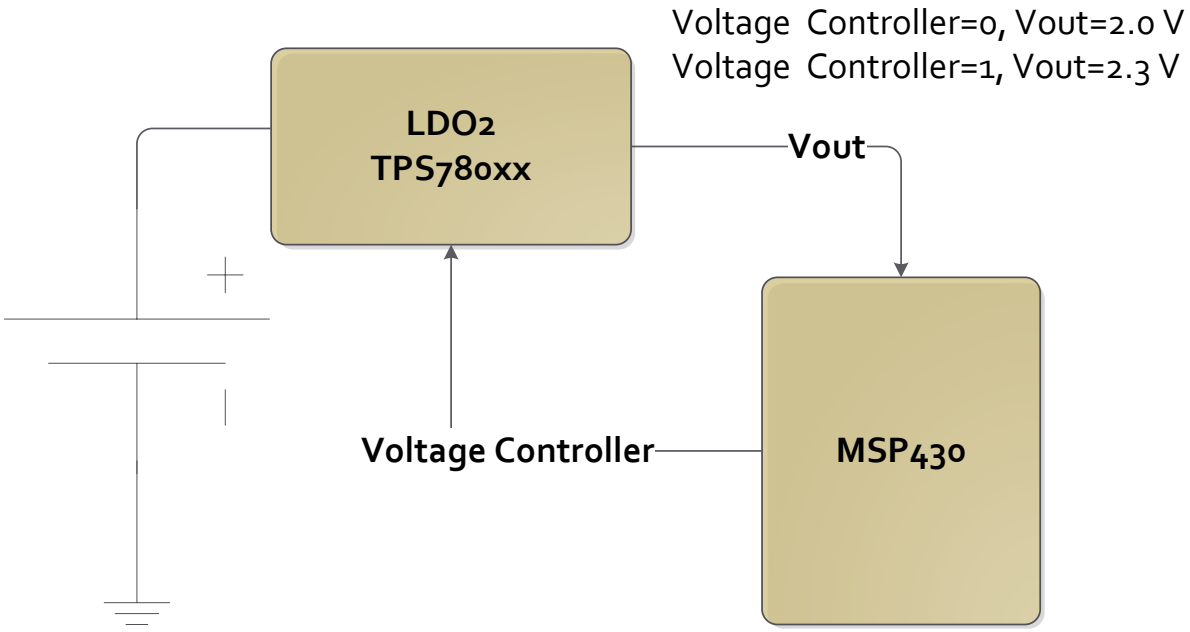


Figure 8-8 Voltage Scaling Implementation on MSP430

8.2.4.3 Communication

As discussed in Sections 8.2.2 and 8.2.3, TI MSP430 communicates with the GPS receiver and the GSM modem using UART serial communication protocol. For this purpose, the Universal Serial Communication interface (USCI_A) is configured on channel A. A Quad Bilateral Switch (CD4016) is used for switching the transmission between the GPS receiver and the GSM modem. The TI MSP430 controls this switching by setting and unsetting the UART_SELECTION pin.

In order to program and configure the accelerometer, the second USCI module (USCI_B) on the MCU is configured to support SPI.

8.3. Software and Algorithm Design

The embedded software development consists of back-end embedded programs implementation on the MSP430G2553 from Texas Instruments, as well as the front-end graphical user interface developed on the mobile. The software algorithms and program details are discussed in the following sections.

8.3.1. Back-end Low Power Design Topology

The back-end software is designed by considering the low-power design topology. In order to reduce its power consumption, the system is designed as an event-driven system. This design is helpful in that it divides and profiles tasks into multiple states. The states are profiled and optimized for less power consumption.

8.3.1.1 Layers

The software is designed into two layers. The top layer has the responsibility of handling tasks. The tasks have different definitions as calling user, sending a text message, getting position, and etc. Depending on the status of the system, it is possible to define multiple tasks to handle in series. For instance, when a bike theft is happening, the system will call the user, and after that it will send a text message to the user. These tasks are defined as “CALL” and “SendMSG”. They are saved in an array, and so the system will execute the tasks in row. Figure 8-9 demonstrates the process.

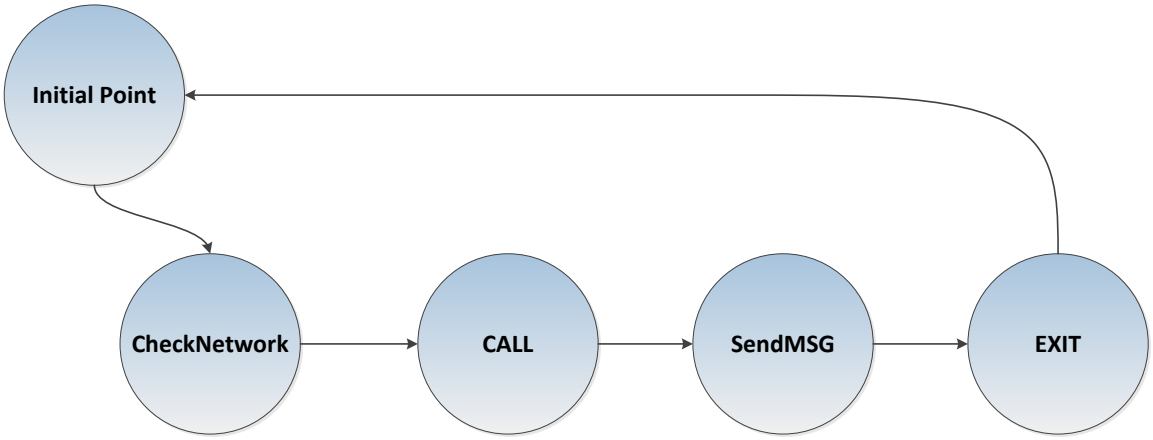


Figure 8-9 The process of executing tasks stored in $\boldsymbol{array[\ ] = \{CheckNetwork, Call, SendMSG, Exit\}}$

The “Task Handler” starts a task by calling “Event Handler Routine” and giving the name of the task as the input. Each task may contain multiple states. “Event Handler Routine” monitors and handles the process of each state in order to fulfil a task.

