System architecture for an intelligent implantable bio-telemetry device

Karthik Kumar Nattamai Kumaresan
University of Nevada, Las Vegas

Repository Citation
http://digitalscholarship.unlv.edu/thesesdissertations/1173

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Theses, Dissertations, Professional Papers, and Capstones by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
SYSTEM ARCHITECTURE FOR AN INTELLIGENT IMPLANTABLE BIO-TELEMETRY DEVICE

by

Karthik Kumar Nattamai Kumaresan

Bachelor of Science in Engineering
Anna University, India
2006

A thesis submitted in partial fulfillment
of the requirement for the

Masters of Science Degree in Electrical Engineering
Department of Electrical and Computer Engineering
Howard R. Hughes College of Engineering

Graduate College
University of Nevada, Las Vegas
April 2009
Thesis Approval
The Graduate College
University of Nevada, Las Vegas

April 15, 2009

The Thesis prepared by

Karthik Kumar Nattamai Kumaresan

Entitled

System Architecture for An Intelligent Implantable Bio-Telemetry Device

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering

Examination Committee Chair

Dean of the Graduate College

Graduate College Faculty Representative
ABSTRACT

System Architecture for an Intelligent Implantable Biotelemetry Device

by

Karthik Kumar Nattamai Kumaresan

Dr. Biswajit Das, Examination Committee Chair
Professor of Electrical and Computer Engineering
University of Nevada, Las Vegas.

Biotelemetry has long been used for environmental and life science research to study animal populations and behavior. The use of implantable bio-telemetric techniques makes it possible to record and study physiological variables during long-term experiments with a minimum disturbance to the animal. Fully implantable telemetric techniques greatly reduce the risk of infection associated with leads and catheters protruding from the skin. In this research the design and implementation of a completely programmable bio-implantable digital system which can measure two physiological signals extended over a period of time is considered. The proposed system consists of a standalone implantable transmitter unit and a receiving base station unit. The transmitter unit measures the physiological parameter converts it to an 8-bit digital data, sends it to the inbuilt Bluetooth transceiver which then wirelessly transmits the digital data to the base station. The system utilizes the power intelligently by turning on only when needed, the rest of the time it goes to sleep mode. The biotelemetry system proposed is simple, flexible and reliable, provides accurate, continuous measurement of physiological parameters of small
freely moving laboratory animals such as mice, rats or rabbits. The absence of restraints during the collection of physiological data allows studying animals with minimal stress during a long period of time in their normal housing.
# TABLE OF CONTENTS

ABSTRACT................................................................................................. iii

LIST OF FIGURES...................................................................................... vii

LIST OF TABLES........................................................................................ viii

ACKNOWLEDGEMENTS.......................................................................... ix

CHAPTER 1 INTRODUCTION........................................................................ 1
  1.1 Need for the Current Work................................................................. 1
  1.2 Review of the Past research............................................................ 3
  1.3 Intelligent Implantable System Overview........................................ 5
  1.4 Research Objectives and Organization of Thesis............................. 6

CHAPTER 2 SYSTEM DESIGN................................................................. 8
  2.1 Implantable System Design Considerations..................................... 8
  2.2 Components Selection..................................................................... 9
    2.2.1 DS2422 1-wire Temperature/Data logger................................. 10
    2.2.1.1 Memory and Control............................................................ 10
    2.2.1.2 Datalog Memory Usage........................................................ 12
    2.2.2 PIC16F88................................................................................. 15
    2.2.3 KC 22................................................................................... 16

CHAPTER 3 MATERIALS AND METHODS............................................. 19
  3.1 PIC16F88 to DS2422 Communication............................................. 19
    3.1.1 Hardware Configuration........................................................... 19
    3.1.2 Missioning DS2422................................................................. 20
      3.1.2.1 Time and Date................................................................. 22
      3.1.2.2 Sample Rate................................................................. 23
      3.1.2.3 Alarm Thresholds......................................................... 23
      3.1.2.4 Alarm Controls............................................................... 24
      3.1.2.5 General Mission Controls............................................ 24
      3.1.2.6 Temperature Conversions........................................... 25
    3.1.3 1-Wire Signaling................................................................. 26
    3.1.4 Missioning through 1-Wire Protocol..................................... 28
      3.1.4.1 Initialization................................................................. 29
      3.1.4.2 Rom Function Command............................................. 30
      3.1.4.3 Memory/Control Function Command............................ 30
      3.1.4.4 Transaction Data............................................................ 31
  3.2 PIC16F88 TO KC22 Communication........................................... 35
## Table of Contents

1. **3.2.1 Hardware Configuration** .......................................................... 35
2. **3.2.2 Firmware Interface** ................................................................. 36
3. **3.2.3 Modes of Operation** ................................................................. 36
4. **3.2.4 Data Transmission** ................................................................. 37
5. **3.3 User Interface** ............................................................................ 38

**CHAPTER 4 BIO IMPLANTABLE SYSTEM TEST RESULTS** .................. 41
1. **4.1 Temperature Test** ........................................................................ 41
2. **4.2 Distance Test** .............................................................................. 43
3. **4.3 Meat Test** .................................................................................... 44

**CHAPTER 5 CONCLUSION AND RECOMMENDATION** ....................... 47
1. **5.1 Conclusion** .................................................................................. 47
2. **5.2 Suggestions for Further Research** .............................................. 48
   1. **5.2.1 Low Power Consumption** ..................................................... 48
   2. **5.2.2 Induction charging** .............................................................. 48
   3. **5.2.3 Implantable rechargeable battery** ......................................... 48
   4. **5.2.4 Fully Integrated System** ....................................................... 49
   5. **5.2.5 Biomedical Issues** ............................................................... 49

**BIBLIOGRAPHY** .................................................................................. 50

**APPENDIX** ......................................................................................... 52
1. **1. DS2422 Datasheet** ...................................................................... 52
2. **2. PIC16F88 Datasheet** .................................................................. 54
3. **3. KC22 Datasheet** .......................................................................... 55
4. **4. PIC16F88 Microcontroller code** .................................................. 57

**VITA** .................................................................................................... 57
<p>| Figure 1.1 | Intelligent Implantable System architecture | 6 |
| Figure 1.1 | Sensor Unit | 7 |
| Figure 2.1 | DS2422 Block diagram | 11 |
| Figure 2.2 | One Channel logging | 14 |
| Figure 2.3 | Two Channel Logging, Equal Resolution | 14 |
| Figure 2.4 | Two Channel Logging Different Resolution | 15 |
| Figure 2.5 | PIC16F88 Microcontroller pin diagram | 16 |
| Figure 2.6 | KC22 Physical dimensions | 17 |
| Figure 3.1 | PIC16F88 - DS2422 Hardware interface | 20 |
| Figure 3.2 | 1-Wire Timing Diagram | 28 |
| Figure 3.3 | Initialization Waveform | 29 |
| Figure 3.4 | 64 bit Lasered ROM | 30 |
| Figure 3.5 | PIC 16F88 – KC22 Communication | 35 |
| Figure 3.6 | KC22 to PIC16F88 USART connection | 36 |
| Figure 3.7 | User Interface created using C# for the Bio-implantable System | 40 |
| Figure 4.1 | Bio-Implantable System Layout Design | 41 |
| Figure 4.2 | Temperature Test Setup | 42 |
| Figure 4.3 | Temperature Comparison Graph | 43 |
| Figure 4.4 | Sealed Prototype Board placed near the meat | 44 |
| Figure 4.5 | Prototype Board placed inside the Meat | 45 |
| Figure 4.6 | Graph showing the Meat Test Result | 46 |</p>
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 2.1</td>
<td>DS2422 Register Pages Map</td>
<td>13</td>
</tr>
<tr>
<td>Table 3.1</td>
<td>Real Time Clock and RTC Alarm register Bitmap</td>
<td>22</td>
</tr>
<tr>
<td>Table 3.2</td>
<td>Sample Rate Register</td>
<td>23</td>
</tr>
<tr>
<td>Table 3.3</td>
<td>Temperature Alarm Threshold registers</td>
<td>24</td>
</tr>
<tr>
<td>Table 3.4</td>
<td>Alarm Control Registers</td>
<td>24</td>
</tr>
<tr>
<td>Table 3.5</td>
<td>General Mission Control Register</td>
<td>25</td>
</tr>
<tr>
<td>Table 3.6</td>
<td>Latest Temperature Conversion Result Register Bitmap</td>
<td>26</td>
</tr>
<tr>
<td>Table 3.7</td>
<td>1-Wire Operation</td>
<td>27</td>
</tr>
<tr>
<td>Table 3.8</td>
<td>Protocol to read ROM</td>
<td>31</td>
</tr>
<tr>
<td>Table 3.9</td>
<td>Protocol to send Memory/ Control Function command</td>
<td>32</td>
</tr>
<tr>
<td>Table 3.10</td>
<td>Protocol to send Memory/ Control Function command continuation</td>
<td>33</td>
</tr>
<tr>
<td>Table 3.11</td>
<td>Protocol for Transaction/Data</td>
<td>34</td>
</tr>
<tr>
<td>Table 3.12</td>
<td>Transaction sequence</td>
<td>39</td>
</tr>
<tr>
<td>Table 4.1</td>
<td>Meat Test Results</td>
<td>46</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Dr. Biswajit Das who served as my advisor and guided me through the entire course of study. This thesis would have been beyond my capability without his countless helps and cooperation. It has been a great experience and fun working with him in both the research and the classes.

