University of Southern Queensland
Faculty of Health, Engineering & Sciences

Upgrade of Datalogger System

A dissertation submitted by

S. Ellis

in fulfilment of the requirements of

ENG4112 Research Project

towards the degree of

Bachelor of Computer Systems Engineering,

Submitted: October, 2013
Abstract

The aim of the project was to recommend an MCU upgrade for the Rimik DataNode data logger. Current and prospective users were surveyed to determine what features are desired in a modern data logger and the alternatives in the market were analysed also. Internally, Rimik intended to use the opportunity to fix several issues with the DataNode.

After gathering the data, it was analysed through applying the Engineering Design and Development Cycle to determine the system specifications that the upgrade would be required to address.

From those specifications, an MCU was chosen to recommend for the upgrade, a solution was developed to an ongoing issue whereby the DataNode runs SLA batteries to a totally discharged state and a prototype for a combined solar panel controller and SLA battery charger was designed and implemented.

The data generated confirmed that the existing charging system was to blame for each deficiency in that charging system, and showed that most of the errors were resolved when using the prototype charger.
Limitations of Use

The Council of the University of Southern Queensland, its Faculty of Health, Engineering & Sciences, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of material contained within or associated with this dissertation.

Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Health, Engineering & Sciences or the staff of the University of Southern Queensland.

This dissertation reports an educational exercise and has no purpose or validity beyond this exercise. The sole purpose of the course pair entitled “Research Project” is to contribute to the overall education within the student’s chosen degree program. This document, the associated hardware, software, drawings, and other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.

Dean
Faculty of Health, Engineering & Sciences
Acknowledgments

I would like to firstly thank Mr Luke Hogan of Rimik for the opportunity to undertake this project. His time, resources, enthusiasm and knowledge have made this possible.

Thanks to my supervisor, Hong Zhou, for her insight and guidance.

To my father David Ellis and my colleague Chris Johnstone, thanks for being a sounding board for me to bounce ideas off and for checking my work for errors.

To my girlfriend, family and friends, for being patient with me as I poured hours into my work and for helping me to de-stress.

This thesis was typeset using \LaTeX.

S. Ellis

University of Southern Queensland
October 2013
Contents

Abstract i
Acknowledgments iii
List of Figures xi
Nomenclature xiii

Chapter 1 Introduction 1

1.1 Aims and Objectives ........................................... 2

1.2 Overview of the Dissertation ............................... 3

Chapter 2 The Rimik DataNode 4

2.1 Chapter Overview ..................................................... 4

2.2 The Rimik DataNode system ................................. 4

2.3 Survey of Existing Users ....................................... 7

2.3.1 Internal Response ............................................... 7

2.3.2 External Response ............................................ 10

2.3.3 Limitations of the DataNode .............................. 12
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.4 GPS</td>
<td>37</td>
</tr>
<tr>
<td>3.3.5 I2C</td>
<td>38</td>
</tr>
<tr>
<td>3.3.6 Inputs</td>
<td>38</td>
</tr>
<tr>
<td>3.3.7 Interrupts</td>
<td>39</td>
</tr>
<tr>
<td>3.3.8 Record</td>
<td>40</td>
</tr>
<tr>
<td>3.3.9 RTC</td>
<td>41</td>
</tr>
<tr>
<td>3.3.10 SPI</td>
<td>41</td>
</tr>
<tr>
<td>3.3.11 USART</td>
<td>41</td>
</tr>
<tr>
<td>3.3.12 Utils</td>
<td>42</td>
</tr>
<tr>
<td>3.3.13 ZigBee</td>
<td>42</td>
</tr>
<tr>
<td>3.4 Chapter Summary</td>
<td>44</td>
</tr>
</tbody>
</table>

Chapter 4  The Method for System Development  

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Chapter Overview</td>
<td>45</td>
</tr>
<tr>
<td>4.2 The Traditional Engineering Design and Development Cycle</td>
<td>45</td>
</tr>
<tr>
<td>4.2.1 Conception</td>
<td>46</td>
</tr>
<tr>
<td>4.2.2 Specification</td>
<td>47</td>
</tr>
<tr>
<td>4.2.3 Design</td>
<td>50</td>
</tr>
<tr>
<td>4.2.4 Implementation</td>
<td>51</td>
</tr>
<tr>
<td>4.2.5 Testing</td>
<td>52</td>
</tr>
<tr>
<td>4.2.6 Maintenance</td>
<td>52</td>
</tr>
<tr>
<td>4.3 Software</td>
<td>52</td>
</tr>
</tbody>
</table>
Chapter 5  MCU Selection  

5.1 Chapter Overview  

5.2 MCU selection considerations  

5.2.1 Atmel  

5.2.2 Freescale  

5.2.3 The Data  

5.3 Analysis  

5.4 The Top Three  

5.5 Chapter Summary  

Chapter 6  Solar Controller and SLA Charger Design  

6.1 Chapter Overview  

6.2 The Existing Design  

6.3 The Solar Charger Design  

6.4 Production of the Prototype  

6.5 Testing of the Prototype  

6.6 Test Results  

6.7 Chapter Summary  

Chapter 7  The DataNode Upgrade  

7.1 Chapter Overview
B.5.5 C₃ and C₄ .............................................. 101
B.5.6 L₁ ....................................................... 102
B.5.7 D₂ ....................................................... 102
B.5.8 R₄ and R₅ ............................................. 102
B.5.9 R₁ and R₂ ............................................. 102
B.5.10 R₇ and R₈, D₅ and D₆ ............................. 102
B.5.11 C₅ ..................................................... 102
B.5.12 R₆ ..................................................... 103
B.6 Solar Charger Design: BOM ............................................. 104
B.7 Solar Charger BOM ............................................. 105
B.8 Solar Charger Design: Recorded Data ..................... 106
B.9 Solar Charger Data: Node 1 ................................. 107
B.10 Solar Charger Data: Node 1: 17 & 18 ....................... 126
B.11 Solar Charger Data: Node 1: 18 & 19 ....................... 127
B.12 Solar Charger Data: Node 1: 19 & 20 ....................... 128
B.13 Solar Charger Data: Node 1: 20 & 21 ....................... 129
B.14 Solar Charger Data: Node 2 ................................. 130
B.15 Solar Charger Data: Node 1: 17 & 18 ....................... 149
B.16 Solar Charger Data: Node 1: 18 & 19 ....................... 150
B.17 Solar Charger Data: Node 1: 19 & 20 ....................... 151
B.18 Solar Charger Data: Node 1: 20 & 21 ....................... 152
List of Figures

3.1 DataNode Hardware Block Diagram ................................. 21
3.2 DataNode Connector Pinouts ........................................... 21
3.3 DataNode Exterior .......................................................... 22
3.4 DataNode Hardware: Mainboard ....................................... 22
3.5 DataNode Hardware: Connect Board .................................. 22
3.6 DataNode Main Function Flowchart ................................. 28
6.1 LT3652 Circuit Diagram .................................................. 69
6.2 Prototype Solar Panel Controller and 12 V SLA Battery Charger PCB
    Design ................................................................. 70
6.3 Prototype Solar Panel Controller and 12 V SLA Battery Charger .......... 72
6.4 Prototype Analogue Interface Board for DataNode .................. 74
6.5 Maximums and minimums recorded across the Nodes ............... 75
Listings

3.1 An example preprocessor directive and usage . . . . . . . . . . . . . . . . . . . . 34
### Nomenclature

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC</td>
<td>Analogue to Digital Converter</td>
</tr>
<tr>
<td>ASCII</td>
<td>American Standard Code for Information Interchange</td>
</tr>
<tr>
<td>AVR</td>
<td>A family of 8 bit RISC MCUs from Atmel</td>
</tr>
<tr>
<td>Bit Banging</td>
<td>A software, as opposed to hardware, technique for serial communications</td>
</tr>
<tr>
<td>Board mod</td>
<td>AKA “green wire”, a physical modification wire often used in prototyping</td>
</tr>
<tr>
<td>BOM</td>
<td>Bill of Materials</td>
</tr>
<tr>
<td>DAC</td>
<td>Digital to Analogue Converter</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
</tr>
<tr>
<td>FET</td>
<td>Field Effect Transistor</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>GND</td>
<td>Electrical ground</td>
</tr>
<tr>
<td>$$GPGGA</td>
<td>NMEA GPS Sentence providing GPS fix data</td>
</tr>
<tr>
<td>IC</td>
<td>Integrated Circuit</td>
</tr>
<tr>
<td>IO</td>
<td>Input / Output</td>
</tr>
<tr>
<td>$I^2C$</td>
<td>Inter Integrated Circuit; communications protocol</td>
</tr>
<tr>
<td>IP67</td>
<td>Ingress Protection; totally protected against dust, protected against the effect of immersion between 15cm and 1m</td>
</tr>
<tr>
<td>ISP</td>
<td>In System Programmer</td>
</tr>
<tr>
<td>ISR</td>
<td>Interrupt Service Routine</td>
</tr>
<tr>
<td>$I_{SC}$</td>
<td>Short circuit current</td>
</tr>
<tr>
<td>LCD</td>
<td>Liquid Crystal Display</td>
</tr>
<tr>
<td>LED</td>
<td>Light Emitting Diode</td>
</tr>
<tr>
<td>LiFePO$_4$</td>
<td>Lithium ferrous phosphate</td>
</tr>
<tr>
<td>MCU</td>
<td>Microcontroller Unit</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>MEA</td>
<td>Measurement Engineering Australia</td>
</tr>
<tr>
<td>MOSFET</td>
<td>Metal Oxide Semiconductor Field-Effect Transistor</td>
</tr>
<tr>
<td>MSB</td>
<td>Most Significant Bit</td>
</tr>
<tr>
<td>MSOP12</td>
<td>Mini (or Micro) Small Outline Package with 12 pins</td>
</tr>
<tr>
<td>NCEA</td>
<td>The National Centre for Engineering in Agriculture</td>
</tr>
<tr>
<td>Node</td>
<td>The Rimik DataNode</td>
</tr>
<tr>
<td>OTG</td>
<td>This USB specification allows a device to act as a host to other devices, or a device to other hosts</td>
</tr>
<tr>
<td>PCB</td>
<td>Printed Circuit Board</td>
</tr>
<tr>
<td>PDIP-8</td>
<td>Plastic Dual Inline Package with 8 pins</td>
</tr>
<tr>
<td>PIMS</td>
<td>Pressurised Irrigation Monitoring System</td>
</tr>
<tr>
<td>PTH</td>
<td>Plated Through Hole</td>
</tr>
<tr>
<td>RTC</td>
<td>Real-Time Clock</td>
</tr>
<tr>
<td>PIN</td>
<td>Personal Identification Number</td>
</tr>
<tr>
<td>SCADA</td>
<td>Supervisory Control And Data Acquisition</td>
</tr>
<tr>
<td>SDI-12</td>
<td>Serial Data Interface at 1200 Baud</td>
</tr>
<tr>
<td>SLA</td>
<td>Sealed Lead Acid battery</td>
</tr>
<tr>
<td>SPI</td>
<td>Serial Peripheral Interface; communications protocol</td>
</tr>
<tr>
<td>SWL</td>
<td>Standing Water Level</td>
</tr>
<tr>
<td>TCP/IP</td>
<td>Transmission Control Protocol/Internet Protocol</td>
</tr>
<tr>
<td>TWI</td>
<td>Two Wire Interface; communications protocol</td>
</tr>
<tr>
<td>UART</td>
<td>Universal Asynchronous Receiver Transmitter</td>
</tr>
<tr>
<td>UAV</td>
<td>Unmanned Aerial Vehicle</td>
</tr>
<tr>
<td>UHF</td>
<td>Ultra High Frequency</td>
</tr>
<tr>
<td>USART</td>
<td>Universal Synchronous/Asynchronous Receiver Transmitter</td>
</tr>
<tr>
<td>$V_{MP}$</td>
<td>Maximum power voltage</td>
</tr>
<tr>
<td>$V_{OC}$</td>
<td>Open circuit voltage</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Good decision making is the basis for any successful endeavour. Good decisions are made possible with an accurate view of the circumstances affecting that decision, and that accuracy comes from the quality of the information available to the decision maker. High quality data is used to give an accurate depiction of a situation, which enables the best possible decisions to be made. It follows, then, that a means of gathering high quality data must be found.

This data is generated by measuring the variables which define the system, while the quality of the data can be determined by the accuracy and the timeliness of the measurements. Measuring these system variables results in a data stream, which must be captured and analysed in order to provide feedback for the decision making process. The most common method for capturing this data stream is to record measurements across fixed time intervals; this is a task to which a computer system is particularly well suited, especially as the time interval falls beyond the ability of a human operator to determine. As electronic components and computer systems have become increasingly miniaturised over time, data logging systems have reduced in size and cost to the point that simple logging devices are available in sizes no bigger than a USB memory stick.

With this reduced size and cost comes increased application and data logging systems have now found their way into a multitude of industries, with agriculture no exception. Decisions in this industry are based on forecasted weather trends, soil temperature, moisture and nutrient levels. In addition, a significant amount of data is required during the growing season to maintain or improve crop growth. Grazing and inten-
sive livestock systems also require similar data in order to maintain animal growth. While a majority of the agricultural industry still appears to operate on intuition and experience, relatively inexpensive logging and control systems are becoming more commonplace. In particular, logging systems with remote access are reducing the cost of labour and improving decision making processes. In agriculture, as in all other industries, improved quality and increased productivity are required to grow a successful business.

The Rimik DataNode is primarily aimed at the Agricultural industry with its widely dispersed data points. In its simplest form the DataNode allows logged data to be manually collected in the field, but the industry is rapidly demanding inexpensive systems that allow data to be collected remotely and transferred to centralised locations for analysis.

1.1 Aims and Objectives

The aim of this project is to design a suitable upgrade to the existing DataNode system. Certain limitations and design constraints have been identified and will be discussed. The specific objectives of the project are to:

- Perform a survey of existing users and define the attributes of the system that are appreciated, and what areas for improvement exist
- Survey the state of the market to determine what competitors exist and what they offer
- Evaluate the DataNode platform, in light of the completed surveys
- Develop selection criteria which an upgrade to the microprocessor and associated hardware, if deemed necessary, would need to address
- Based on the selection criteria, investigate available MCU families with the view of recommending a MCU to underpin the upgraded node
- Design a platform built on the recommended MCU
- If time permits, prepare a prototype for testing
This dissertation is organised as follows:

**Chapter 2** Presents an overview of the DataNode system, the results of the survey and discusses some alternatives

**Chapter 3** The existing DataNode design is described in detail

**Chapter 4** Develops the Method used for System Development

**Chapter 5** Discusses the process of selecting an MCU

**Chapter 6** Describes the Prototype Solar Panel Controller and Battery Charger

**Chapter 7** Presents an overview of how the DataNode system upgrade will address the Specifications

**Chapter 8** Conclusions and Further Work, summarises the progress of the project and discusses remaining work
Chapter 2

The Rimik DataNode

2.1 Chapter Overview

Having provided an overview of data loggers in general in the previous chapter, the Rimik DataNode data logging system will be presented. An overview of the system and some of its applications will be given. During the time that the system has been in production several limitations of the platform have been discovered and these will be discussed. In order to determine whether to undertake an upgrade of the system, a short survey was taken that Rimik and some of its customers responded to and these survey results will be discussed also. A selection of data loggers from other manufacturers have been selected for comparison with the DataNode system and this analysis is presented below. Finally, some recommendations will be made in light of all these data points in order for Rimik to move forward with its DataNode.

2.2 The Rimik DataNode system

Rimik’s versatile DataNode system is a continuation of an NCEA Pressurised Irrigation Monitoring System (PIMS). In 2010 Rimik negotiated a non exclusive license with the NCEA to manufacture and develop the PIMS with the view to producing a suite of monitoring systems for typical agricultural use. To date, the IrriWatch, WeatherWatch, BoreWatch, TankWatch and PumpWatch systems have been developed and installed at various locations around Queensland.
2.2 The Rimik DataNode system

The DataNode comprises the backbone of each system, interfacing with sensors appropriate to the task at hand. These sensors may be digital, interrupting or analogue utilising a 0-5 Volt or 4-20 mA scale. Up to 8 sensors may be attached to a single DataNode. In addition, communication with the DataNode is available via RS-232 and RS-422. External GPS, GSM/3G modems and ZigBee modules can be attached to either the RS-232 port or the RS-422 port in conjunction with a signal converter. The Node is powered by a 12 Volt supply which is usually an SLA battery backed up by a 10 Watt solar panel utilising the DataNode’s charging circuit. The node has two status LEDs; one (Power LED) displays current battery information, while the second (Logging LED) gives information on the function the Node is currently performing.

The Node can store up to 512 bytes of system variables in non-volatile Flash memory, while logged data is first cached in RAM before being written to Flash. The Node has access to 4 MB of Flash, which is organised into 512 byte pages. The ATMega644 MCU has 4 kB of RAM and 64 kB of code space, although some 2 kB of that code space is dedicated to the bootloader.

Systems with a 3G modem attached are capable of data download via a circuit switched phone call and real time updates via SMS. Additionally, a 'system status' SMS, showing the state of the Node and the current reading of attached sensors, can be sent at 9 AM on a user selectable day.

The IrriWatch system is deployed for centre pivot irrigators. In IrriWatch trim, the DataNode interfaces an interrupting tipping bucket rain gauge, a 4-20 mA pressure transmitter and a Garmin RS-232 GPS. The Node records the rainfall as it occurs, while the GPS is read and location data appended to the record only when there is water pressure in the line, as read by the pressure transmitter. In this way, several months worth of data may be stored on the Node’s 4 MB Flash.

In WeatherWatch trim, the DataNode interfaces an interrupting tipping bucket rainfall gauge, an interrupting anemometer, a 0-5 Volt wind vane as well as an SPI connected digital temperature and Relative Humidity sensor. In addition to logging the sensor readings at a user specified interval, the Node keeps track of short term historical information such as average wind speed over the last 15 minutes, average wind direction over the last 15 minutes, rainfall over the last hour, rainfall in the 24 hours to 9 AM and rainfall since 9 AM. The node also calculates delta T and dewpoint. This data may
2.2 The Rimik DataNode system

then be used for crop spraying, for example.

In BoreWatch trim, the DataNode interfaces a variety of water quality sensors. A typical installation includes a Tyco flow meter attached to the RS-422 port, along with 4-20 mA standing water level sensor, electrical conductivity meter, pH sensor, water temperature sensor and tipping bucket anemometer.

In TankWatch trim, the DataNode interfaces with 4-20 mA water depth sensors and digital level probes, in any combination of up to 8 sensors. Each digital sensor may be individually set to trigger an alarm on either a high to low or low to high transition. Each 4-20 mA sensor may be set up with one or more set points, with each set point individually capable of triggering an alert on either a high to low or low to high transition. These alerts are in the form of an SMS sent when the alert is triggered, specifying the nature of the alert, along with a user defined number of reminders on the hour until the alert condition is resolved.

In PumpWatch trim, the DataNode interfaces with 4-20 mA standing water level sensors inputs and relay outputs, in any combination of up to 8 attached devices. Each 4-20 mA sensor may be set up with one or more set points, with each set point individually capable of triggering a relay response on either a high to low or low to high transition. These responses can either turn the relay on or off, allowing for control of an external pump, for example.

The DataWatch node is installed in an IP67 rated enclosure, utilising IP67 rated connectors and switches. It can be mounted into an existing enclosure to complement other systems installed at a location, or can be mounted onto a free standing pole inside a larger enclosure which houses the Node in addition to battery, modem and other ancillaries particular to the installation. The solar panel is then installed on the top of the post in the case of a free standing pole, or in a suitable fashion in the case of a pre-existing installation. The Node’s sensors are then installed as appropriate; for example, down a bore in the case of Standing Water Level pressure transmitters, or to the post in the case of weather sensors.

Currently under development is a communications system designed around ZigBee modules to allow, for example, a TankWatch unit to negotiate with a PumpWatch unit in order to fill tanks that the former is monitoring. By using the ZigBee modules, the
units will be able to communicate even though separated by several kilometres.

2.3 Survey of Existing Users

As internal development of the DataNode has progressed, several issues have come to light that Rimik would like to address. At the same time, users have been suggesting ways in which the system could be improved to better suit their needs. In light of this, it was decided that some investigation would be undertaken in order to determine under what circumstances an upgrade to the existing design would be justified. The first step in this investigation was to identify any limitations that had been uncovered by Rimik, as well as what features could be classed as ‘desirable’; nice to have but not strictly necessary.

The collated results can be found on page 95.

2.3.1 Internal Response

First and foremost, the main factors limiting further development of the DataNode system are the lack of space in flash for additions to the Node’s firmware and lack of space in RAM for storing additional data structures. The existing MCU, the Atmel ATmega644pA, provides 64 kB of program space and 4 kB of SRAM. As development continues, more and more functionality is being built into the DataNode and this requires more and more available space.

For example, a DataNode system can be set up to log weather data, attaching wind speed, wind direction and rainfall sensors along with a temperature and relative humidity sensor. One application for such a system is with farmers who would use that data in near real time to determine whether to engage in crop dusting. However temperature and relativity alone is not enough information for an informed decision; spraying should occur when the temperature is under 28°C, when the difference between wet bulb and dry bulb temperatures (ΔT) is less than 10°C and wind speed is in the range of 7 - 10 km/h (Storrie 2004). The wet bulb temperature must therefore be calculated in order to calculate ΔT. The equation presented by Stull (2011) is used to determine the wetbulb temperature from the dry bulb temperature and relative humidity results.
2.3 Survey of Existing Users

obtained from sensors. This calculation is CPU intensive, which has program execution
time as well as power usage costs, as well as requiring code space for the various func-
tions to determine the result and display that result appropriately, along with RAM
space for storing the results.

Not every Node needs to be able to calculate $\Delta T$, however. Indeed, most Nodes would
not attach such a temperature sensor. The decision has therefore been made that the
source for the DataNode be written in such a way as to allow the programmer to include
only the sections of code which are determined to be required for a Node to provide
the desired functionality, and to be able to leave out any unnecessary code. This can
be used to free up valuable program memory and RAM space for other purposes. On
the other hand, it should be readily apparent that in the event of multiple code paths
which having been trimmed out of the firmware in this fashion, it could be possible to
require a certain selection of functionality which could push the firmware size over the
upper limit defined by the amount of program storage available minus any overhead
taken up by the bootloader. Additionally, it could be possible to assign more RAM to
data structures than is available to the Node.

With all possible code paths selected the firmware requires 73,742 bytes of program
space, or almost 113% of the available flash, and uses 4,307 bytes of data space, or
slightly more than 105% of the available RAM. Since the bootloader that the Node
employs requires approximately 3% of the flash space, the firmware would take up
roughly 116% of the usable flash. This limits the configurations that the DataNode can
be set up in; for example, it is not currently possible to have the DataNode set up as a
weather station and to also include a ZigBee module for short range communications.
It is therefore required that an upgrade to the DataNode MCU provide more flash and
RAM; a baseline of double for each has been chosen with which to select an upgrade,
meaning MCUs with 128 kB or more flash and 8 kB or more RAM will be considered.

The DataNode has access to two UARTs provided by the ATmega644PA MCU; one
is dedicated to an RS232 interface in order to attach GPS or 3G modem devices, as
well as to program the Node, while the other is able to be switched between an RS422
interface or a socket on the PCB which accepts a ZigBee module. This means that a
unit cannot attach both a GPS and 3G modem, for example. In addition, the baud
rates that the Node can achieve are dependant on the system crystal used and hence
can be either 9600 or 38,400 baud in the existing design. In practical terms, this means
that to download a full 4 MB of data from the Node takes just over 20 minutes to complete on the fastest setting. As such, an upgraded Node should be designed in such a way that a minimum of two RS232 interfaces are available to allow attaching both a GPS unit and 3G modem at the same time. RS422 or RS485 connectivity should be retained in order to be able to attach devices from other vendors, such as the Tyco flowmeters used in some BoreWatch applications, as well as to implement a Modbus interface moving forward, although this may be implemented in such a way that an RS232 and RS422 interface share an MCU UART.

After attaching the various ICs to the ATmega644, as discussed in Chapter 3, there are eight remaining IO pins available for attaching sensors to the Node. Four of these inputs can be configured by setting jumpers on the main PCB, while four are by default set up for digital IO and require surface mount resistors to be removed and reconfigured in order to attach 0 - 5 V or 4 - 20 mA sensors, for example. One of the four preconfigured channels can be used to attach an interrupting sensor, such as an anemometer or tipping bucket rainfall sensor, with the possibility of reading a second interrupting sensor through the use of a board mod. There is no provision in the existing design for logging the battery voltage without using an IO channel which may have otherwise attached a sensor. The determination within Rimik has been that 8 IO channels is the appropriate number for the DataNode, so an upgraded MCU would not need to provide additional IO for connecting more external sensors. At the same time, Rimik would require an upgraded Node to be able to log battery condition as well as ambient temperature, which would require additional IO. Most importantly every external IO channel should be capable of attaching each type of sensor without requiring modifications to the PCB componentry, bar a jumper or switch position change.

Another area which requires attention is the way the Node handles the power supply, especially the solar panel and battery. In the current design the rudimentary solar panel control scheme allows the battery plate voltage to rise above recommended limits. This causes surplus gas to form which is vented out of the battery and represents an irreversible loss of battery capacity.

Additionally, at some point during development at the NCEA the decision was made to modify the design of the power circuitry in order to provide a switched 5 V supply to the sensors instead of the original switched 12 V supply. Unfortunately this has the unintended side effect of removing the Node’s ability to shut itself off in the event that
the input voltage falls below the safe limit determined by the analogue comparator. Since the Node is unable to shut down, it will continue to draw on the battery until the input voltage falls below the under voltage lockout point on the MAX5033 switchmode power supply IC, which is at 5.2 V. At this point sulfation on the battery plates has occurred which drastically reduces the ability of the battery to provide current - in practical terms, the battery is rendered useless and must be replaced. This will be discussed further in Chapter 6. An upgrade to the Node would need to address these issues.

With these limitations defined, the next step was to determine what other features would be beneficial to provide, whether that be making it easier for users to interact with the Node, improving the functionality of the system or even reducing the cost to produce. The ability to record data directly to USB storage devices or SD cards was seen as highly desirable, along with the ability for the solar panel to charge the battery even when the Node is powered down, for whatever reason. Other suggestions included adding Ethernet and Bluetooth communications, incorporating a character or graphic display and introducing analogue output - the ability to drive a 4 - 20 mA or 0 - 5 V device, in addition to the current ability to read such sensors.

2.3.2 External Response

The limitations that the surveyed users saw all tied to collecting the data that the Node had logged. Firstly, using the 3G modem to download data requires a circuit switched data call to be made. This presents a number of issues, not least of which is the fact that at some indeterminate point in the near future Telstra will be removing the circuit switched data capability from its network. Secondly, the Node is limited to 9600 baud transfer rates during a circuit switched call. This means that a download of a full 4 MB of data would take longer than 80 minutes to complete if each data page transmitted arrived without errors and did not need to be retransmitted. Not only is this transmission time slow but for the duration of the call the telephone line would remain unavailable for other use. Work is currently in progress to transition the Node to using a packet data scheme, allowing much faster data rates via a TCP/IP connection with the option of direct uploading to an FTP server. The program memory and RAM constraints preclude implementing the TCP/IP stack on the MCU, but the
2.3 Survey of Existing Users

most recent 3G modem used by the system incorporates an embedded TCP/IP stack. An MCU upgrade would need to address the speed of off-site downloads.

While many of the systems utilising the DataNode are installed on poles embedded into the ground, the IrriWatch system is installed directly onto an irrigator well above ground height. To aid in collecting data from such Nodes a form of wireless communication could be used to provide access to the Node. If implemented, this could allow simpler management of any Nodes in remote or inconvenient locations - the user could move close enough to achieve a communications link without requiring physical access unless necessary. This would also open up the ability to utilise UAVs to periodically download data from DataNode systems in the field. It has been decided that while wireless communications will not be a defining feature in an MCU upgrade, all possible steps will be taken to ensure that such functionality may be added at a later date.

Finally the method of downloading the data from the Node has been identified as a limitation. The Node has been designed to communicate via RS232 over a special cable utilising a 6 pin IP67 rated connector at the Node end, with the pin out defined by the NCEA during the development process. Users would appreciate having the data accumulate on an SD card or USB flash stick to simplify the process. It was determined that USB connectivity would be a useful criterion for selecting an upgraded MCU.

Considering the suggestions for things that the users would find desirable, the theme of increased user accessibility continued. Several suggestions for improved user interaction were made, including the addition of an LCD module on the Node and a SCADA style interface on the user’s PC for displaying current data. Similarly, being able to interface with a smartphone app, either via direct connection or a wireless interface such as Bluetooth, was put forth. Extensions to the existing SMS notifications scheme were also suggested, such as alerting the user to prime crop spraying conditions based on parameters for $\Delta T$, wind speed and direction or for lucerne crop management with dew point data. Implementing the notifications via email, the SCADA style interface and the existing SMS connections would be appreciated. After discussions at Rimik, it was decided that this functionality would be another usage case driving the inclusion of a short range wireless interface such as Bluetooth. In conjunction with a smartphone app, wirelessly communicating data to a phone’s display could remove the desire for a dedicated display on the Node, reducing cost and complexity there at the expense of the design and maintenance of the smartphone app. Development of a SCADA style
2.3 Survey of Existing Users

interface has begun in response to internal development in remotely controlling a pump based on water levels in a tank, and it is expected that this style of interface could be extended to work with other DataNode setups.

Finally, further suggestions for extending the DataNode logger into control applications were made. This included the previously mentioned case of the operation of a pump based on levels in a tank, with Nodes attached at each end negotiating when the pump should run and for how long. Similarly, irrigation channel and stream flow control has been requested, where interfacing with proportional control devices would be required. Control loops for refrigeration and heating installations, as well as for use as rate controllers in applying fertiliser, are further features that the users consider desirable.

At this stage, it was decided that the transition from a logging unit to a control unit would be better investigated as a separate device; while some of those features could definitely be useful in conjunction with the DataNode, it was determined that the Node should not deviate too far from the core functions of a data logging unit.

2.3.3 Limitations of the DataNode

In summary of the responses discussed above, here are the limitations itemised:

- limited code space to include additional functions
- limited data space for storing data structures
- limited to 8 IO channels
- 4 of the IO channels are configurable in the field via jumpers, but 4 are not and require hardware mods to change
- a maximum of 2 interrupt IO channels
- no access to hardware I\(^2\)C interface; employing a software interface consumes two IO channels
- can not monitor current battery level without tying up one of the IO ports
- Node can not power itself down, for example when battery level is low
- system can not charge while powered down
2.4 Alternative Systems

- charging system is rudimentary and destroys batteries
- Node can not power itself up, for example when the battery level returns to an acceptable level
- Node is limited to 9600 or 38400 baud rates over UARTs
- UARTs are fixed at one RS232 and one RS422; can’t use GPS and modem on one unit, for example

2.3.4 Desirables for the DataNode upgrade

Additionally, certain desirable features have been suggested. These will be addressed as time permits.

- USB or SD connectivity for data download
- better control of power rails:
  - easy switched sensor voltage selection, 5 V or 12 V
  - easy selection of switched peripheral voltage supply, 5 V or 12 V
- ambient temperature monitoring
- analogue output channels
- dual comms options; wireless mesh connectivity in addition to current 3G modem connectivity
- in field user interface. This could be done with a character or a graphical LCD module, or via an application for smartphones

2.4 Alternative Systems

Having undertaken to define what Rimik expects of its DataNode system and how it could be improved, as well as surveying what current and prospective users would like out of the system, the next step was to investigate the industry and determine what is available in comparable data logging units.
2.4 Alternative Systems

There are several different systems currently available on the market which aim to service similar fields as the DataNode system. Several systems have been considered and are discussed below. Some of these systems are general purpose data loggers similar to the DataNode, while others cater to a particular subset of sensors such as the MEA Weather Stations.

2.4.1 MEA

Measurement Engineering Australia offer a range of data logging systems, of which their Junior Weather Station is most similar to the WeatherWatch system. It includes sensors to read air temperature, relative humidity, wind speed and direction, rainfall as well as solar radiation. The sensors employed are interfaced using either interrupting inputs or “DC voltage IO” (MEA 2013).

The sensors, solar panel and control box are attached to a short pole which is clamped to a short free standing pole concreted into the ground. The unit includes a modem which is used to send recorded data to an internet based FTP site (Mr J Hoogland 2013, pers.comm., 23 May).

A data connection to the logger is achieved over a USB interface which provides a 9600 Baud communication link. No other interfaces are exposed to the user. MEA supplies its Magpie interface software free with the logger.

2.4.2 Observant

Of the loggers that Observant produces, the C3 data logger is most similar to the DataNode. The C3 is designed to attach to a standard DIN mounting rail as a modular unit or to be installed inside a matching enclosure with an integrated solar panel as a standalone unit. The C3 incorporates a 75 W h LiFePO₄ battery and GPS module as standard.

The C3 offers four digital inputs which can double as 0 - 10 V inputs and four digital outputs which can double as 4 - 20 mA inputs. It also provides two 2 wire RS485 communications ports as well as a dual purpose RS232 and SDI-12 communication port. A USB port is provided also. Other wireless communication interfaces offered as
2.4 Alternative Systems

options are a 470 MHz UHF radio, 915 MHz data radio and a 3G GSM radio.

The digital inputs may be connected to devices which give a pulsed output, although the C3 is only capable of reading pulses down to 0.5 ms duration, or 1 kHz with a 50% duty cycle. While traditional sensors may be attached to the four digital input and four digital output channels, Observant maintains a list of “Supported Devices” on the C3’s product page, the vast majority of which feature RS485 and Modbus connectivity (Observant 2013). Since these are addressed multi-drop buses, more than one device can be daisy chained on a single port, assuming unique addresses are used.

Observant offers the Observant Global web interface which allows users to view and manage their installation. They also provide a free iPhone app which allows the user to view data their installation has recorded.

2.4.3 Campbell Scientific

Of the systems Campbell Scientific produce, the CR800 Measurement and Control System is closest to the DataWatch system. It utilises the Renesas H8S/2322 MCU, a 16 bit MCU with 2 MB of Flash available for the operating system. There are 4 MB of battery backed SRAM, which is shared by the MCU, program storage and data storage. There are 6 analogue voltage inputs, 2 pulse counters, 2 switched voltage excitation outputs, 4 digital IO ports which can be paired as 0-5 V UARTs; an isolated UART for communicating with mains powered equipment, an RS-232 port for communicating with battery powered equipment and a switched unregulated 12 V output (Campbell Scientific 2013).

An RS485 interface is conspicuous in its absence. Campbell Scientific do offer a wide range of expansion modules for the CR800 data logger, increasing the total cost of ownership dramatically. Starter software is included with the datalogger, although further cost is associated with software of higher functionality. Campbell Scientific require the user to define how and when the attached sensors will be read by writing their own code.
Maxon produce a wide range of wireless modems and have recently developed IO functionality into their top of the range SmartMax modem. Maxon have released their EnviroMax telemetry solution which combines the SmartMax modem, its IO daughterboard and two 12 V 7 Ahr SLA batteries inside an IP66 rated enclosure.

The SmartMax offers USB, RS232 and RS485 interfaces alongside its 3G modem capabilities. The IO daughterboard offers 6 digital inputs and 4 digital outputs, one pulse input capable of reading from 0 - 1 kHz and two analogue inputs with 12 bits of resolution and sampling rates up to 10 samples/s.

Maxon can also supply an optional 20 W solar panel along with a mounting bracket and arm. The SmartMax employs a web interface for interacting with the modem, which allows the user to set up the attached sensors and specify a data collection scheme. The EnviroMax offers no local storage of data; in order to record data it relies on its 3G modem to send the data off site to an FTP server.

**2.4.5 dataTaker**

The dataTaker DT80M was the final data logger investigated and also the most feature rich. Its 5 analogue channels allow it to attach up to 3 2-wire inputs per channel when the inputs use a common reference, allowing up to 15 analogue sensors to be read. It has 8 digital IO channels, where all 8 channels are logic level inputs, while 4 channels are logic level outputs and the other 4 use a FET with open drain in order to supply up to 100 mA at up to 30 V. The 8 digital inputs can also count pulses up to 10 Hz. For higher speed input signals the DT80M has 4 counters capable of reading up to 100 kHz. Additionally, the DT80M includes a latching relay output which can supply up to 1 A at 30 V$_{DC}$.

dataTaker has provided the DT80M with a USB port, which can be used in ASCII mode or TCP/IP mode, a 10 Mbps Ethernet interface and a “Serial Sensor Port”, which can be used for RS232, RS422 or RS485 communications and supports the Modbus protocol. In addition to this, the 8 digital IO can be used to provide up to 4 SDI-12 channels, which support up to 10 SDI-12 sensors each for a total of up to 40 SDI-12 sensors.
2.5 Comparison of the systems

With 128 MB of data storage capacity, dataTaker rate the DT80M as being capable of storing roughly 10 million points of data. There is also the option to attach a USB flash device, increasing capacity by approximately 90,000 points of data/MB (dataTaker 2013). The integrated modem allows remote wireless access to the logger, as well as providing FTP upload and the ability to email data or alarms directly from the DT80M.

dataTaker’s dEX logger software come installed on the DT80M and is accessed through a web browser over the Ethernet or USB ports or via the modem. dEX provides a GUI environment allowing the user to configure the data logger as well as display logged data via charts, tables or virtual mimic panels in near real time. A terminal interface is part of the GUI, allowing access to the native command language of the logger.

2.5 Comparison of the systems

The data loggers shown above offer a view of a cross section of the market in terms of functionality, features and cost.

Considering firstly the IO of the systems, it can be seen that the DataNode is positioned reasonably well against the other loggers. It offers greater configurability over the MEA, Observant and Maxon systems and approaches the level of the Campbell Scientific system. The dataTaker, aimed at the high end of the market, outstrips the Node’s capacity handily. In terms of sheer volume of IO, also, the Node holds up well. It offers as many IO channels as the MEA and Observant, half as many as the Campbell Scientific and Maxon and a fraction of the dataTaker system’s.

This suggests that the configuration of IO in the DataNode is balanced well against the alternative systems. Thus if the upgrade addressed the IO limitations determined earlier, namely the ability to easily configure all IO channels, dedicating an I²C interface instead of tying up two IO channels and being able to monitor battery state without using the IO channels then the number of additional IO channels need not necessarily be increased. Now, for the Node to dedicate pins to an I²C interface, as well as monitor battery state, for example, would require more pins than the ATmega644PA has available.

Looking at external connectivity next, several interesting factors stand out. Firstly,
2.5 Comparison of the systems

the MEA system is alone in offering a single wired external connection alongside its modem; all other systems offer at least two wired connections in addition to their various wireless options. Secondly, that the DataNode is surprisingly joined by the Campbell Scientific logger in not offering USB connectivity given the ubiquity of the interface. Finally, the fact that the DataNode is the only system not making use of TCP/IP for communicating with remote servers and user systems shows that this is functionality that Rimik must implement in order to remain competitive.

On other fronts, the DataNode is represented well. By offering the ZigBee module, it joins the Observant system in providing shorter range wireless communications, allowing systems in the field to communicate between each other. While other interfaces, notably RS485, allow communication between systems at reasonable distance, the fact that using wireless communication so greatly simplifies inter system cabling is more than enough reason to utilise such technologies in the open paddock environments where DataNode systems are generally installed.

The DataNode offers an RS422 port which is physically compatible with RS485. Where the other systems which employ RS485 have a distinct advantage over the DataNode is in implementing the Modbus protocol, which allows multiple devices to share the same interface with the MCU. This protocol is used widely in industry and represents another area of functionality that Rimik would be wise to incorporate into the DataNode.

In terms of data storage, the dataTaker DT80M completely eclipses all of the other systems considered. The rest of the systems offer much more similar capacities. In the case of the DataNode, its 4 MB of logged data storage is enough to record over three months worth of data at 5 minute logging intervals. When it is considered that each system mentioned integrates, or can optionally include, a 3G modem it becomes hard to justify large storage sizes. