11 March 2005

Mr. Lakshman One
School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, BC V5A 1S6

Re: ENSC 440 Design Specifications for a Posture Measurement and Data Logging System

Dear Mr. One:

The attached document, *Design Specifications for a Posture Measurement and Data Logging System*, examines the technical specifications of our posture sensing device. Our device incorporates inclination measurement, data storage, and data transmission capabilities, and will allow researchers to monitor human movement and activity level for up to two weeks.

The design specifications provide an in-depth look at the design of the system and its subcomponents. It outlines the design decisions that have been made, and explains how the components and data structure were chosen to satisfy the requirements in our functional specifications. The document also describes the vigorous testing that will be used to verify the proper operation of our device.

The microSense team is comprised of three dynamic engineering students: Brandon Ngai, Lawrence Wong, and Josephine Wong. If you have any further comments or concerns, please contact us by e-mail at ensc440-u-sense@sfu.ca or by phone at (604) 724-8864. Thank you for your time.

Sincerely,

*Josephine Wong*

Josephine Wong
President and Chief Executive Officer
microSense metrics

Enclosure: *Design Specifications for a Posture Measurement and Data Logging System*

cc: Mike Sjoerdsma
Scott Logie
Amir Niroumand
Tony Ottaviani
Steve Whitmore

Design Specifications
for a

# Posture Measurement and Data Logging System


Team Personnel: Josephine Wong
Lawrence Wong
Brandon Ngai

Contact Email: ensc440-u-sense@sfu.ca


Submitted To: Mr. Lakshman One – ENSC 440
Mr. Mike Sjoerdsma – ENSC 305
School of Engineering Science
Simon Fraser University


Issue Date: 11 March 2005
Revision: 1.0



microSense metrics

School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, British Columbia
Canada V5A 1S6

http://usense.cjb.net

# Executive Summary

microSense metrics is developing an affordable and reliable posture measurement and data logging device that will enable researchers to monitor the movement of human subjects in their daily activities. Each device can be used as a stand-alone unit, or in conjunction with other sensor modules to track multiple parts of the body simultaneously.

The microSense system uses a commercially-available micro-electromechanical system (MEMS) accelerometer that outputs dual-axis inclination measurements with respect to gravity. The outputs of the sensor are digitized using the analog-to-digital converter (ADC) in the microcontroller. This data is then stored in a NAND flash memory chip until it can be transmitted to a computer terminal for further analysis. A direct Universal Serial Bus (USB) link is used to transfer the recorded data onto the computer.

Our system will output a time-stamped Excel spreadsheet for each sensor. Each module can be powered for up to two weeks using non-rechargeable 3-volt batteries.

The microSense posture measurement device will undergo stringent quality assurance testing to ensure that it meets the environmental, physical, operational, and power requirements listed in our functional specifications. We plan on completing our design and prototyping by April 2005.

# Table of Contents

# List of Figures

## List of Tables

# Glossary

| | |
|---|---|
| ADC | Analog-to-Digital Converter |
| CSV | Comma Separated Values (Excel file format) |
| MCU | Microcontroller |
| MEMS | Micro-electromechanical System |
| USB | Universal Serial Bus |
| USB 2.0 | Most updated Universal Serial Bus Standard |

# 1. Introduction

The microSense metrics posture measurement and data logging system aims to aid researchers in preventing injuries in construction workers and the elderly [1]. Our device uses a micro-electromechanical system (MEMS) accelerometer to monitor human subjects as they perform their daily routines. The built-in data logger allows for up to two weeks of data acquisition. At the end of each test period, a direct Universal Serial Bus (USB) link is used to transfer the recorded data onto a computer terminal, where researchers can manipulate the data and retrace the subject's movement for further analysis [2].

This document provides a technical overview of our final product design. The testing procedures that will be used to verify the proper operation of our device will be described.

The design specifications begin with a general system overview, before providing a more in-depth examination of the design of each functional block.