I am grateful to the Department of Electrical and Computer Engineering and its chair for the financial assistantship and other support they provided during my studies. I am also grateful to Dr. Venkatesan Muthukumar, Dr. Yingtao Jiang and Dr. Frank Van Breukelen for kindly accepting to serve in my graduate committee.

I also take this opportunity to thank everyone who directly and indirectly helped me complete this thesis in time. I am grateful to my friends Priyank, Arjun and Subbu for their understanding and encouragement at the time, when it was most needed. Finally, I would like to give special thanks to my family for their continuing love and affection throughout my life and career.
CHAPTER 1

INTRODUCTION

1.1 Need for the Current Work

Biotelemetry is a practical and efficient way for long-time monitoring of various physiological signals. For this reason, biotelemetry systems have been of high interest in various fields of biomedical engineering. Biotelemetry systems have been used for telemetry of physiological signals including ECG, EEG, ultrasonic Doppler signals. A telemetry system employing digital data transmission and processing, which makes the use of digital computers possible, has many advantages because computers are very useful for collection, processing and visualization of digital physiological data.

The use of telemetric techniques makes it possible to record and study physiological variables during long-term experiments with a minimum of disturbance to the animal. The absence of physical restraint and the ability to free range reduces stress and leads to more easily interpretable data. Fully implantable telemetric techniques greatly reduce the risk of infection associated with leads and catheters protruding from the skin. In addition, telemetric techniques make it possible to study animals in their natural habitat (ecophysiology), during social interactions and to correlate normal behaviors with physiological variables [14]. It is clear that many interpretations of previous physiological data are hampered by stress induced from confinement or human
interaction. This unwanted stress is greatly reduced by using a fully implantable telemetric system.

Although many bio-implantable systems have been developed over the years, these prior systems suffer from many problems [5][6][7]. Most of the bio-implantable systems available in the market are analog communication systems which uses frequency range from 10MHz to 450 MHz. There is lot of ways in which the analog signal can be distorted when they pass through a lossy medium. On the other hand Digital communication systems have various advantages over the analog communication such as less expensive, more reliable, easy to manipulate, flexible, compatibility with other digital systems, only digitized information can be transported through a noisy channel without degradation.

Most of the implantable systems in market require the user or the animal to wear a receiver belt or receiving antennas station which receives the signal from the implant, amplifies and transmits it to the base station. The receiver belt can cause unwanted stress on the animal; the communication can get disrupted if the receiver belt is damaged. Another major issue about these implants is they cannot be used over a long period of time (months) because of very limited power supply.

In this research we consider an alternate design and implementation of a completely programmable bio-implantable digital system which can be capable of simultaneous measurements of two physiological signals extended over periods of time. One important aspect is the longer transmission range of the implant and base station with a capacity for bidirectional communication within the room as compared to most commercially available systems. The system converts analog signals into 8-bit digital form and then
transmits these bits through a Bluetooth transmitter. Although it is based on standard electronic components which are very cheap and can be found everywhere, it offers very high performance within a distance of up to 20 feet, which is more than sufficient in almost all cases.

1.2 Review of the Past Research

The first successful transmission of biological information from a living animal was performed in 1869 by Marey [1]. This was followed by Einthoven’s experiment in 1903 where he used a telephone line to transmit ECG data over a 1.5 km distance [2]. If we adopt a modern definition of biotelemetry as “measurements from an unrestrained animal or patient using radio links” [3], Winters may be the inventor of telemetry in 1921 [4], followed by Fuller & Gordon in 1948 [5]. A breakthrough in the design of telemetric and electronic equipment came in 1952 with the introduction of transistor technology. Radio transmitters attached outside the body provided some of the early tools for monitoring of unrestrained animals. The development of the transistor made possible the design of circuits sufficiently small and low in power consumption for implantation within the body. Mackay and Jacobson [6] reported on the implanted devices in 1957, coining the word “endoradiosonde”. The ready availability of transistors simulated the development of a wide variety of telemetry systems [7].

This was followed by the development of integrated circuits that was critical in the development of compact, portable, implantable devices. With these new techniques, fabrication of a variety of bio-implantable devices was made practical. In the area of telemetry powered implantable devices, Loucks, Chaffee and Light in 1935 [8]
independently investigated the possibility of transferring power and data to an implanted device via an inductively coupled transformer link, which greatly reduced the infection and discomfort.

During the last decade, significant work has been done in the development of RF implantable electronics to stimulate the nerves and retinas of those with certain diseases. In 1998, Von Arx presented a fully integrated neuromuscular stimulation system, called FINESS, used to stimulate nerves in patients with paralysis. Von Arx used a wireless system, which eliminated leads but it tend to break off or damage nerves when implanted [9]. Mark Clements (1999) at North Carolina State University described a retinal stimulator being developed with the Wilmer Eye Institute at John Hopkins University. The idea is to restore eyesight lost due to macular degeneration or retinal pigmentosa, two common conditions that cause failure in the rods and cones that act as the eye's photoreceptors. The system consists of an extraocular unit [11] and an intraocular unit that are coupled together via a wireless inductive telemetry link. The extraocular unit provides for image acquisition and processing and consists therefore of an external camera, image processor, telemetry encoder, RF amplifier and a primary coil. The intraocular unit receives power and data from the extraocular unit [11].

The physicists and engineers seeking information on the possible biological effects of non-ionizing electric and magnetic (EM) fields face a difficult task. EM fields appear to have emerged as a new public health issue in 1979 with publication of Wertheimer and Leeper's study of childhood cancer in relation to power distribution line proximity [12]. However, concerns about possible health effects of microwave and radiofrequency
energy go back to the Second World War and before, and reports of health effects appear in the Soviet and Eastern European literature in the 1950s and 1960s.

The word "radiation" refers to the fact that energy can radiate. It has not been shown conclusively that microwaves (or other non ionizing electromagnetic radiation) have significant adverse biological effects at low levels [13]. This is separate from the risks associated with very high intensity exposure, which can cause heating and burns like any heat source, and not a unique property of microwaves specifically.

1.3. Intelligent Implantable System Overview

A simplified block diagram of the Intelligent Implantable system is shown in Figure 1. The system has a transmitter unit which wakes up when turned on, reads the physiological signal converts it into 8 bit digital data, transmits the data wirelessly to the receiver host station and goes to sleep mode. The host station receives the digital data and displays it onto the screen using the visual interface created.

The sensor unit is a completely programmable and a low power consumption device, it consists of a physiological sensor, A/D convertor, real time clock, memory and a pump or an external circuit as shown in the Figure 1.2. When the sensor gets a call signal from the microcontroller, it wakes up, starts a mission, collects data, converts it into a digital form, stores the 8bit data into its memory and then goes to sleep mode again. The low power microcontroller is a programmable device which acts as a brain of the implant system. It tells the sensor device when to wake up, when to start a mission, reads the stored data from the sensor memory after the mission, stores the data read into its
Figure 1.1. Intelligent Implantable System architecture

EEPROM and sends the stored data to the RF transceiver at user desired time. The received 8-bit digital data is then transmitted wirelessly to the receiver host station by the RF transceiver. The RF transceiver is made active by the microcontroller only when it transmits or receives data, the rest of the time it goes into sleep mode, thus saving power.

1.4 Research Objectives and Organization of Thesis

This thesis describes the development of a circuitry for an intelligent wireless bio implantable system. The aim is to design and implement a stand alone fully integrated system that can wirelessly transmit information from inside the body of an animal to a host station outside the animal. This research aims at overcoming some of the limitations
and shortcomings of the present Bio implantable systems that use wires protruding out of
the body or need constant external transmitter. In particular the goals of this thesis are:

- Develop a circuitry for a fully implantable battery powered system which
  can transmit information from inside the body to outside.

- To design a stand alone system that is completely programmable, keeping
  in mind the design considerations of a bio implantable system.

- To make the circuit design flexible so that it can be adopted for other
  applications with simple modifications.

Chapter 2 of this thesis gives the general background of the system design, Chapter 3
deals with the architecture and the working of the implantable system followed by the
ircuitry, Chapter 4 explains various test results obtained from the implemented
implantable system. Chapter 5 gives conclusions and suggestions for further work.
CHAPTER 2

SYSTEM DESIGN

Before designing the bio-implantable system, there are various considerations that have to be taken into account. This chapter explains the design considerations for an implantable device, how devices are selected and a brief study of the selected devices is done.

2.1 Implantable System Design Considerations

A wholly implanted telemetry system must be so small that the animal is not disturbed physiologically by it. Since small animals, such as rats, rabbits and monkeys, are most frequently used for laboratory experiments, the device must be quite small. Because of costs and convenience, large animals are seldom used in laboratory and are even less suited to experiments in space.

Low consumption of power is another requirement in design; it contributes to smallness in size since batteries are the largest single component in a miniature telemetry system, and it is essential for long-term operation. The batteries of the external telemetry system can be changed periodically, but not those of an implanted device. The required life of the battery is further lengthened by period of several weeks required for covalence of animal between surgical implantation and the yield of significant data; during convalescence, of course the battery cannot be replaced. The operating life of the systems
has to be from 1 month to 2 years with a single battery- ample time for both recovery from surgery and long term experiment. Hence the implant has to be an intelligent system that utilizes power in a very efficient way.

Most of the life science experiments are done in laboratories where the animals are inside the cage. On an average the laboratories are around (long and wide) dimensions, thus the implant has to transmit the data to a host station which is placed at least 5 feet away from the cage. The host station can then receive the data process it and use internet, WIFI or any other means of communication to transmit the data further to a network or a personal computer which is far apart.