Event Handler Routine is programmed to acknowledge the system status and initialize the system for a requested task before proceeds, e.g., to make a call on cellular network (CALL task), Event Handler Routine first points to a predefined state where it checks the GSM modem’s responsibility, then continues to another state to check the network signal quality. If the modem is not ready, Event Handler Routine may continue to other states to initialize the GSM modem, as in Figure 8-10.

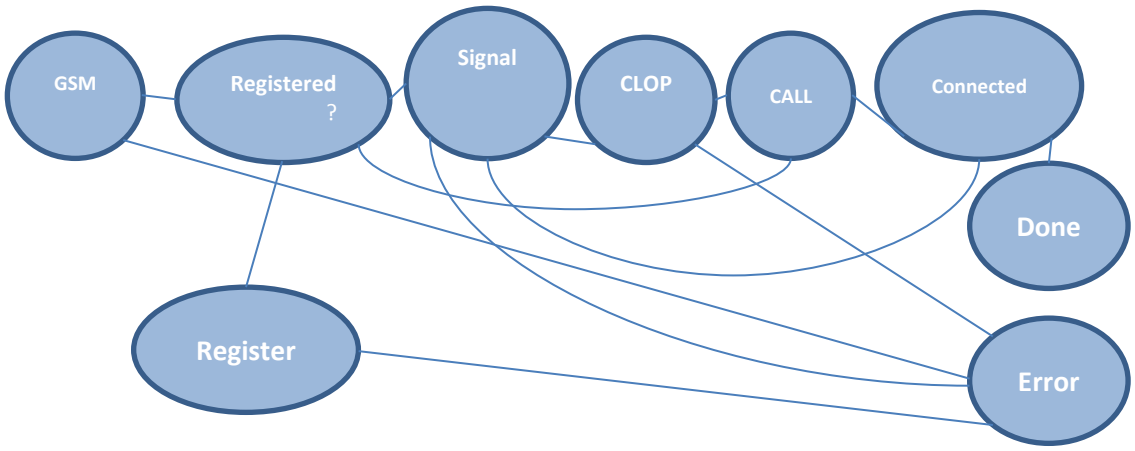


Figure 8-10 Implementation of the "CALL" Task

Practically, the system is an event-driven system, and is based on a state machine. The system switches between pre-defined states in order to fulfil a task. Around 100 states have been defined for the back-end core, and the transmissions between the states make the system functional. Switching between stats happens upon an event triggers the system. Interrupts mainly provide the event triggers. The system uses interrupt functionality for its peripherals, such as Serial Communication, timer, and I/O pins.

The purpose of the state machine is to provide a clear image of the status of the system and its consuming power component so that it can be optimized. To create a power profile:

1. The system is broken down into states.
2. Each state is measured in terms of its power needs and execution time.
3. The total energy consumed in each state is measured.

The result is used to eliminate unwanted power consumption, and to optimize each state.

8.3.1.2 Delays and Real-time Tasks

In order to implement an event-driven system, the system uses timer interrupts for delays. Using this method delivers almost accurate timing interrupts in the system. Additionally, this function is used for required delays or frequent tasks.

Moreover, it does not freeze the system like a polling system would, and it gives space to the system for other tasks. Also, it lets the system to go back to sleep mode.

8.3.1.3 Communications

The communication is designed with having low-power topology in mind. The topology can be expressed as keeping the system on sleep mode until an event happens, as shown in Figure 8-11. For example, an event can happen when a module tries to communicate with the microcontroller. As result, the interrupt drives the microcontroller on active mode to receive the data. Then, the module will switch to sleep mode.

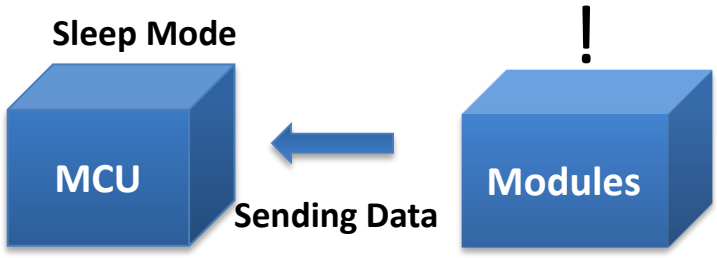


Figure 8-11 Awaking MCU from Sleep Mode

This method has been used widely in serial communication protocol. The GSM modem uses UART and AT command protocol to communicate with the microcontroller. Since its responses are mostly slow (because of its nature), the system saves significant power by using this method. For instance, GPRS activation may take up 10-15 seconds depending on the signal condition, and the microcontroller can be in sleep mode during this period.

In order to awaken the GSM modem or the microcontroller, a hardware flow control system has been used. It was possible to use a USCI hardware interrupt instead, but the GSM modem may have missed the first character, since UART is an asynchronous communication protocol. The hardware flow control system will be explained in the next section.

The same topology is used for the GPS receiver because it also uses UART communication. But the receiver does not provide additional flow control pins, and the

first character is negligible. Therefore, the USCI interrupt used for this project. The GPS receiver uses NMEA protocol to send the information to the microcontroller.

The accelerometer uses SPI, and since it is a synchronous communication, the USCI interrupt used here.

8.3.1.4 Hardware Flow Control

The Hardware Flow Control is used for switching the operation modes of the GSM modem and the microcontroller. The two components consume a wide range of stored energy in the battery, and reducing the time they spend in active mode will noticeably reduce the power consumption.

For this purpose, the RI and CTS are configured to notify the other components about a communication request. In this configuration, the microcontroller commands the flow control. The CTS line directs the GSM modem to operate in the active mode and to enable its UART interface. Driving the CTS line into the ON state or to the OFF state will change the state of the modem to active or sleep mode.