# 2. System Overview

The microSense posture measurement and data logging system combines the angle measurement capabilities of existing inclination sensors with the data capture capabilities of a data logging device. Our system is subdivided into three major functional blocks: data acquisition, data storage, and data transmission.

Our system block diagram is shown in Figure 1.

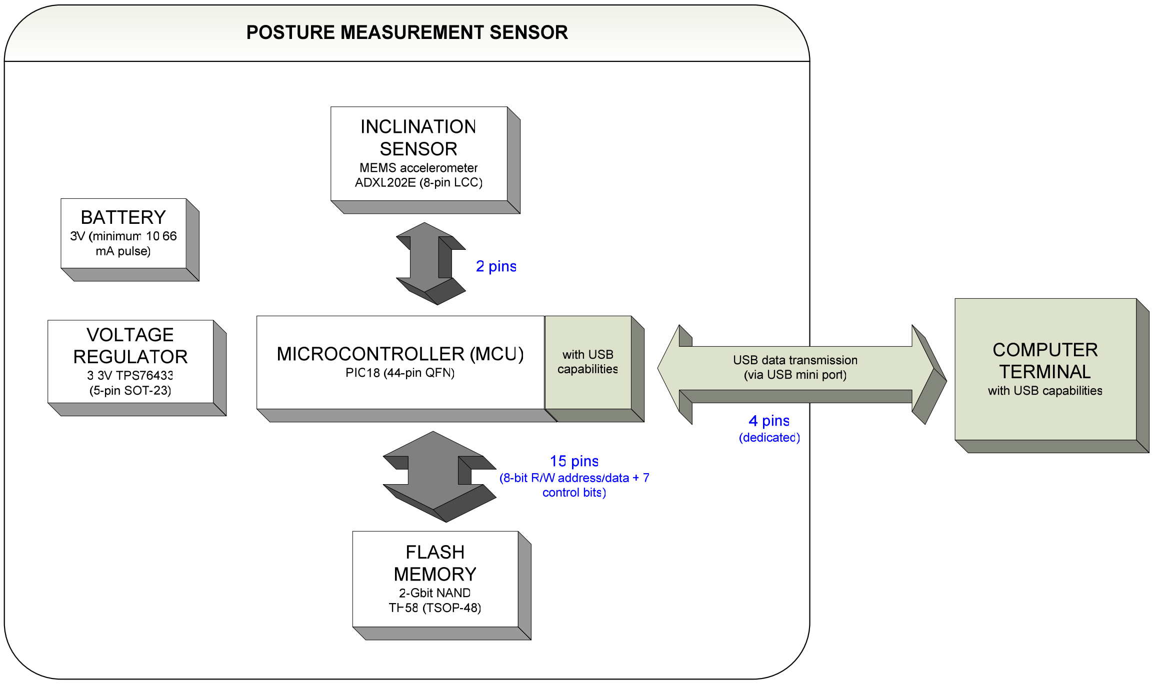



**Figure 1: System Block Diagram**

In the data acquisition stage, a dual-axis MEMS accelerometer is used to measure the angle of inclination of the device with respect to gravity. This accelerometer generates continuous analog outputs that are proportional to the angle of inclination. These outputs are connected to the analog-to-digital converter of the microcontroller and sampled at the base sampling rate of 30 samples per second.

In the data storage stage, the data measurements are committed to the NAND flash memory. The storage capacity of the memory chip is chosen based on a measurement range of $\pm 90^{\circ}$ with $0.5^{\circ}$ resolution, a sampling rate of 30 samples per second, and the maximum test duration of two weeks.

In the data transmission stage, the captured data is transferred from the module to a computer terminal via a USB connection. The raw data is then parsed and processed by an in-house software package and outputted as a Microsoft Excel-compatible comma separated value (CSV) data file.

Our device is powered by two non-rechargeable 3-volt batteries. A voltage regulator is used to maintain a steady input voltage to our other components.

# 3. System Requirements

This section describes how we will satisfy the general system requirements for our device.

## 3.1 Environmental Specifications

The sensor modules must operate between −20 ºC and 40 ºC in order to accommodate the anticipated range of climate conditions. Each component within the system must therefore meet its minimum performance expectations within that temperature range. One component that is particularly susceptible to temperature changes is the battery. The output voltage of most non-rechargeable batteries decreases with temperature. This concern has motivated the need to include a voltage regulator to maintain a steady supply voltage for the other system components.

The performance of the chosen components must also not be affected by RF signal interference to ensure reliable operation of the device.

## 3.2 Physical Specifications

The weight and size of the device should be kept to an absolute minimum. In particular, we have specified that the device should not exceed 10 cm by 5 cm by 2.5 cm in size.

Figure 2 is an illustration of a possible physical layout of the device within a 10 cm by 5 cm by 2.5 cm enclosure. The system components have been drawn to scale.

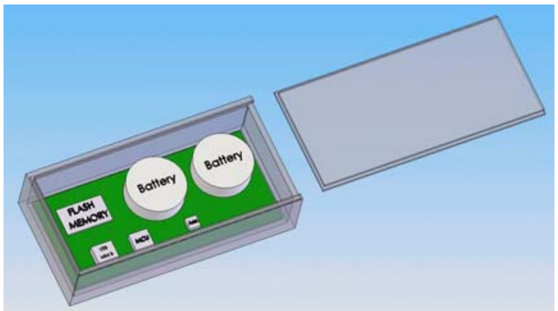



**Figure 2: Potential Physical Layout of Sensor Module**

## 3.3 Modes of Operation

The device has four modes of operation: data acquisition, data transfer, system configuration, and error-reporting.

A detailed firmware flowchart of the system is given in Appendix A.

## 3.4 Power Requirements

The need for a commercially-available battery that can continuously power the device for two weeks poses a significant design challenge that have yet to be resolved. The team initially considered using a rechargeable battery such as a cellphone battery, but abandoned the idea since non-rechargeable batteries offer a higher power density.

Table 1 shows the current usage estimates for the components in our system in operating and standby modes.

**Table 1 - Current Usage Estimates [3]–[5]**

| Component | Operating Current (μA) | Standby Current (μA) |
|---|---|---|
| MCU | 60 | 0.1 |
| Flash | 10 000 | 10 |
| Accelerometer | 600 | 600 |
| **Total** | **10.66 mA** | **0.610 mA** |

Under our current design, the MCU and the accelerometers will operate all the time. The flash memory will remain standby except when the MCU buffer is filled up and needs to store the data to the flash. Therefore, we can estimate the average power consumption in the following way:

Given that the programming time is 200μs and we are writing 30 samples per second,

$$30\ sample \times 200\ \mu s/sample \times 10000\mu A + 1\sec - (30 \times 200\mu s) \times 10\mu A = 69.94\mu A$$

$$\text{Therefore, the average operating current} = 60 + 69.94 + 600 = 729.94 \approx 0.73\text{mA}$$

Furthermore, the NAND flash memory chip that we have chosen requires an input voltage of 3.3 volts. To ensure that the supply voltage remains at a constant level, we have chosen a 3.3 V voltage regulator from Texas Instruments (TI). The TPS76433 device is a low-power low-noise 150-mA low-dropout linear regulator, and is available in a 5-pin SOT-23 package [6].

Considerable research has been put into identifying batteries that are small and lightweight, but possess a high rated capacity. Three types of 3 V batteries have been identified as possible candidates, and further lab testing will be used to select the battery that will power our device.

The technical data for the three candidate batteries are listed in Table 2.

**Table 2 - Battery Comparison Chart [7]-[9]**

| Model Number | Manufacturer | Capacity (mAh) | Voltage (V) | Type | Diameter (mm) | Height (mm) | Weight (g) |
|---|---|---|---|---|---|---|---|
| 123 | Energizer | 1500 | 3 | Cylinder | 17 | 34.5 | 16.8 |
| 1CR2 | Energizer | 800 | 3 | Cylinder | 15.6 | 27 | 11 |
| CR2477 | Renata | 950 | 3 | Coin | 24.5 | 7.7 | 8.3 |

# 4. Data Acquisition

This section describes the data acquisition components in our system.