The animal body is made up tissues, bones blood vessels and other biological materials. Since the bio implantable system has to transmit data wirelessly from inside the body, the radio frequency effects on the body of has to be considered while selecting the Radio frequency range of the wireless transmitter.

There are various physiological signals like temperature, ECG, EEG, blood flow, pressure etc that can be measured from a human body using the sensors that are readily available in the market. The system should have the flexibility to measure more than one physiological system with easy adjustments. These are the considerations taken into account before designing the circuit.

2.2 Component Selection

Depending on the discussed design considerations for an intelligent bio implantable system, a wide search for programmable sensors, microcontrollers and wireless
transceivers was done. Finally the following parts which suited the circuitry design of the system were selected.

2.2.1. DS2422 1-Wire Temperature/Data logger

The DS2422 is a temperature/data logger that combines the core functions of a fully featured data logger in a single chip. It measures the temperature and/or reads the ADC at a user-defined rate. Additionally, the DS2422 can store 8192 8-bit readings or 4096 16-bit readings taken at equidistant intervals ranging from 1 second up to 273 hours. A mission to collect data can be programmed to begin immediately, after a user defined delay, or after a temperature alarm. DS2422 is factory-lasered with a guaranteed unique 64-bit registration number that allows for absolute traceability.

The relationships between the major control and memory sections of the DS2422 are shown in the block diagram in Figure2.1. The device has six main data Components: 64-bit lasered ROM, 256-bit scratchpad, 512-byte general-purpose SRAM, Two 256-bit register pages of timekeeping, control, status, and counter registers and passwords, 64 bytes of calibration memory and 8192 bytes of data-logging memory.

Except for the ROM and the scratchpad, all other memory is arranged in a single linear address space. The data-logging memory, counter registers and several other registers are read-only for the user. Both register pages are write-protected while the device is programmed for a mission. The password registers; one for a read password and another one for a read/write password can only be written to, never read.

2.2.1.1. Memory and Control

The memory map of the DS2422 is shown in Table2.1. The 512 bytes general-purpose SRAM are located in pages 0 through 15. The various registers to set up and
control the device fill page 16 and 17, called Register Pages 1 and 2 (details in Figure 2.1). Pages 18 and 19 provide storage space for calibration data. They can alternatively

be used as extension of the general-purpose memory. The "data log" logging memory starts at address 1000h (page 128) and extends over 256 pages. The memory pages 20 to

Figure 2.1 DS2422 block diagram
31 and 33 to 127 are reserved for future extensions. The scratchpad is an additional page that acts as a buffer when writing to the SRAM memory or the register page. The data and calibration memory can be written at any time. The access type for the two register pages and the Trim Register Page is register-specific and depends on whether the device is programmed for a mission. Figure 5 shows the details. The data log memory is read-only for the user. It is written solely under supervision of the on-chip control logic. Due to the special behavior of the write access logic (write scratchpad, copy scratchpad) it is recommended to only write full pages at a time. This also applies to all the register pages and the calibration memory.

2.2.1.2 Data-log Memory Usage

Once setup for a mission, DS2422 starts to log temperature. The datalog memory is able to store 8192 entries in 8-bit format or 4096 entries in 16-bit format (Figure 2.3). If temperature as well as external data is logged, both in the same format, the memory is split into two equal sections that can store 4096 8-bit entries or 2048 16-bit entries (Figure 2.4). If the device is set up to log data in different formats, e.g., temperature in 8-bit and external data in 16-bit format, the memory is split into blocks of different size, accommodating 2560 entries for either data source (Figure 2.5). In this case, the upper 256 bytes are not used. In 16-bit format, the higher 8 bits of an entry are stored at the DS2422’s 23 of 48 lower addresses. Knowing the starting time point (Mission Time Stamp) and the interval between temperature measurements one can reconstruct the time and date of each measurement. There are two alternatives to the way the DS2422 behaves after the datalog memory is filled with data. The user can program the device to either stop any further recording (disable rollover) or overwrite the previously recorded data.
(enable rollover), one entry at a time, starting again at the beginning of the respective memory section. The contents of the mission

Table 2.1 DS2422 Register Pages Map

<table>
<thead>
<tr>
<th>ADDR</th>
<th>b7</th>
<th>b6</th>
<th>b5</th>
<th>b4</th>
<th>b3</th>
<th>b2</th>
<th>b1</th>
<th>b0</th>
<th>Function</th>
<th>Access*</th>
</tr>
</thead>
<tbody>
<tr>
<td>0200h</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10 Seconds</td>
<td>R/W; R</td>
</tr>
<tr>
<td>0201h</td>
<td>0</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10 Minutes</td>
<td>R/W; R</td>
</tr>
<tr>
<td>0202h</td>
<td>0</td>
<td>12/24</td>
<td>20h.</td>
<td>AM/PM</td>
<td>10h.</td>
<td></td>
<td></td>
<td></td>
<td>Single Hours</td>
<td>R; R</td>
</tr>
<tr>
<td>0203h</td>
<td>0</td>
<td>0</td>
<td>10 Date</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Single Date</td>
<td>R; R</td>
</tr>
<tr>
<td>0204h</td>
<td>CENT</td>
<td>0</td>
<td>0</td>
<td>10m.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Single Months</td>
<td>R; R</td>
</tr>
<tr>
<td>0205h</td>
<td>10 Years</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Single Years</td>
<td>R; R</td>
</tr>
<tr>
<td>0206h</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Latest Temp.</td>
<td>R; R</td>
</tr>
<tr>
<td>0207h</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Latest Data</td>
<td>R; R</td>
</tr>
<tr>
<td>0208h</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Low Byte</td>
<td>R; R</td>
</tr>
<tr>
<td>0209h</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>High Byte</td>
<td>R; R</td>
</tr>
<tr>
<td>020Ah</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Low Threshold</td>
<td>R; R</td>
</tr>
<tr>
<td>020Bh</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>High Threshold</td>
<td>R; R</td>
</tr>
<tr>
<td>020Ch</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Temp.</td>
<td>R; R</td>
</tr>
<tr>
<td>020Dh</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Data Alarms</td>
<td>R; R</td>
</tr>
<tr>
<td>020Eh</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Latest Temp.</td>
<td>R; R</td>
</tr>
<tr>
<td>020Fh</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Latest Data</td>
<td>R; R</td>
</tr>
</tbody>
</table>

Sample counter in conjunction with the sample rate and the Mission Time Stamp will then allow reconstructing the time points of all values stored in the datalog memory. This
gives the exact history over time for the most recent measurements taken. Earlier measurements cannot be reconstructed.

Figure 2.2 One Channel Logging

ETL = 1; EDL = 0 or ETL = 0; EDL = 1
TLFS = DLFS = 0

ETL = 1; EDL = 0 or ETL = 0; EDL = 1
TLFS = DLFS = 1

8192 8-bit entries
Temperature or External data

4096 16-bit entries
Temperature or External data

1000h → 2FFFh

With 16-bit format, the most-significant byte is stored at the lower address.

Figure 2.3 Two-Channel Logging, Equal Resolution

ETL = EDL = 1
TLFS = DLFS = 0

ETL = EDL = 1
TLFS = DLFS = 1

Temperature
4096 8-bit entries

External Data
4096 8-bit entries

Temperature
2048 16-bit entries

External Data
2048 16-bit entries

1000h → 1FFFh → 2000h → 2FFFh

1000h → 1FFFh → 2000h → 2FFFh

With 16-bit format, the most-significant byte is stored at the lower address.
ETL = EDL = 1
TLFS = 0; DLFS = 1

Temperature
2560
8-bit entries
1000h
19FFh

External Data
2560
16-bit entries
1A00h
2DFFh

(not used)
2E00h
2FFFh

ETL = EDL = 1
TLFS = 1; DLFS = 0

Temperature
2560
16-bit entries
1000h
23FFh

External Data
2560
8-bit entries
2400h
2DFFh

(not used)
2E00h
2FFFh

With 16-bit format, the most-significant byte is stored at the lower address.

Fig 2.4 Two-Channel Logging, Different Resolution

2.2.2 PIC16F88

PIC16F88 microcontroller (Figure 2.6) is a powerful (200 nanosecond instruction execution), easy-to-program (only 35 single word instructions) CMOS Flash-based 8-bit microcontroller. The PIC16F88 features 8 MHz internal oscillator, 256 bytes of EEPROM data memory, a capture/compare/PWM, an Addressable USART, a synchronous serial port that can be configured as either 3-wire Serial Peripheral Interface (SPI™). The special features of the microcontroller are listed below.

- Special Microcontroller Features
- 100,000 erase/write cycle Enhanced Flash Program memory
- 1,000,000 typical erase/write cycles EEPROM data memory
- EEPROM Data Retention: > 40 years
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Processor read/write access to program memory
- Low-Voltage Programming
- In-Circuit Debugging via two pins
- Extended Watchdog Timer (WDT):
  - Programmable period from 1ms to 268s
- Wide operating voltage range: 2.0V to 5.5V

![PIC 16F88 Microcontroller pin diagram](image)

**Figure 2.5 PIC 16F88 Microcontroller pin diagram**

### 2.2.3 Kc22

Kc22 is one of the smallest Bluetooth modules available. The Kc22 Bluetooth OEM Micro Module (figure 2.7) is designed for maximum performance in a minimal space. The Kc-22 module includes 4 programmable input/output lines, and offers high speed serial communications up to 921Kbaud.
Kc22 is a cable replacement application that provides point-to-point communication between two Bluetooth devices. The Kc22 is a pre-engineered and pre-licensed PCB module that provides fully embedded, ready to use Bluetooth wireless technology. Multi-surface pads provide both bottom pads for high volume reflow soldering and edge pads for low volume hand soldering. The reprogrammable flash memory contains embedded firmware for serial cable replacement deploying the Bluetooth Serial Port Profile (SPP). Custom firmware can be pre-loaded into these highly tuned and tested modules so that they are ready to install without additional procedures. Some of the other features of Kc22 listed below.