On the other hand, the RI is configured to change the state of the MCU. When the modem is ready to communicate with the microcontroller or to notify it about an incoming message, it drives RI to the ON state. This will end with an interrupt in the microcontroller. This behavior allows the modem to go back to sleep mode until the microcontroller gives the modem a response.

8.3.1.5 Positioning

As described before, depending on the signal quality, it can take about 13 minutes to determine the position on Cold Start mode. During this period, a GPS receiver is required to work on the active mode. It drains the battery quickly and reduces the battery life dramatically. Consequently, it is important to turn off the receiver in order to reduce the leakage power.

With the intention of reducing the TTFF, the Assist GPS method is used to download the almanac data directly from the internet. The satellite data is downloaded

based upon the positioning requested. First, the microcontroller turns on the GPS receiver by keeping the Gate Controller of LDO1 high. The GPS receiver requires time to initialize and receive the satellite signals. Meanwhile, the microcontroller commands the GSM modem to create a TCP/IP connection and download the assist data from a predefined URL. Then, the downloaded data is put into the GPS memory to accelerate the positioning time. Figure 8-14 briefly shows the process of the locating task.

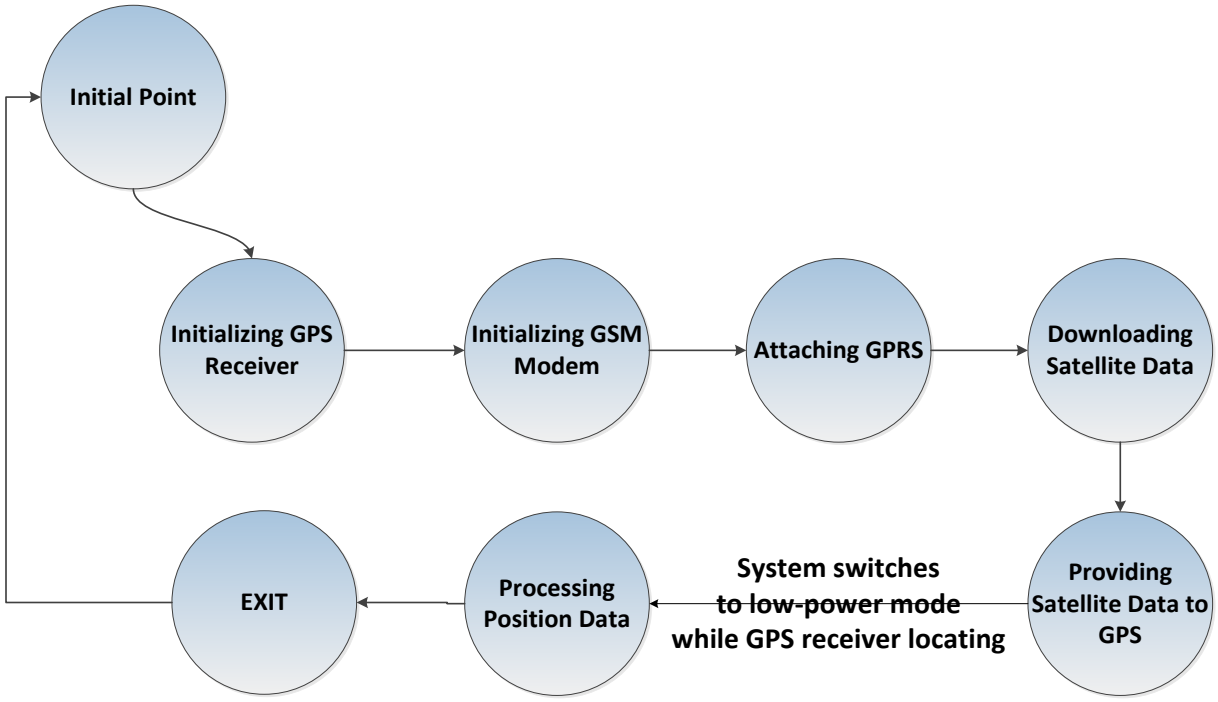


Figure 8-12 Locating States

In poor signal conditions where the receiver cannot identify at least three satellites signals, positioning is not possible. As a result, even A-GPS is not a practical solution. CellLocate Technology is used to estimate the position of the receiver in these situations. The system essentially uses a combination of A-GPS to locate its position. The combined method reduces the TTFF to less than 1 minute in the worst case scenario. Figure 8-13 shows the current consumption for locating when using only the GPS receiver compared to combinations of A-GPS and CellLocate Technology. In this locating example, the total energy consumed for tracking using a standalone GPS

modem is 118.9 Joules, and the total energy consumed using combination of A-GPS, CellLocate Technology, and the GPS modem is 11.45 Joules. It can be seen that reducing the TTFF remarkably reduces the total energy consumed in order to obtain a position.

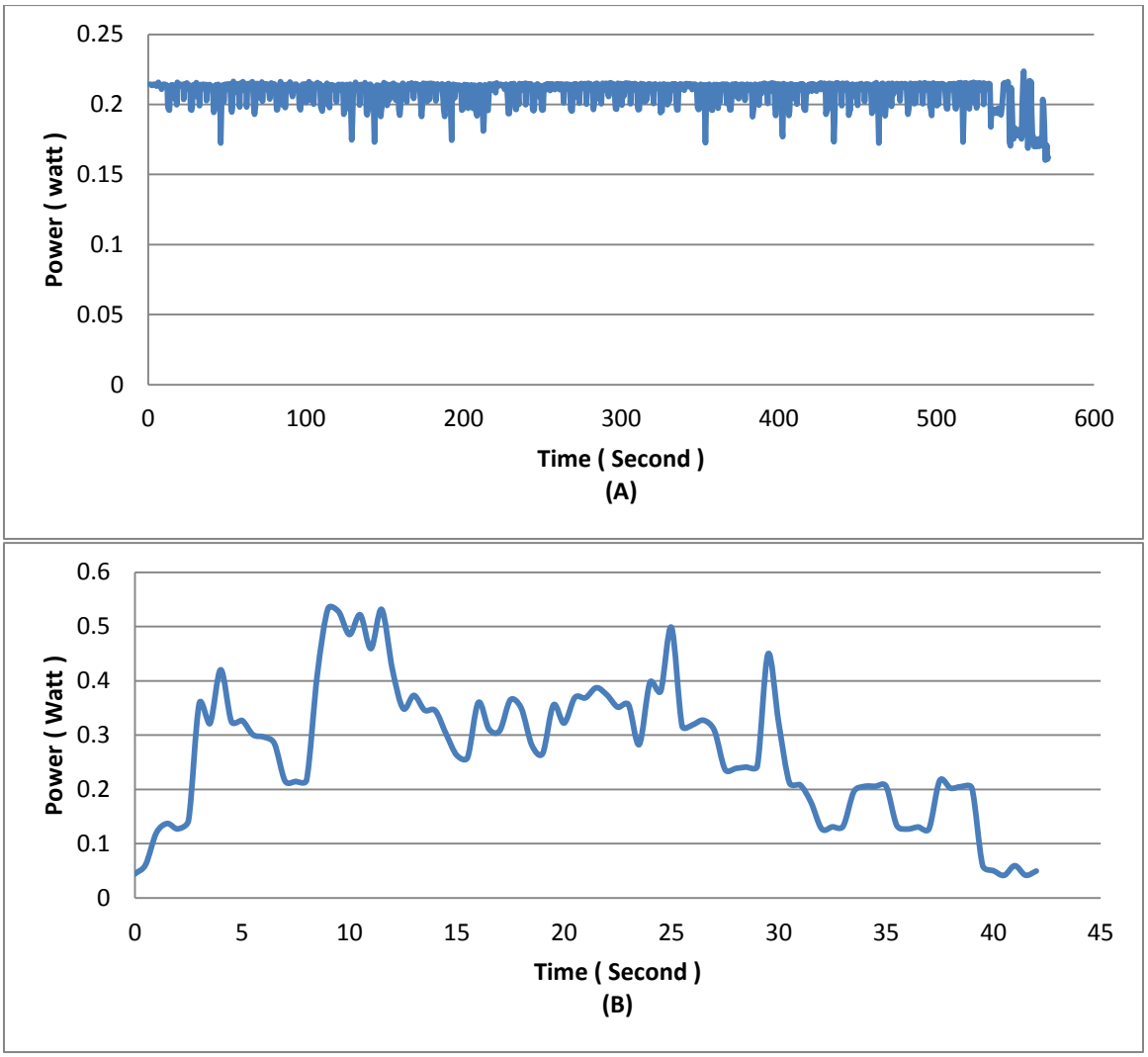


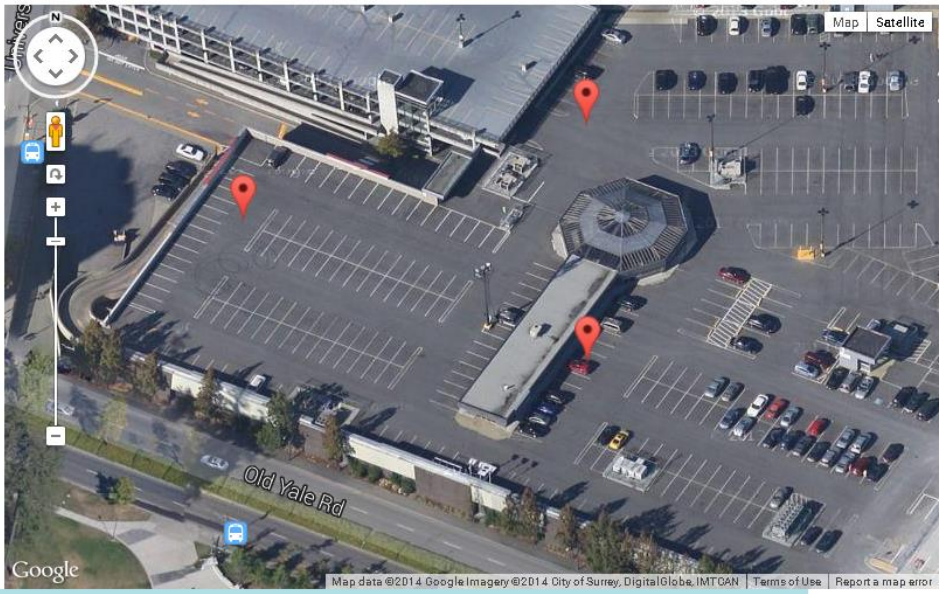
Figure 8-13 Comparing tracking using a (A) GPS modem, (B) combination of A-GPS, CellLocate technology, and a GPS modem

8.3.2. Front-end Software Design

The front-end software contains a server, mobile application, and web application. The server is designed to communicate with the system using UDP protocol on the mobile network (Figure 8-15).

The web application is available from any browser, and it is designed to show the location of the device on Google maps. Figure 8-15 shows a screen shot of the page.

Webpage Screenshot



#	Time	Latitude	Longitude	X	Y	Z
1	2014-12-22 12:58:21	49.1860665	-122.850269	65533	65518	117
2	2014-12-22 12:56:19	49.1863665	-122.851058	8	65512	121
3	2014-12-22 12:54:49	49.1865665	-122.850271	-5	-9	120
4	2014-12-22 12:52:49	49.1865665	-122.850223	7	65513	122
5	2014-12-22 12:50:55	49.1865665	-122.850211	65530	65516	95
6	2014-12-22 12:49:41	49.1865665	-122.850252	65530	65511	101
7	2014-12-22 12:46:51	49.1865865	-122.850261	65528	65519	120
8	2014-12-22 12:45:25	49.1865940	-122.850260	5	65516	120
9	2014-12-22 12:44:58	49.1866059	-122.850269	4	65516	120
10	2014-12-22 12:43:15	49.1866611	-122.850170	65534	65513	110

http://www.novelglance.com/api/map.php Mon Dec 22 2014 15:40:24 GMT-0800 (Pacific Standard Time)

Figure 8-14 Web Interface

The mobile application can also show the location on Google maps (Figure 8-15). Also, it can send a tracking request via server to the system. This function is accessible via a two-way communications system. The designed two-way communication system is out of the scope of this thesis and will not be discussed.