Our device monitors the changes in the tilt angle of a body segment with respect to gravity. We can incorporate either inclinometers or accelerometers into our design, but we elected to use the ADXL202E accelerometers from Analog Devices because of their low cost and low power consumption. These ±2 g dual-axis MEMS accelerometers can track both static and dynamic acceleration. They provide analog and digital outputs, and can measure an inclination of ±90° in both the *x*- and *y*- directions as shown in Figure 3.

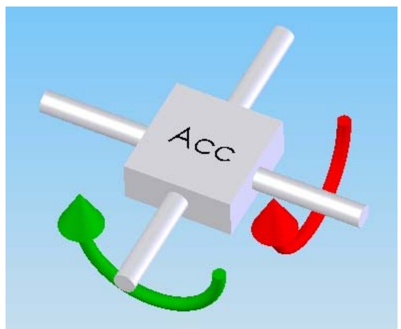



**Figure 3: A Dual-Axis Inclination Sensor**

Figure 4 provides a functional block diagram of the ADXL202E device.

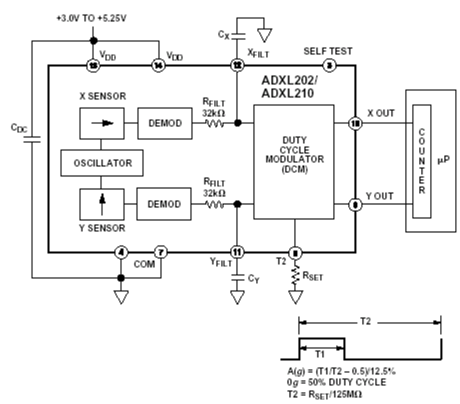



**Figure 4: Functional Block Diagram of an ADXL202E Accelerometer [5]**

Each sensor module contains one ADXL202E device. This design allows us to measure two quadrants of 3-dimensional Cartesian space. One limitation of our design is the inability of our sensor module to track horizontal movement in the *xy*-plane. This limitation is inherent to any sensor that measures inclination with respect to gravity and can unfortunately not be avoided.

Our design utilizes the analog outputs of the accelerometer. The analog outputs are proportional to the acceleration generated by gravity. These *x*- and *y*-outputs are fed into the 13-channel 10-bit ADC on the PIC18 microcontroller. The outputs are sampled at a rate of 30 samples per second, and provide a resolution of less than $\pm 0.5^{\circ}$.

# 5. Data Storage

This section describes the data storage components in our system.

## 5.1. Data Structure

Once the pitch and roll accelerations have been sampled, the raw sample data will need to be stored in memory until the user initiates a data download.

Each angle will be stored across two bytes, as shown in Figure 5.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| - | An | Sn | - | - | - | - | - | D8 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

| Bits | Symbol | Description |
|---|---|---|
| 0 – 8 | D0 – D8 | 9-bit sensor data for storing 1024 ADC output |
| 9 – 13 | - | Not implemented |
| 14 | Sn | Positive/Negative sign |
| 15 | An | Pitch/Roll angle |

**Figure 5: Data Structure of One Sample**

## 5.2. Memory Requirement

According to the proposed data structure, the storage requirements for the system are as follows. Given that each sample data occupies two bytes and there are two channels per sample, at a sampling rate of 30Hz and over a duration of fourteen days, the total memory requirement is

(2 bytes/sample) x (30 samples/s) x (86 400 s/day) x (14 days) = 72.6MB,

(72.6MB) x (2 channels) = 145.2MB.

The TH58 NAND flash chip we have chosen has 2Gbit, or 256MB, of storage and will be more than sufficient in meeting our storage requirements.

# 6. Data Transmission

This section describes the data transmission components in our system.

## 6.1. General Design

The device will communicate with a computer according to the USB 2.0 Full Speed specification. As the USB protocol gained its popularity in recent years, the traditional RS-232 serial port has become obsolete among computers. With the goal to design a user-friendly, reliable and portable system, we have chosen the USB over the RS-232 serial bus for data transmission between a computer and the device.

The USB protocol provides a well-defined packet structure that ensures fast and reliable data communication. The detail of the structure is documented in the USB 2.0 specification.

## 6.2. Transmission Rate and Error Detection

According to the USB 2.0 specifications, full speed transmission provides a transfer rate up to 12Mbit/s, which converts to a maximum speed of 1.5Mbyte/s [10]. Assuming at full speed, the two weeks sampling data can be transmitted to the computer within two minutes. In addition to its high transfer rate, the protocol implements several error detection mechanisms to ensure reliable data communication. For example, the protocol includes separate CRCs for control and data fields of each packet. Under such design, the CRC can give 100% coverage on single- and double-bit errors. Since each data sample are critical information in our device, the USB protocol ensure a reliable platform for data transmission.

# 7. Data Processing Software

A software package will be developed in-house to support data communications between the sensor modules and USB-enabled computer terminals.

## 7.1. General Design

The software package is required to retrieve the x-axis and y-axis output of the sensor and calculate the pitch and roll angles based on those values. Then the software will save the angle measurement in a CSV file format for further analysis. Moreover, the device would synchronize its internal time clock with the clock in the computer. Such design will produce a more accurate time stamp in the sample.

Since our design allows multiple sensing devices to connect to the computer to transmit data, the software would be able to identify which data comes from which devices and store the data in an organized manner.

The software flowchart of the system is provided in Appendix B.

## 7.2 Data conversion and Time Stamping

As soon as the device is plugged in to the USB port in a computer, the operating system can detect the device and automatically look for its corresponding drivers. The device driver should be pre-installed with the software package and therefore the device would be recognized and assigned a proper hardware ID for the software to communicate with it.

Once the values have been acquired, the corresponding angles will be calculated according to the following equations [5]:

$$Pitch = \arcsin\left(\frac{A_x}{1g}\right), \tag{1}$$

$$Roll = \arcsin\left(\frac{A_y}{1g}\right). \tag{2}$$

The calculated angles will be stored in a CSV file so that the data can be imported into Microsoft Excel for further analysis. Also, the software will generate the time stamp for each data sample, based on the initial time stamp created at the beginning of data acquisition. Since the time stamp is generated based on the fixed time interval between each sample, one can see that the time stamp will not serve as an accurate indication of when the sample has been taken. However, such design will allow the user to relate the changes in movement with time and provide a balanced solution between limited storage space and usability.

# 8. Quality Assurance

Our system will be tested thoroughly to ensure proper and reliable operation.

## 8.1. Data Acquisition

The data acquisition performance of our device will be verified under static and dynamic testing conditions. In static testing, we will measure the angle of inclination of a stationary object and compare our sensor measurements to measurements from a digital level. In dynamic testing, we will compare our data recordings to measurements given by a motion capture system. In both scenarios, the results should agree to an accuracy of $\pm 0.5^{o}$.

## 8.2. Data Transmission

The device will be connected to Windows 2000 and Windows XP computer terminals to ensure compatibility with both operating systems. Bidirectional data transmission via USB will be verified by sending data from the computer to the sensor and retrieving the data from the sensor.

## 8.3. Data Processing

The timestamping of data samples will be verified through a timed experiment. An external timer will be used to mark several key points during the test. These recorded timestamps will then be compared to the timestamps in the output file.

The output CSV file will be tested to ensure compatibility with Microsoft Excel.

## 8.4. Error Detection

The ability of the battery to power the device for 14 days will be verified. The device will stop acquiring data when the battery is drained and can no longer power the device.

The ability of the device to store data for 14 days at 30 samples per second will be tested. The data will be stored on the flash memory chip and uploaded to a computer afterwards for data verification. The device will also be allowed to store data continuously beyond the 14-day period. Once the 2-Gbit limit has been reached, the memory-full error will be generated and data acquisition will halt.

# 9. Conclusion

This document provides a technical overview of the microSense posture measurement and data logging system and its subcomponents. It explains the design choices that have been made, and describes how the functional and physical requirements of the system will be achieved.

The document further expands on the testing scenarios that will be used to verify the operation and reliability of our devices. We are confident that our system will satisfy the basic requirements as set out in our functional specifications for the product.

# 10. References

[1] J. Wong, L. Wong, and B. Ngai, "Project Proposal for a Posture Measurement and Data Logging System," microSense metrics, Burnaby, Canada, January 2005.

[2] J. Wong, L. Wong, and B. Ngai, "Functional Specifications for a Posture Measurement and Data Logging System," microSense metrics, Burnaby, BC, Canada, February 2005.

[3] "PIC18F2455/2550/4455/4550 Device Data Sheet," Microchip Technology Inc., 2004.

[4] "TH58NVG1S3AFT05 NAND $E^2$PROM Data Sheet," Toshiba America Electronic Components Inc., January 2003.

[5] "ADXL202E Accelerometer Data Sheet," Analog Devices Inc., 2000.

[6] "TPS76425, TPS76427, TPS76428, TPS76430, TPS76433 Linear Regulator Data Sheet," Texas Instruments Inc., 2001.

[7] "EL123 Battery Data Sheet," Eveready Battery Co. Inc.

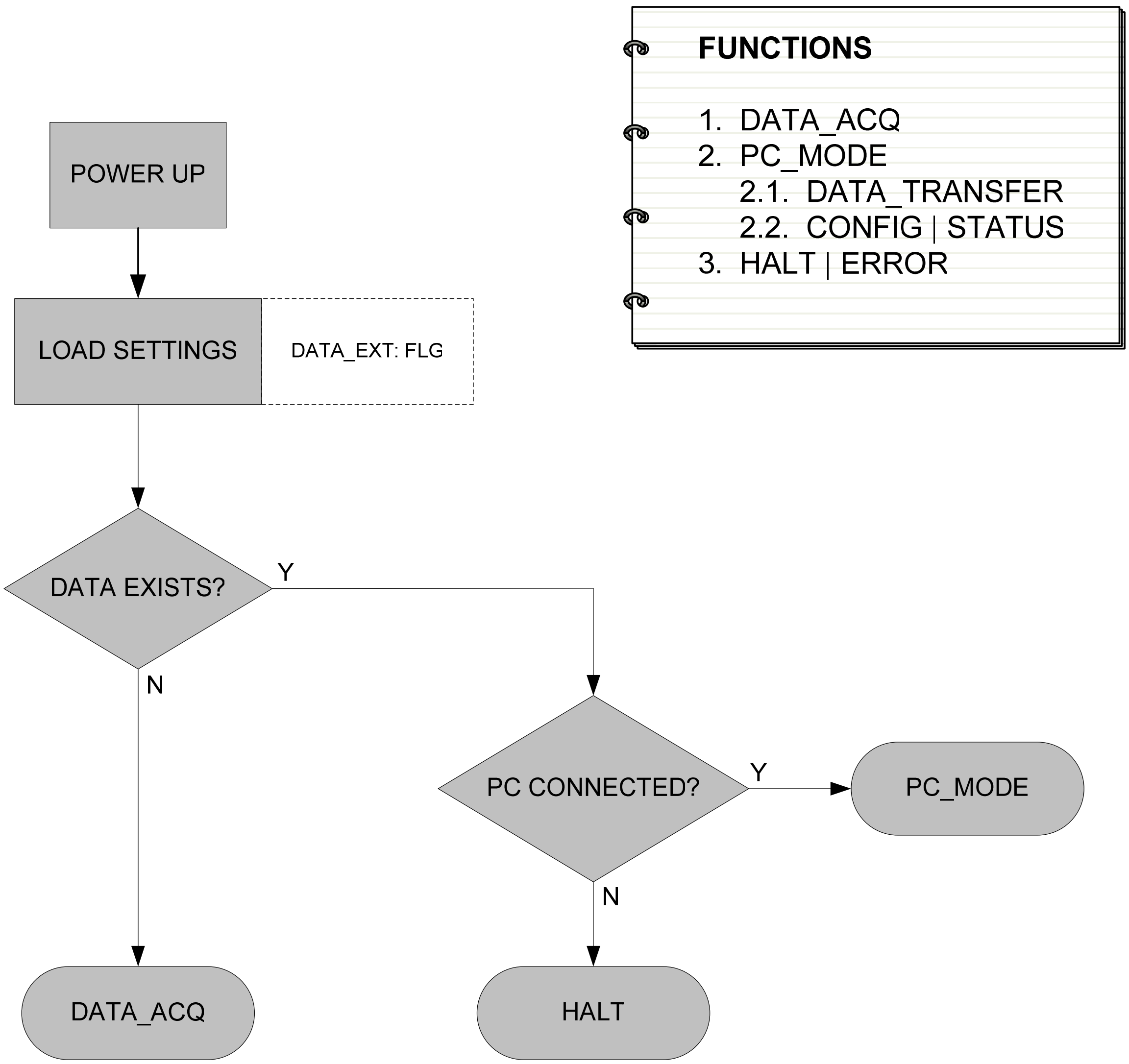
[8] "EL1CR2 Battery Data Sheet," Eveready Battery Co. Inc.

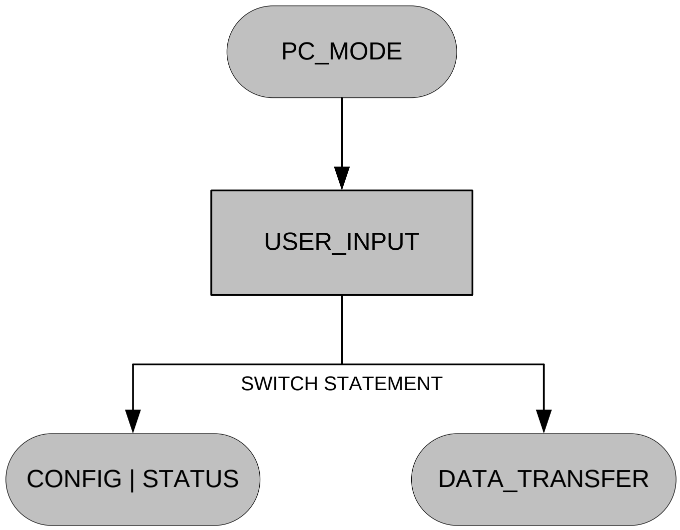
[9] "Engineering Specifications for RENATA 3V Lithium Button Cells ($MnO_2$/Li)," Renata SA, October 2002.

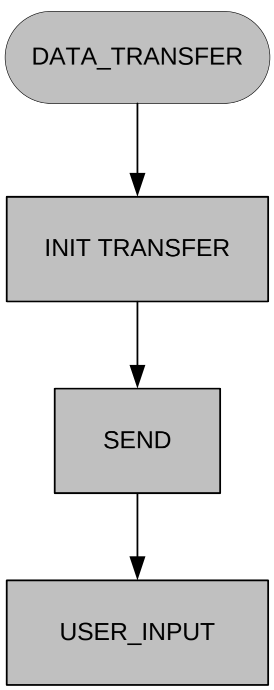
[10] "Universal Serial Bus Specification Revision 2.0," April 2000.

# Appendix A

The firmware flowchart of the system is provided in Appendix A.






**CONFIG**

1. SYNC TIME
   - write into NAND flash
2. ERASE DATA
   - set all to 0xFF

**STATUS**

1. BATTERY
   - connect battery to A/D
   - compare with hard-coded value
2. MEMORY
   - read memory counter
   - divide by total capacity

# Appendix B

The software flowchart of the system is provided in Appendix B.