- **Serial Port Profile** – SPP is supported with KcSerial for both Client and Server application.
• Command and Bypass modes – it is possible to switch between Command and Bypass (data transmit/receive) modes during an active connection

• Security – Bonding and data encryption provides a secure link between two devices.

• Multiple Device Bonding – special security keys can be exchanged with multiple devices to allow different devices to securely connect with KcSerial (although not simultaneously)

• Power conservation – use of Park, Sniff, and Hold features to minimize power consumption

• Variable Baud Rates – the serial port can be configured for the following baud rates: 9600, 19.2K, 38.4K, and 57.6K.
CHAPTER 3

MATERIALS AND METHODS

3.1 PIC16F88 to DS2422 Communication

This chapter gives a brief explanation about the communication between PIC16F88, DS2422 and KC22 devices, both the hardware configurations and logic programmed in the microcontroller is discussed.

3.1.1 Hardware Configuration

The 16f88 PIC microcontroller has multiple general purpose input/output (GPIO). Pin RB0 was configured to communicate with the DS2422 through the serial 1-Wire protocol. The PIC 16F88 microcontroller initiates the communication, acts as the bus master and drives the DS2422 slave. The 1-wire protocol is so called because it requires only a single data lead and a ground return (Figure3.1.). As DS2422 provides a relatively small amount of data, the typical data rate of 16 kbps is sufficient for the intended tasks. The Idle state for the 1-Wire bus is high. If, for any reason, a transaction needs to be suspended, the bus must be left in the idle state.

At standard speed the 1-Wire bus has a maximum data rate of 16.3kbps. The speed can be boosted to 142kbps by activating the Overdrive mode. The maximum data rate for DS2422 in standard speed mode is 15.4kbps and 125kbps in Overdrive. The DS2422 was connected to the PIC16F88 micro controller as shown in Figure3.1. The value of the
pull up resistor primarily depends on the network size and load conditions. The DS2422 requires a pull-up resistor of maximum 2.2kΩ at any speed, the pull up resistor of 220 ohm was used to connect to VCC. Before writing the algorithm and code for the one wire protocol it is necessary to understand the signals used for the communication through 1-wire protocol.

3.1.2 Missioning DS2422

The typical task of the DS2422 is recording temperature and/or external data. Before the device can perform this function, it needs to be set up properly. This procedure is called missioning. First of all, DS2422 needs to have its RTC set to valid time and date. This reference time may be the local time, or when used inside of a mobile unit, UTC (also called GMT, Greenwich Mean Time) or any other time standard that was agreed upon. The RTC oscillator must be running (EOSC = 1). The memory assigned to store the Mission Time Stamp, Mission Samples Counter, Sample Rate, and Alarm Flags must be cleared using the Memory Clear command. To enable the device for a mission, at least
one of the enable logging bits needs to be set to 1. These are general settings that have to be made in any case, regardless of the type of object to be monitored and the duration of the mission.

If alarm signaling is desired, the temperature alarm and/or data alarm low and high thresholds must be defined. How to convert a temperature value into the binary code to be written to the threshold registers is described under Temperature Conversion section 3.2.6 in this document. The setting of the RO bit (rollover enable) and sample rate depends on the duration of the mission and the monitoring requirements. If the most recently logged data is important, the rollover should be enabled (RO = 1). Otherwise one should estimate the duration of the mission in minutes and divide the number by 8192 (single channel 8-bit format) or 4096 (single channel 16-bit format, two channels 8-bit format) or 2048 (two channels 16-bit format) or 2560 (two channels, one 8-bit format and one 16-bit format) to calculate the value of the sample rate (number of minutes between temperature conversions). If the estimated duration of a mission is 10 days (= 14400 minutes), for example, then the 8192-byte capacity of the datalog memory would be sufficient to store a new 8-bit value every 1.8 minutes (110 seconds).

If the datalog memory of the DS2422 is not large enough to store all readings, one can use several devices and set the Mission Start Delay to values that make the second device start logging as soon as the memory of the first device is full, and so on. The RO-bit needs to be set to 0 to disable rollover that would otherwise overwrite the logged data. After the RO bit and the Mission Start Delay are set, the sample rate needs to be written to the Sample Rate Register. The sample rate may be any value from 1 to 16383, coded as an unsigned 14-bit binary number. A sample rate of all zeros is not valid and must be
avoided under all circumstances. This causes the device to enter into an undefined state, requiring a power-on reset and restore of the trim settings to recover. The fastest sample rate is one sample per second (EHSS = 1, Sample Rate = 0001h) and the slowest is one sample every 273.05 hours (EHSS = 0, Sample Rate = 3FFFh). To get one sample every 6 minutes, for example, the sample rate value needs to be set to 6 (EHSS = 0) or 360 decimal (equivalent to 0168h at EHSS = 1).

3.1.2.1 Time and Date

The Real time clock/alarm and calendar information is accessed by reading/writing the appropriate bytes in the register page, address 0200h to 0205h. The number representation of the RTC registers is BCD format (binary-coded decimal). The following date and the time were set by writing the corresponding values to memory addresses as shown in the Table 3.1.

Table 3.1 Real-Time Clock and RTC Alarm Register Bitmap

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Values</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0200h</td>
<td>00</td>
<td>10:00:00</td>
<td>Time</td>
</tr>
<tr>
<td>0201h</td>
<td>00</td>
<td>10:00:00</td>
<td></td>
</tr>
<tr>
<td>0202h</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0203h</td>
<td>01</td>
<td>1st April of 2009</td>
<td>Date</td>
</tr>
<tr>
<td>0204h</td>
<td>04</td>
<td>1st April of 2009</td>
<td></td>
</tr>
<tr>
<td>0205h</td>
<td>09</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3.1.2.2 Sample Rate

The content of the Sample Rate Register (addresses 0206h, 0207h) specifies the time elapse (in seconds if EHSS = 1, or minutes if EHSS = 0) between two temperature/data logging events. The DS2422 is set to sample the temperature once in 5 seconds as shown in Table 3.2 below. (Refer DS2422 datasheet for more info)

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Values</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0206h</td>
<td>05h</td>
<td>Every 5 seconds</td>
<td>Sample rate</td>
</tr>
<tr>
<td>0207h</td>
<td>00h</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0212h</td>
<td>03h</td>
<td>Sample rate in seconds</td>
<td>RTC Oscillator Control, sample rate selection</td>
</tr>
</tbody>
</table>

3.1.2.3 Alarm Thresholds

The DS2422 has two Temperature Alarm Threshold registers (address 0208h, 0209h) to store values, which determine whether a critical temperature has been reached (Table 3.3). A temperature alarm is generated if the device measures an alarming temperature and the alarm signaling is enabled. The low alarm threshold was set to xx°C and the high alarm threshold was set to xxC as shown below in the table. The temperature alarm is not activated and hence any values can be entered in for xx values in Table 3.3.
Table 3.3 Temperature Alarm Threshold registers

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Values</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0208h</td>
<td>xx</td>
<td>xx</td>
<td>Temperature Alarm</td>
</tr>
<tr>
<td>0209h</td>
<td>xx</td>
<td>xx</td>
<td>Threshold</td>
</tr>
<tr>
<td>020Ah</td>
<td>xx</td>
<td>xx</td>
<td>External Data Alarm</td>
</tr>
<tr>
<td>020Bh</td>
<td>xx</td>
<td>xx</td>
<td>Threshold</td>
</tr>
</tbody>
</table>

3.1.2.4 Alarm Controls

The bits ETLA and ETHA enable the temperature alarm are located in the Temperature Sensor Control Register (refer DS2422 data sheet). The temperature alarms are disabled as shown in the Table 3.4.

Table 3.4 Alarm Control Registers

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Values</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0210h</td>
<td>0h</td>
<td>Disabled</td>
<td>Temp. Alarm control</td>
</tr>
<tr>
<td>0211h</td>
<td>FC</td>
<td>Disabled</td>
<td>Data. Alarm control</td>
</tr>
</tbody>
</table>

3.1.2.5 General Mission Control

This register is used to set the general mission parameters e.g., channels to log and logging format, rollover and start mode. The mission was set for a normal start with rollover enabled and 8-bit temperature log. The corresponding hex value for it is D3h (refer
The following data will setup the DS2422 for a mission that logs temperature using 8-bit format as in Table 3.5.

Table 3.5 General Mission Control Register

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Values</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0213h</td>
<td>D3</td>
<td>Normal start, roll over enabled; 8bit temperature log</td>
<td>General Mission control</td>
</tr>
</tbody>
</table>

These are some other mission control registers that has not been discussed above which can set according to the user needs, for detailed information refer DS2422 datasheet provided my Maxim.

3.1.2.6 Temperature Conversion

The DS2422 can measure temperatures from -40°C to +85°C. Temperature values are represented as an 8- or 16-bit unsigned binary number with a resolution of 0.5°C in the 8-bit mode and 0.0625°C in the 16-bit mode. The higher temperature byte TRH is always valid. In the 16-bit mode only the three highest bits of the lower byte TRL are valid. The five lower bits all read zero. TRL is undefined if the device is in 8-bit temperature mode. An out of range temperature reading is indicated as 00h or 0000h when too cold and FFh or FFE0h when too hot. With TRH and TRL representing the decimal equivalent of a
temperature reading the temperature value is calculated as

$$\theta(°C) = \frac{TRH}{2} - 41 + \frac{TRL}{512} \text{ (16 bit mode, TLFS = 1, see address 0213h)}$$

$$\theta(°C) = \frac{TRH}{2} - 41 \text{ (8 bit mode, TLFS = 0, see address 0213h)}$$

This equation is valid for converting temperature readings stored in the datalog memory as well as for data read from the Latest Temperature Conversion Result Register.

3.1.3 1-Wire Signals

The four basic operations of a 1-Wire bus are Reset, Write 0 bit, Write 1 bit and Read bit. Using these bit operations, one has to derive a byte or a frame of bytes. The bus master initiates and controls all of the 1-Wire communication. Figure 3.2 illustrates the 1-Wire communication timing diagram, It is similar to Pulse-Width Modulation (PWM) because the data is transmitted by wide (logic ‘0’) and narrow (logic ‘1’) pulse widths during data bit time periods or time slots. The timing diagram also contains the recommended time values for robust communication across various line conditions. Table 3.6 provides a list of operations with descriptions and also implementation steps at standard speed.

A communication sequence starts when the bus master drives a defined length “Reset” pulse that synchronizes the entire bus. The slave responds to the “Reset” pulse with a logic-low “Presence” pulse. To write the data, the master first initiates a time slot by driving the 1-Wire line low, and then, either holds the line low (wide pulse) to transmit a logic ‘0’ or releases the line (short pulse) to allow the bus to return to the logic
'1' state. To read the data, the master again initiates a time slot by driving the line with a narrow low pulse. A slave can then either return a logic '0' by turning on its open-drain output and holding the line low to extend the pulse, or return a logic '1' by leaving its open-drain output off to allow the line to recover.

Table 3.7 1-Wire operation

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reset</td>
<td>Reset the 1-Wire bus slave device and get it ready for a command.</td>
<td>Drive bus low, delay 480 µs. Release bus, delay 70 µs. Sample bus: 0 = device(s) present, 1 = no device present Delay 410 µs</td>
</tr>
<tr>
<td>Write 0 bit</td>
<td>Send '0' bit to the 1-Wire slave (Write 0 slot time).</td>
<td>Drive bus low, delay 60 µs. Release bus, delay 10 µs.</td>
</tr>
<tr>
<td>Write 1 bit</td>
<td>Send '1' bit to the 1-Wire slave (Write 1 slot time).</td>
<td>Drive bus low, delay 6 µs. Release bus, delay 64 µs.</td>
</tr>
<tr>
<td>Read bit</td>
<td>Read a bit from the 1-Wire slave (Read time slot).</td>
<td>Drive bus low, delay 6 µs. Release bus, delay 9 µs. Sample bus to read bit from slave. Delay 55 µs.</td>
</tr>
</tbody>
</table>
3.1.4 Missioning DS2422 through 1-wire protocol

The protocol for accessing the DS2422 through the 1-Wire port is as follows:

1. Initialization
2. ROM Function Command
3. Memory/Control Function Command
4. Transaction/Data
3.1.4.1 Initialization

All transactions on the 1-Wire bus begin with an initialization sequence. The initialization sequence consists of a reset pulse transmitted by the bus master followed by presence pulse(s) transmitted by the slave(s). The presence pulse lets the bus master know that the DS2422 is on the bus and is ready to operate.

![Figure 3.3 Initialization Waveform](image)

A program to generate the wave functions as in the Figure 3.3 was coded using assembly language. The PIC 16F88 microcontroller sends a reset signal to DS2422 and waits for 480\(\mu\)s to receive a presence pulse. If it receives a presence pulse the next instruction is executed. If the PIC16F88 doesn’t receive a presence pulse a Reset pulse is sent again, this is done until a presence pulse is detected.
3.1.4.1.2 ROM Function Commands

Once the PIC16F88 master has detected a presence, it issues READ ROM function command (Table 3.7). The READ ROM function command is 8 bits long.

<table>
<thead>
<tr>
<th>MSB</th>
<th>LSB</th>
<th>MSB</th>
<th>LSB</th>
<th>MSB</th>
<th>LSB</th>
<th>MSB</th>
<th>LSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8-Bit CRC Code</td>
<td>48-Bit Serial Number</td>
<td>8-Bit Family Code (41h)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.4 64-bit lasered ROM

The corresponding hexadecimal value for the READ ROM command is 33h. This command allows the bus master to read the DS2422’s 8-bit family code, unique 48-bit serial number, and 8-bit CRC (Figure3.4). Once this is done the Bus master sends the memory/control function command to start the mission.

3.1.4.3 Memory/Control Function Command

Before starting a mission the previous data stored in the DS2422 must be cleared. The memory assigned to store the Mission Time Stamp, Mission Samples Counter, Sample Rate, and Alarm Flags is cleared using the Memory Clear command. The PIC16F88 microcontroller next sends the data to set the DS2422 for a mission. The 25 bytes of data is first written to the scratch pad and then copied to the page0 of the mission control register starting from the address 0200h. To check if the 25 bytes of data has been written to the scratch pad, the scratch pad is read and verified. Next the data in the scratch pad is copied to the mission control memory using the copy
Table 3.8 Protocol to read ROM

<table>
<thead>
<tr>
<th>Master mode</th>
<th>Data</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>33h</td>
<td>Issue “Read ROM” command</td>
</tr>
<tr>
<td>RX</td>
<td>64 bit serial number</td>
<td>Reads 64 bit serial number</td>
</tr>
</tbody>
</table>

scratch pad command. The table below clearly explains the sequence in which the commands are executed.

A reset signal and a Skip Rom command is always sent before the memory and function control commands is sent to DS2422. A mission is then started by sending the Start mission with password command. As soon as the DS2422 receives the start mission command and verifies the password, a mission is immediately started. A time delay to measure the temperature according to the user needs is given to DS2422 to measure the temperature at equal intervals of time as set in the mission control register.

3.1.4.4 Transaction/Data

After the mission delay expires, a Stop mission with password command is sent to DS2422 by the PIC16F88 bus master (Table 3.9). Once the DS2422 slave receives this command it verifies the password and abruptly stops the mission. If the password is incorrect the DS2422 skips the STOP mission command and waits for the next command.
Table 3.9 Protocol to send Memory/Control Function Command

<table>
<thead>
<tr>
<th>Master mode</th>
<th>Data</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TX</td>
<td>96h</td>
<td>Issue “clear memory” command</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;8FF bytes&gt;</td>
<td>Send dummy password</td>
</tr>
<tr>
<td>TX</td>
<td>FF</td>
<td>Send dummy password</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue “skip ROM” command</td>
</tr>
<tr>
<td>TX</td>
<td>0Fh</td>
<td>Issue “write scratchpad” command</td>
</tr>
<tr>
<td>TX</td>
<td>00h</td>
<td>TA1, beginning offset=00h</td>
</tr>
<tr>
<td>TX</td>
<td>02h</td>
<td>TA2, address=0200h</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;25 data bytes&gt;</td>
<td>Write 25 bytes of data to scratchpad</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;7FFh data bytes&gt;</td>
<td>Write through the end of the scratchpad</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue “skip ROM” command</td>
</tr>
<tr>
<td>TX</td>
<td>AAh</td>
<td>Issue “read scratchpad” command</td>
</tr>
<tr>
<td>RX</td>
<td>00h</td>
<td>Read TA1, beginning offset=00h</td>
</tr>
<tr>
<td>RX</td>
<td>02h</td>
<td>Read TA2, address=0200h</td>
</tr>
<tr>
<td>RX</td>
<td>1Fh</td>
<td>Read E/S, ending offset=1Fh, flags=0h</td>
</tr>
</tbody>
</table>
All the temperature measurements are saved in the datalog memory starting from the address 1000h during the mission. The data remains in the memory until a clear mission command is received. The data can be accessed by the microcontroller using Read memory with password and CRC followed by the starting address from which the data has to be read. DS24S22 allows reading the memory of one complete page at a time (32

![Table 3.10 Protocol to send Memory/Control Function Command continuation](image)

<table>
<thead>
<tr>
<th>Master mode</th>
<th>Data</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>RX</td>
<td>&lt;32 bytes of data&gt;</td>
<td>Read scratchpad data and verify</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue skip ROM command</td>
</tr>
<tr>
<td>TX</td>
<td>99h</td>
<td>Issue &quot;copy scratchpad command&quot;</td>
</tr>
<tr>
<td>TX</td>
<td>00h</td>
<td>TA1</td>
</tr>
<tr>
<td>TX</td>
<td>02h</td>
<td>TA2 (autorisation code)</td>
</tr>
<tr>
<td>TX</td>
<td>1Fh</td>
<td>E/S</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;8FF bytes&gt;</td>
<td>Send dummy password</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue skip ROM command</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue Start mission with password command</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;8FF bytes&gt;</td>
<td>Send dummy password</td>
</tr>
</tbody>
</table>
bytes). The data read from the DS2422’s memory is stored in the PIC 16F88’s EEPROM. Finally the memory has to be cleared for the future use of DS2422 for another mission as shown in Table 3.9.

Table 3.11 Protocol for Transaction/Data

<table>
<thead>
<tr>
<th>Master Mode</th>
<th>Data</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue “skip ROM” command</td>
</tr>
<tr>
<td>TX</td>
<td>33h</td>
<td>Stop mission with password</td>
</tr>
<tr>
<td>TX</td>
<td>&lt;8FF bytes&gt;</td>
<td>Send dummy password</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>CCh</td>
<td>Issue “skip ROM” command</td>
</tr>
<tr>
<td>TX</td>
<td>69h</td>
<td>Issue “Read memory with Password and CRC” command</td>
</tr>
<tr>
<td>TX</td>
<td>00h</td>
<td>Starting address of the memory to read from</td>
</tr>
<tr>
<td>TX</td>
<td>10h</td>
<td></td>
</tr>
<tr>
<td>TX</td>
<td>&lt;8FF bytes&gt;</td>
<td>Send dummy password</td>
</tr>
<tr>
<td>TX</td>
<td>Reset</td>
<td>Reset pulse</td>
</tr>
<tr>
<td>RX</td>
<td>Presence</td>
<td>Presence pulse</td>
</tr>
<tr>
<td>TX</td>
<td>96h</td>
<td>Issue “clear ROM” command</td>
</tr>
</tbody>
</table>
3.2 PIC16F88 TO KC22 Communication

The data stored in the PIC16F88’s EEPROM has to be sent wirelessly to the receiver, for this reason the PIC16F88 is connected to the KC22 micro module. The methods of connecting the KC22 to the PIC16F88 and configuring KC22 through software are discussed below.

3.2.1. Hardware Configuration

The PIC 16F88 microcontroller can be easily connected to the KC22 device using the inbuilt USART (Figure3.7 and Figure3.8). The UART in KC22 is compatible with the 16450 industry standard. Four signals are provided with the UART interface: the TXD and RXD pins are used for data, while the CTS and RTS pins are used for flow control. The PIC 16F88 microcontroller also has an inbuilt USART which can be configured. The PIC16F88 is connected to KC22 as shown in the Figure3.8 below.

![PIC16F88 - KC22 Communication Diagram](image)
Illustration of a KC-22 module to MCU connection.

3.2.2 Firmware Interface

KCSerial firmware provides an easy to use AT command interface using the UART. The firmware interface allows persistent storage of configuration parameters such as device name, default baud rate, and security PIN. Additionally kcSerial provides operational commands such as connections, security, read/write commands for I/O pins, and remote command mode offering same programming interface on the linked remote device as well. Please refer to kcSerial User Guide for additional information.

3.2.3 Modes of Operation

The software behavior of the kcSerial interface is similar to a Hayes compatible modem. The application has two modes, a command mode and a bypass mode. In the command mode, the host can issue specially formatted text strings called commands. These command strings can be used for configuration or to manage a connection with a remote device. Instead, it has commands that leverage off the vendor-specific command form.
Once a connection is established, the application transitions to the bypass mode. In the bypass mode, bytes sent from the host will be sent over the Bluetooth link to the remote device. Any data received from the remote device will be delivered to the host. All bytes received on the UART by the host are transferred to the remote device with the exception of the Escape sequence. While in the bypass mode, the KcSerial interface will search for this Escape sequence from the host. If this sequence is found, the application will go back to command mode. This allows commands to be issued again from the host, but the connection to the remote device will remain. Any data received on the Bluetooth link will be discarded while in command mode. While in the command mode, the kcSerial interface will send responses back to the host for commands received. Responses from the kcSerial interface will also be sent on system reset. However, it is possible to configure a disconnect notification to be sent during bypass mode. Please refer to the kcSerial Reference Guide for more details.

3.2.4 Data Transmission

Once the PIC 16f88’s UART is configured as in the fig14, it should also be configured programmatically. Pins 8 and 11 of the 16F88 are configured as RXD and TXD respectively. Now the pic16f88 is ready for the communication the AT commands can be sent directly to the Kc22.

First the version command is given to the Kc22 which replies by giving the version number of the device as shown in the Table 3.10 below. Next a bonding with the remote device is created using the Bond [BD address] [passkey] command where BD address is the Bluetooth device address, the passkey can be any four digit number. The pass key is used to create secured link with the remote device and avoid other Bluetooth devices.
accessing the Kc device. Then the Kc22 is made to go into the Bypass mode. Now the PIC 16F88 transmits the data stored in its EEPROM to the Kc22 which is transmitted to the remote device. In the bypass mode all the data that is received by the Kc22 is transmitted wirelessly to the remote host device. The table below shows the order in which the PIC16F88 transmits and receives the commands and data.

After the data has been transmitted, the Kc22 transmits the escape sequence "^#%" so it returns again to command mode. The Serial Port Profile connection is then closed using SPPDisconnect command. At last after the data transaction is done the Kc device is put to sleep mode to save power, it can be wake up from the sleep mode anytime using Exit Park command.

3.3 User Interface

The System reads of the temperature at equal intervals of time and transmits the data in ASCII format wirelessly to the remote host device. Hence for the user to understand, the data has to be converted into a human readable form. For this reason a graphical user interface as shown below Figure 3.11 was created using C# programming. The user interface reads ASCII values converts it into decimal numbers and displays it on the screen.
### Table 3.12 Transaction sequence

<table>
<thead>
<tr>
<th>Master mode</th>
<th>Command/Data</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>TX</td>
<td>AT+ZV Version</td>
<td>Command to get the version number of the KC device</td>
</tr>
<tr>
<td>RX</td>
<td>AT-ZV KC2.2</td>
<td>Version number from KC device</td>
</tr>
<tr>
<td>TX</td>
<td>AT+ZV Bond [BD address] [passkey]</td>
<td>Bond the transmitter KC device to the receiver Bluetooth device</td>
</tr>
<tr>
<td>RX</td>
<td>AT-ZV Bond enabled/ AT-ZV Bond disabled</td>
<td>Reply from the Bluetooth receiver in turn to the microcontroller</td>
</tr>
<tr>
<td>TX</td>
<td>AT+ZV SPPConnect BD address</td>
<td>Serial port connect with the remote device</td>
</tr>
<tr>
<td>RX</td>
<td>AT-ZV ConnectionUp AT-ZV -BypassMode- AT-ZV SPPConnectionClosed</td>
<td>Transition from command mode to bypass mode / Unable to enter into bypass mode.</td>
</tr>
<tr>
<td>TX</td>
<td>Data stored in the EEPROM can now be transmitted</td>
<td></td>
</tr>
<tr>
<td>TX</td>
<td>^#^$%^</td>
<td>Escape sequence</td>
</tr>
<tr>
<td>RX</td>
<td>AT+ZV Command mode</td>
<td>Enters command mode again after data transmission</td>
</tr>
<tr>
<td>TX</td>
<td>AT+ZV SPPDisconnect</td>
<td>Serial port disconnect</td>
</tr>
<tr>
<td>RX</td>
<td>AT-ZV ConnectionDown</td>
<td>Connection closed</td>
</tr>
<tr>
<td>TX</td>
<td>AT+ZV Park [BD address] [Beacon Period]</td>
<td>Enter Sleep mode</td>
</tr>
<tr>
<td>RX</td>
<td>AT-ZV Park mode</td>
<td>Sleep mode activated</td>
</tr>
</tbody>
</table>
Figure 3.7 User Interface created using C# for the Bio-Implantable System
CHAPTER 4

BIO IMPLANTABLE SYSTEM TEST RESULTS

For testing purpose, a prototype board as shown in the Figure 4.1 was built. Most of the components used were SOIC except the PIC16f88. Size was not taken into consideration when building the first prototype. The aim was make the system work i.e. to collect the information from DS2422 and transmit it wirelessly to the remote Bluetooth device and to make sure it works with any kind of Bluetooth receivers with SPP profile from different manufacturers. The PCB layout of the board is shown in figure 4.1. The following tests were done with the prototype board.

Figure 4.1 Bio-implantable System PCB layout design
4.1 Temperature Test

A thermometer was placed besides the temperature sensor DS2422 in the prototype board. A heated soldering rod was placed around 5cms above the temperature sensor and air was blown using a hand fan from top as shown in the Figure 4.2. The readings from the prototype board that was wirelessly transmitted and the readings from the thermometer were noted and a graph was plotted as shown in the Figure 4.3.

![Temperature Test Setup](image)

From the graph in Figure 4.3, it is clear known that the bio implantable system records the change in the temperature very slowly. The readings from the bio implantable system show that it takes around 3 minutes to show the correct increase in temperature.
4.2 Distance Test

A blue tooth dongle was connected to a laptop and hence the laptop served as the host station. The prototype board was turned on and after one minute the communication started. The implant system started sending temperature readings at the time interval of 1 minute. The Prototype board was moved slowly away from the host station and the temperature was noted. The transmission of data was accurate until 15m, when the board was moved beyond that, the communication failed. When the board was moved back inside the 15m range and the communication restarted again. From this experiment it was understood that the implantable system works accurately only for 15m at room temperature and normal conditions. The experiment was performed inside a laboratory at a room temperature of 21°C.
4.3 Meat Test

The final implant product has to transmit information from inside the body of an animal. For this reason the board was placed inside a meat and tested. A big piece of pork as shown the Figure 4.2 was selected. The meat had bone, flesh and blood in it. The meat was 6cms thick 20cms wide. The prototype board was placed inside a plastic zip lock and sealed properly to prevent any moisture entering. The meat was cut into two equal slices and the sealed board was placed at the center and turned on. Then the other slice was place on it covering the whole board as shown in the Figure 4.2. The communication started after a minute as expected.

Figure 4.4 Sealed Prototype board placed near the meat
Then the laptop was moved away from the meat slowly to find the maximum transmission. The same experiment was done with different meat with varying thickness and the following results as shown in Table 4.1 were obtained. A graph as shown in the Figure 4.3 was plotted with the above Table 4.1. From the graph it is clear that the RF signal will interacts with the body of the animals. The signal strength becomes weaker and weaker as the thickness of the meat increases. In research laboratories only small animals like rats, rabbits and monkeys are used and the implant is just placed below the skin the thickness of the skin is well below 20 mm. Hence it can be decided that the proposed system will be able to transmit for at least 12 feet.
Table 4.1 Meat Test results

<table>
<thead>
<tr>
<th>S no.</th>
<th>Meat type</th>
<th>Meat thickness in mm</th>
<th>Maximum Transmission distance In feet</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Beef</td>
<td>5</td>
<td>22.10</td>
</tr>
<tr>
<td>2</td>
<td>Beef</td>
<td>10</td>
<td>17.01</td>
</tr>
<tr>
<td>3</td>
<td>Beef</td>
<td>15</td>
<td>14.6</td>
</tr>
<tr>
<td>4</td>
<td>pork</td>
<td>30</td>
<td>6.01</td>
</tr>
<tr>
<td>5</td>
<td>Free air</td>
<td>0</td>
<td>50</td>
</tr>
</tbody>
</table>

Figure 4.6 Graph showing the Meat Test result
CHAPTER 5

CONCLUSIONS AND SUGGESTION FOR THE FUTURE RESEARCH

5.1 Conclusions

A novel, intelligent bio implantable system using Bluetooth technology has been designed and implemented as part of this thesis. This stand alone system records the internal body temperature of the animal within the range -45°C to +85°C at equal intervals of time and transmits it to the base station. The system requires a 5V battery with a peak current of 20mAmps to operate. The system is completely programmable and can be used for wide range of other applications by just altering the software. The system was able to communicate for a maximum 15m in free space and when placed inside a meat, it was able to transmit data up to 20 feet. Temperature resolution of 0.5°C was obtained. The system doesn’t require a specially designed receiver unit, any kind of generic Bluetooth receiver with SPP profile will receive data, and this makes the system more flexible. The digital data received by the Bluetooth receiver was converted into human readable form and displayed on the screen, this data can be further processed using processing tools for interpretations. The system consumes a maximum power of 115.5 mW, with a average power consumption of around 66mW at 35°C.
5.2 Suggestions for Further Research

There are several areas in which the bio implantable system can be improved. A few of these are mentioned below.

5.2.1 Low Power Consumption

The battery life in portable devices is greatly reduced by high power consumption. An effective method for low power design is to reduce the operating voltage. Therefore smaller capacity batteries can be chosen for the implantable system and by increasing the number of charge pump stages to get the required boost in voltage if and where necessary.

5.2.2 Induction Charging

The concept of induction charging can be used to charge the battery from outside. Most of the commercially available bio implantable devices make use of the induction charging concept for long life time.

5.2.3 Implantable Rechargeable Batteries

The lithium batteries currently used in many implantable medical devices are too big, cannot be recharged, and last only 3 years at most. It is therefore desirable to use micro miniature, rechargeable and inherently safe lithium batteries that will last for as many as 10 years. No existing lithium battery technology meets these requirements. Size is a particular challenge, because the implants are very compact in size, and typically batteries require significant fraction of the total volume. The implantable rechargeable micro batteries for BION micro stimulators will be of great interest for such type of works [18] [17]. The micro batteries weigh less than 0.2 g and have a dimension of 2.9 mm × 13 mm.
5.2.4 Fully Integrated System

The implant should be miniaturized as far as possible. The challenges in miniaturizing these systems include development of a highly efficient on-chip telemetry coil, and careful power management to eliminate the need for discrete capacitors. Further lowering circuit losses will also help in reducing volume.

5.2.5 Biomedical Issues

There are several biomedical areas that require further investigation before systems like these can find widespread use in animals and humans. Issues such as reliability of batteries, effect of transmitted power on tissues and optimal stimulation parameters need to be examined. Optimal stimulation parameters may vary from one individual to next. Hence, a feedback mechanism to vary the pulse parameters externally is desirable. Biocompatibility of the materials used in the micro system is also a serious issue that needs to be addressed. It is hoped that the system proposed in this thesis will give a start and provide insight into more exciting implantable devices to come.
BIBLIOGRAPHY


[17] Khalil Amine "Implantable, rechargeable microbatteries for BION™ microstimulators", Chemical Technology Division, Argonne National Laboratory, 2002.


APPENDIX 1

DS2422 Datasheet

GENERAL DESCRIPTION

The DS2422 temperature/datalogger combines the core functions of a fully featured datalogger in a single chip. It includes a temperature sensor, real-time clock (RTC), memory, 1-Wire interface, and serial interface for an analog-to-digital converter (ADC) as well as control circuitry for a charge pump. The ADC and the charge pump are peripherals that can be added to build application-specific dataloggers. Without external ADC, the DS2422 functions as a temperature logger only. The DS2422 measures the temperature and/or reads the ADC at a user-defined rate. A total of 6192 8-bit readings or 4096 15-bit readings taken at equidistant intervals ranging from 1s to 273hrs can be stored.

APPLICATIONS

Temperature Logging in Cold Chain, Food Safety, and Bio Science
High-Temperature Logging (Process Monitoring, Industrial Temperature Monitoring)
General-Voltage Datalogging (Pressure, Humidity, Light, Material Stress)

PIN CONFIGURATION

<table>
<thead>
<tr>
<th>TOP VIEW</th>
<th>VPAD 24</th>
<th>TEST_CG</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCLK</td>
<td>23</td>
<td>VSAT</td>
</tr>
<tr>
<td>SDATA</td>
<td>22</td>
<td>PUMP_ONZ</td>
</tr>
<tr>
<td>CNVST</td>
<td>21</td>
<td>TEST_RX</td>
</tr>
<tr>
<td>NC</td>
<td>20</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>19</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>18</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>17</td>
<td>NC</td>
</tr>
<tr>
<td>AGND</td>
<td>16</td>
<td>TEST_SPLY</td>
</tr>
<tr>
<td>X1</td>
<td>15</td>
<td>NC</td>
</tr>
<tr>
<td>ALARM</td>
<td>14</td>
<td>GND</td>
</tr>
<tr>
<td>X2</td>
<td>13</td>
<td>IO</td>
</tr>
</tbody>
</table>

1-Wire is a registered trademark of Dallas Semiconductor.

FEATURES

- Automatically Wakes Up, Measures Temperature and/or Reads an External ADC and Stores Values in 8KB of Datalog Memory in 8 or 16-Bit Format
- On-Chip Direct-to-Digital Temperature Converter with 8-Bit (0.5°C) or 11-Bit (0.0625°C) Resolution
- Sampling Rate from 1s up to 273hrs
- Programmable Recording Start Delay After Elapsed Time or Upon a Temperature Alarm Trip Point
- Programmable High and Low Trip Points for Temperature and Data Alarms
- Quick Access to Alarmed Devices Through 1-Wire Conditional Search Function
- 512 Bytes of General-Purpose Memory Plus 64 Bytes of Calibration Memory
- Two-Level Password Protection of all Memory and Configuration Registers
- Unique Factory-Lasered 64-Bit Registration Number Assures Error-Free Device Selection and Absolute Part Identity
- Built-in Multidrop Controller Ensures Compatibility with Other Dallas Semiconductor 1-Wire Net Products
- Directly Connects to a Single Port Pin of a Microprocessor and Communicates at Up to 15.4kbps at Standard Speed or up to 125kbps in Overdrive Mode
- -40°C to +85°C Operating Range
- 2.9V to 3.6V Single-Supply Battery Operation
- Low Power (1.2μA Standby, 350μA Active)

ORDERING INFORMATION

<table>
<thead>
<tr>
<th>PART</th>
<th>TEMP RANGE</th>
<th>PIN-PACKAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS2422S</td>
<td>-40°C to +85°C</td>
<td>24-lead, 300-mil SO</td>
</tr>
</tbody>
</table>

Commands, Registers, and Modes are capitalized for clarity.

Note: Some revisions of this device may incorporate deviations from published specifications known as errata. Multiple revisions of any device may be simultaneously available through various sales channels. For information about device errata, click here: www.maximic.com/errata.
ABSOLUTE MAXIMUM RATINGS*  

ALARM, PUMP_ONZ, SDATA, SCLK, CNVST, VPAD, I/O Voltage to GND  
2.0V to 5.25V  
ALARM, PUMP_ONZ, I/O Combined Sink Current  
20mA  
Operating Temperature Range  
-40°C to +85°C  
Junction Temperature  
+150°C  
Storage Temperature Range  
-55°C to +125°C  
Soldering Temperature  
See IPC/JEDEC J-STD-020A  
Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to the absolute maximum rating conditions for extended periods may affect device.

ELECTRICAL CHARACTERISTICS  

(V_{PUP} = 3.0V to 5.25V, V_{BAT} = 2.0V to 3.6V, V_{PAD} = 3.0V to 5.5V, T_A = -40°C to +85°C )  

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>SYMBOL</th>
<th>CONDITIONS</th>
<th>MIN</th>
<th>TYP</th>
<th>MAX</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standby Supply Current</td>
<td>I_{SAT}</td>
<td>V_{BAT} at 3.0V, I/O at 0V, RTC on</td>
<td>1200</td>
<td>2000</td>
<td></td>
<td>nA</td>
</tr>
<tr>
<td></td>
<td>I_{SAT}</td>
<td>0.3V, +6V</td>
<td>50</td>
<td>650</td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>Ground Current</td>
<td>I_{GND}</td>
<td>Applies individually to GND, AGND (Note 1)</td>
<td>20</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>I/O Pin General Data</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-Wire Pullup Resistance</td>
<td>R_{PU}</td>
<td>(Notes 1, 2)</td>
<td>2.2</td>
<td></td>
<td></td>
<td>kΩ</td>
</tr>
<tr>
<td>Input Capacitance</td>
<td>C_{I/O}</td>
<td>(Notes 3, 4)</td>
<td>100</td>
<td>800</td>
<td></td>
<td>pF</td>
</tr>
<tr>
<td>Input Load Current</td>
<td>I_{L}</td>
<td>I/O pin at V_{PU}, V_{BAT} = 3.6V</td>
<td>6</td>
<td>10</td>
<td></td>
<td>μA</td>
</tr>
<tr>
<td>High-to-Low Switching Threshold</td>
<td>V_{THL}</td>
<td>(Notes 4, 5, 6)</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>Input Low Voltage</td>
<td>V_{IL}</td>
<td>(Notes 1, 7)</td>
<td>0.3</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>Low-to-High Switching Threshold</td>
<td>V_{THH}</td>
<td>(Notes 4, 5, 8)</td>
<td>0.7</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>Switching Hysteresis</td>
<td>V_{HY}</td>
<td>(Notes 4, 5)</td>
<td>0.09</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>Output Low Voltage</td>
<td>V_{OL}</td>
<td>At 4mA (Note 10)</td>
<td>0.4</td>
<td></td>
<td></td>
<td>V</td>
</tr>
<tr>
<td>Recovery Time (Note 1)</td>
<td>t_{REC}</td>
<td>Standard speed, R_{PU} = 2.2kΩ</td>
<td>5</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed, R_{PU} = 2.2kΩ</td>
<td>2</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed, directly prior to reset pulse, R_{PU} = 2.2kΩ</td>
<td>5</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Rising-Edge Hold-off Time</td>
<td>t_{HOLD}</td>
<td>(Notes 4, 11)</td>
<td>0.6</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Timeslot Duration (Note 1)</td>
<td>t_{SLOT}</td>
<td>Standard speed</td>
<td>65</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed, V_{PU} &gt; 4.5V</td>
<td>8</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed (Note 12)</td>
<td>9.5</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>I/O Pin, 1-Wire Reset, Presence Detect Cycle</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset Low Time (Note 1)</td>
<td>t_{RSTL}</td>
<td>Standard speed, V_{PU} &gt; 4.5V</td>
<td>480</td>
<td>720</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard speed (Note 12)</td>
<td>660</td>
<td>720</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed, V_{PU} &gt; 4.5V</td>
<td>48</td>
<td>80</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed (Note 12)</td>
<td>70</td>
<td>80</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Presence Detect High Time</td>
<td>t_{PWH}</td>
<td>Standard speed, V_{PU} &gt; 4.5V</td>
<td>15</td>
<td>60</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard speed (Note 12)</td>
<td>15</td>
<td>63.5</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed (Note 12)</td>
<td>2</td>
<td>7</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Presence Detect Fail Time (Notes 4, 13)</td>
<td>t_{PWF}</td>
<td>Standard speed, V_{PU} &gt; 4.5V</td>
<td>1.5</td>
<td>5</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard speed</td>
<td>1.5</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed</td>
<td>0.15</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Presence Detect Low Time</td>
<td>t_{PD}</td>
<td>Standard speed, V_{PU} &gt; 4.5V</td>
<td>60</td>
<td>240</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Standard speed (Note 12)</td>
<td>60</td>
<td>287</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed, V_{PU} &gt; 4.5V (Note 12)</td>
<td>7</td>
<td>24</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed (Note 12)</td>
<td>7</td>
<td>28</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td>Presence Detect Sample Time (Note 1)</td>
<td>t_{MS}</td>
<td>Standard speed, V_{PU} &gt; 4.5V</td>
<td>65</td>
<td>75</td>
<td></td>
<td>μs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Overdrive speed</td>
<td>8</td>
<td></td>
<td></td>
<td>μs</td>
</tr>
</tbody>
</table>
APPENDIX 2

PIC16F88 Datasheet

Low-Power Features:
- Power-Managed modes:
  - Primary Run: RC oscillator, 76 µA, 1 MHz, 2V
  - RC_RUN: 7 µA, 31.25 kHz, 2V
  - SEC_RUN: 9 µA, 32 kHz, 2V
  - Sleep: 0.1 µA, 2V
- Timer1 Oscillator: 1.8 µA, 32 kHz, 2V
- Watchdog Timer: 2.2 µA, 2V
- Two-Speed Oscillator Start-up

Oscillators:
- Three Crystal modes:
  - LP, XT, HS: up to 20 MHz
- Two External RC modes
- One External Clock mode:
  - EClO: up to 20 MHz
- Internal oscillator block:
  - 6 user selectable frequencies: 31 kHz, 125 kHz, 250 kHz, 500 kHz, 1 MHz, 2 MHz, 4 MHz, 8 MHz

Peripheral Features:
- Capture, Compare, PWM (CCP) module:
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit, 7-channel Analog-to-Digital Converter
- Synchronous Serial Port (SSP) with SPI (Master/Slave) and I²C (Slave)
- Addressable Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection:
  - RS-232: operation using internal oscillator (no external crystal required)
- Dual Analog Comparator module:
  - Programmable on-chip voltage reference
  - Programmable input multiplexing from device inputs and internal voltage reference
  - Comparator outputs are externally accessible

Pin Diagram

Special Microcontroller Features:
- 100,000 erase/write cycles Enhanced Flash program memory typical
- 1,000,000 typical erase/write cycles EEPROM data memory typical
- EEPROM Data Retention: > 40 years
- In-Circuit Serial Programming™ (ICSP™)
  - via two pins
- Processor read/write access to program memory
- Low-Voltage Programming
- In-Circuit Debugging via two pins
- Extended Watchdog Timer (WDT):
  - Programmable period from 1 ms to 256s
- Wide operating voltage range: 2.0V to 5.5V

<table>
<thead>
<tr>
<th>Device</th>
<th>Program Memory</th>
<th>Data Memory</th>
<th>I/O Pins</th>
<th>16-bit A/D (ch)</th>
<th>CCP (PWM)</th>
<th>AUSART</th>
<th>Comparators</th>
<th>SSP</th>
<th>Timers 8/16-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16F87</td>
<td>7169</td>
<td>4096</td>
<td>369</td>
<td>226</td>
<td>18</td>
<td>NA</td>
<td>1</td>
<td>Y</td>
<td>2</td>
</tr>
<tr>
<td>PIC16F88</td>
<td>7169</td>
<td>4096</td>
<td>369</td>
<td>226</td>
<td>16</td>
<td>1</td>
<td>1</td>
<td>Y</td>
<td>2</td>
</tr>
</tbody>
</table>
APPENDIX 3

Kc22 Datasheet

Firmware Features

- Wireless Data Communications Subsystem
- Embedded Bluetooth Serial Port Profile (SPP)
- Easy to Use AT Command Interface Using UART
- OEM Programmable Configuration
- Remote Command And Control
- Multipoint / Piconet Capable
- 128-Bit Encryption Security
- Custom Firmware Available

Hardware Features

- Bluetooth v1.2
- 2.4 GHz Class 2 Radio
- Range Typically Exceeds 20m
- High Speed 921kbps Data Rate
- 4 Programmable I/O Pins
- External Antenna Port
- 8Mbit Flash Memory

Applications

- Data Cable Replacement
- Zero Installation Data Link
- Wireless Data Acquisition Upload/Download
- Remote Sensing
- Machine Data Uploads/Downloads
- Monitoring And Control
- Secure Mobile Financial Transactions
- Mobile Device Communications

Description

One of the smallest Bluetooth modules available, the KC-22 Bluetooth OEM Micro Module is designed for maximum performance in a minimal space. The KC-22 module includes 4 programmable input/output lines, and offers high speed serial communications up to 921Kbaud.

The KC-22 is a pre-engineered and pre-licensed PCB module that provides fully embedded, ready to use Bluetooth wireless technology. Multi-surface pads provide both bottom pads for high volume reflow soldering and edge pads for low volume hand soldering. The reprogrammable flash memory contains embedded firmware for serial cable replacement deploying the Bluetooth Serial Port Profile (SPP). Other popular Bluetooth profiles are available.

Custom firmware can be pre-loaded into these highly tuned and tested modules so that they are ready to install without additional procedures.
APPENDIX 4

PIC16F88 Microcontroller code
VITA

Graduate College
University of Nevada, Las Vegas

Karthik Nattamai Kumaresan

Local Address:
1555 E Rochelle Ave, app #134,
Las Vegas, NV 89119

Home Address:
5, Myna teppakulam 4th street
Madurai, India 625009.

Degrees:
Bachelor of Science, Engineering, 2006
Anna University, India.

Thesis Title: System Architecture for an Intelligent Implantable Bio-Telemetry Device

Thesis Examination Committee:
Chairperson, Dr. Biswajit Das, Ph.D.
Committee Member, Dr. Venkatesan Muthukumar, Ph.D.
Committee Member, Dr. Yingtao Jiang, Ph.D.
Graduate Faculty Representative, Dr. Frank Van Breukelen, Ph.D.