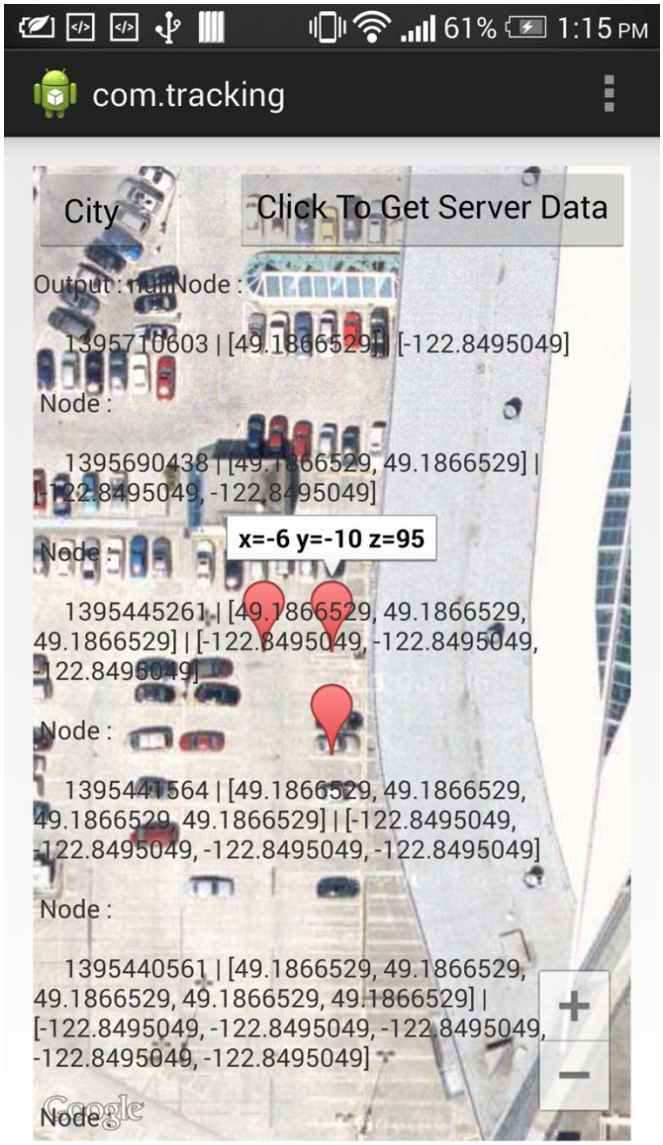


Figure 8-15 Mobile Application Interface

8.4. System Operation Modes

The system's operation modes can be defined as active mode, sleep or low-power mode, and ultra-low power mode. Figure 8-14, presents the circumstances of each mode.

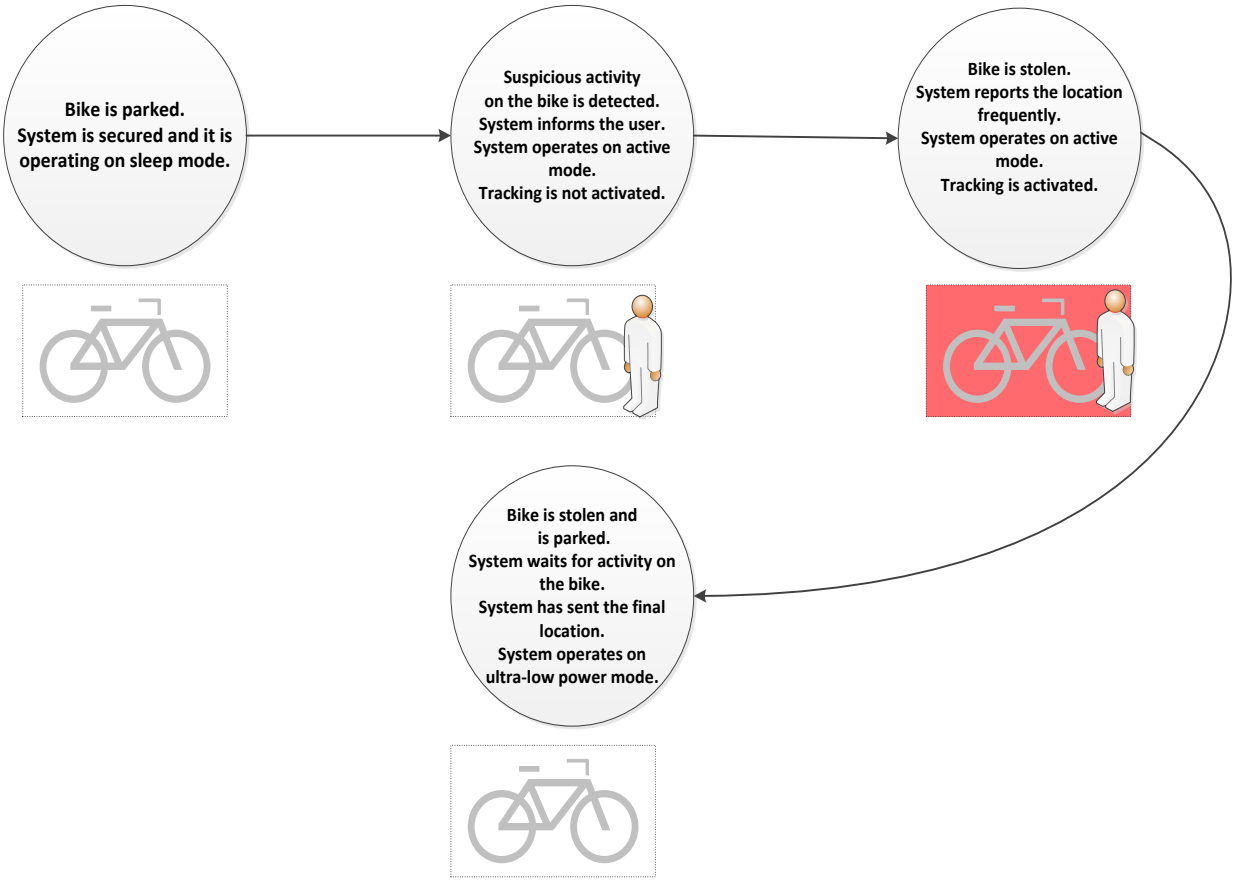


Figure 8-16 System Operation Modes Circumstance

On active mode, the system is either trying to communicate or track the position. In this mode, all of the components are expected to operate on active mode and it is considered the most battery-consuming mode of the system. However, the duration is minimized to reduce the total consuming power, and some components may switch back to sleep mode even for a short amount of time to reduce the power consumption.

In sleep mode, the system is fully responsive and is waiting for an activity. In this mode, all the components operate on sleep mode except for the GPS receiver, which is turned off.

In ultra-low power mode, the MCU turns off the GPS receiver and the GSM modem. Also, the MCU operates on its lowest power operation mode (LPM4). In this mode, only an interrupt from the accelerometer can switch the system into the active mode. Figure 8-17, compares the power consumption of each mode.

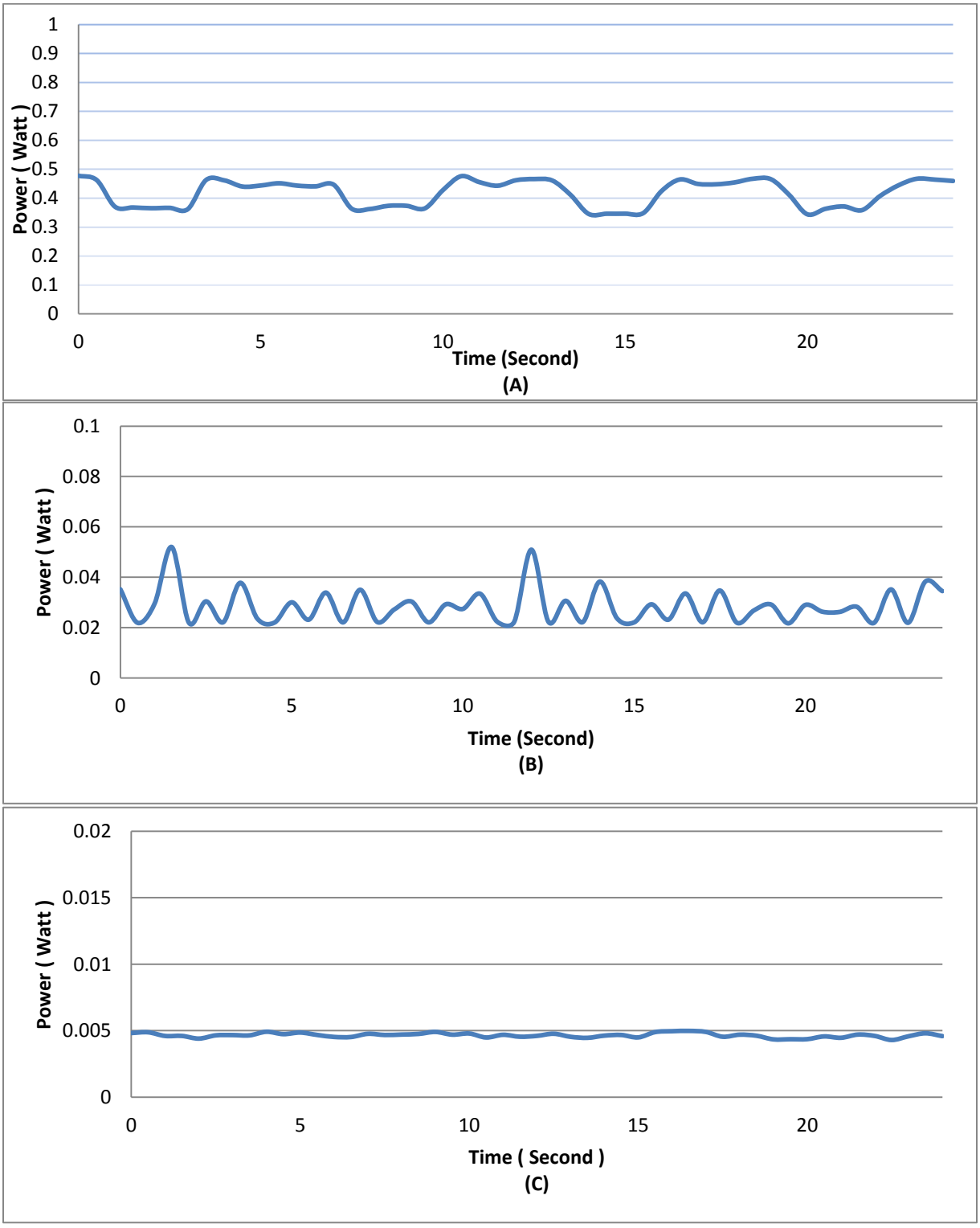


Figure 8-17 Operation modes power consumption comparison: (A) Active Mode, (B) Sleep mode, (C) Ultra-low power mode

8.5. Concluding Remarks and Directions for Future Research

In this thesis, a wireless embedded system was studied and utilized in terms of energy efficiency. The main application addressed was a low power anti-theft system for bikes. The system was designed in two layers of hardware and software. A power management system for this application was designed and implemented into the system. A commercial alpha prototype of the system was developed and tested, as shown in Figure 8-18.

Utilizing energy-optimized systems leads to decreasing power consumption and increasing battery life. This strategy makes it possible to design a highly low-powered portable device. This is especially important when the cost of battery in terms of size and capacity is high for a tiny portable device. To achieve this objective, a low-power tracking device for bikes was implemented. The experimental results demonstrate how the power consumption of a wireless device can be decreased by defining multiple states that are power-optimized individually. Although the concept of a tracking device using GSM and GPS technology is not new, customizing the system for bike anti-theft is a new approach. The most important research outcomes are different operating modes, which give the system options to reduce the power consumption as much as possible. The ultra-low power mode, which is designed to respond only to an external trigger, provides a new approach to increased battery life for portable devices. For instance, this approach can be used for data acquisition in medical portable devices that monitor the activity of patients, and it is recommended for future work.

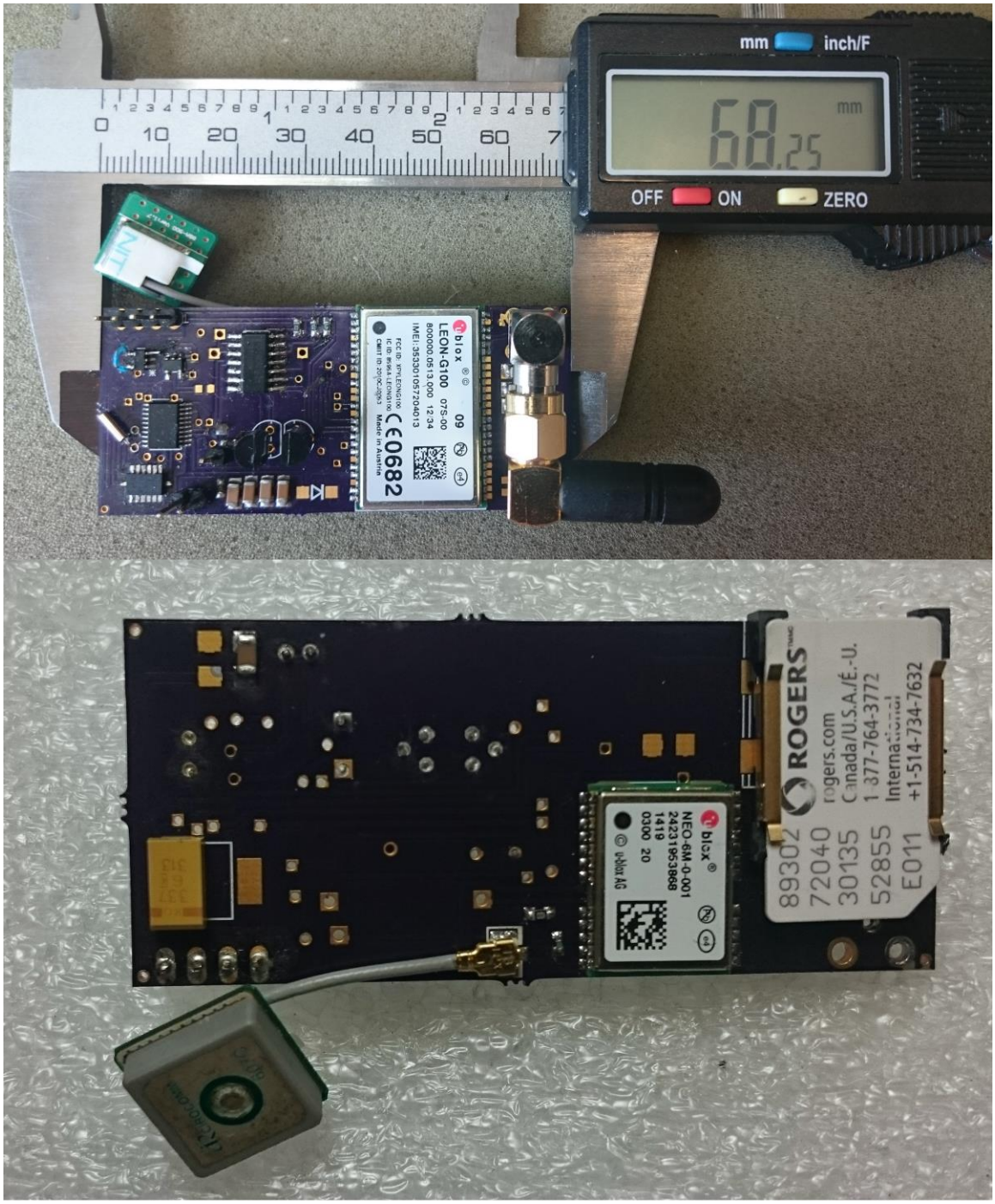


Figure 8-18 Anti-theft System Prototype

References

[1] Toronto Police Service, "Annual Statistical Report," Toronto, 2011.

[2] M. MCCARTHY, "The Vancouver Sun," 4 June 2013. [Online]. Available: http://www.vancouversun.com/news/Bike+theft+growing+industry/8478994/story.html. [Accessed 10 2014].

[3] "Transportation Alternatives - Cycling News: Bike Parking," Fall 2011. [Online]. Available: http://www.transalt.org. [Accessed December 2013].

[4] E. Jafee, "Why Bike Theft Is So Hard to Stop," 14 September 2012. [Online]. Available: http://www.citylab.com/commute/2012/09/why-bike-theft-so-hard-stop/3274/. [Accessed 2014].

[5] Association of the European Two-Wheeler Parts & Accessoriese Industry, "EUROPEAN BICYCLE MARKET," COLIBI, 2012.

[6] G. L. a. H. G. T. E. Melgard, "GPS SignalAvailabilityin an Urban area-receiver performance analysis," in *Position Location and Navigation Symposium*, Las Vegas, NV, 1994.

[7] C. Mathas, "Article Library > What You Need to Know to Choose an Accelerometer," 17 10 2013. [Online]. Available: http://www.digikey.ca/en/articles/techzone/2013/oct/what-you-need-to-know-to-choose-an-accelerometer. [Accessed 2014].

[8] D. Wells, Guide to GPS positioning, Canadian GPS association, 1987.

[9] " U.S. Naval Observatory," [Online]. Available: http://tycho.usno.navy.mil/gpscurr.html.

[10] J. B.-Y. TSUI, Fundamentals of global positioning system receivers: a software approach, New York: Wiley & Sons, Inc, 2000.

[11] G. M. Djuknic and R. E. Richton, "Geolocation and Assisted GPS," February 2001. [Online]. Available: http://www.cs.columbia.edu/~drexel/CandExam/Geolocation_assistedGPS.pdf. [Accessed February 2014].

[12] A. H. J. I. Mikko Lehtinen, "Accuracy and time to first fix using consumer-grade GPS receivers," in *Software, Telecommunications and Computer Networks, 2008. SoftCOM 2008. 16th International Conference on*, Split, 2008.

[13] D. Raskovic, F. Univ. of Alaska Fairbanks and D. Giessel, "Battery-Aware Embedded GPS Receiver Node," in *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, Philadelphia, PA, 2007.

[14] C. Gentile, N. Alsindi, R. Raulefs and C. Teolis, Geolocation Techniques Principles and Applications, Maryland: Springer New York Heidelberg Dordrecht London, 2013.

[15] u-blox, "AssistNow u-blox A-GNSS services," [Online]. Available: http://www.u-blox.com/images/downloads/Product_Docs/MultiGNSSAssistNow_ProductSummary_%28UBX-13003352%29.pdf.

[16] u-blox, "CellLocate™ and Hybrid positioning," [Online]. Available: http://www.u-blox.com/images/downloads/Product_Docs/CellLocate_ProductSummary_%28WLS-SW-11002%29.pdf. [Accessed January 2013].

[17] u-blox, "u-blox 6 GPS Modules Data Sheet," [Online]. Available: http://www.u-blox.com/images/downloads/Product_Docs/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf. [Accessed January 2013].

[18] "GSM: Global System for Mobile Communications," [Online]. Available: http://www.4gamericas.org/index.cfm?fuseaction=page&sectionid=242. [Accessed January 2014].

[19] "GPRS and EDGE Multislot Classes," [Online]. Available: http://mobilesociety.typepad.com/mobile_life/2007/04/gprs_and_edge_m.html. [Accessed October 2014].

[20] "GPRS & EDGE," [Online]. Available: http://www.3gpp.org/technologies/keywords-acronyms/102-gprs-edge. [Accessed September 2014].

[21] "GPRS General Packet Radio Service Tutorial," [Online]. Available: http://www.radio-electronics.com/info/cellulartelecomms/gprs/gprs_tutorial.php. [Accessed September 2014].

[22] I. Poole, "GSM EDGE Tutorial," [Online]. Available: http://www.radio-electronics.com/info/cellulartelecomms/gsm-edge/basics-tutorial-technology.php. [Accessed January 2014].

[23] T. Instrument, "MSP430x2xx Family User's Guide," 2013.

[24] K. Furset and P. Hoffman, "High pulse drain impact on CR2032 coin cell battery capacity," Nordic Semiconductor and Energizer, 2011 .

[25] P. R. Panda, B. V. N. Silpa, A. Shrivastava and K. Gummidipudi, Power-efficient System Design, London: Spribger, 2010.

[26] R. Oshana and M. Kraeling , Software Engineering for Embedded Systems, Elsevier, 2013.

[27] B. Ivey, "Low-Power Design Guide," Microchip Technology Inc., 2011.

[28] P. Padmanabhan and K. G. Shin, "Real-time dynamic voltage scaling for low-power embedded operating systems," in *ACM Symposium on Operating Systems Principles*, Chateau Lake Louise, 2001.

[29] K. Dudacek and V. Vavricka, "Experimental Evaluation of the MSP430 Microcontroller Power Requirements," in *EUROCON 2007 The International Conference on "Computer as a Tool"*, Warsaw, 2007.

[30] D. Snowdon, S. Ruocco and G. Heiser, "Power Management and Dynamic Voltage Scaling: Myths and Facts," National ICT Australia, Australia, 2005.

[31] D. Shin, S. N. U. S. K. Sch. of Comput. Sci. & Eng., J. Kim and S. Lee, "Low-energy intra-task voltage scheduling using static timing analysis," in *Design Automation Conference, 2001. Proceedings* , 2001.

[32] T. L. Martin, "Balancing Batteries, Power, and Performance: System Issues in CPU Speed-Setting for Mobile Computing. PhD thesis," Carnegie Mellon University, Pittsburgh,

2001.

[33] J. Flinn, K. Farkas and J. Anderson, "Power and energy characterization of the ItSY pocket computer (version 1.5). Technical," Western Research Laboratory, 2000.

[34] D. Grunwald, P. Levis, K. I. Farkas, C. B. Morrey and M. Neufeld, "Policies for dynamic clock scheduling," in *In Proceedings of the 4th Symposium on Operating Systems Design and Implementation (OSDI)*, 2000.

[35] M. Weiser, B. Wlech, A. Demers and S. Shenker, "Scheduling for Reduced CPU Energy," in *Operating Systems Design and Implementation*, Palo Alto, 1994.

[36] A. Weissel and F. Bellosa, "Process Cruise Control: Event-Driven Clock Scaling for Dynamic Power Management," in *Architecture, and Synthesis for Embedded Systems (CASES 2002)*, Grenoble, 2002.

[37] J. Yang, D. B. Munturn and F. Hady, "When Poll is Better than Interrupt," Intel Corporation.

[38] G. M. D. a. R. E. Richton, "Geolocation and Assisted GPS," in *Bell Laboratories, Lucent Technologies*, 2002.

[39] "EDGE: Enhanced Data Rates for GSM Evolution," [Online]. Available: http://www.4gamericas.org/index.cfm?fuseaction=page&sectionid=244. [Accessed January 2014].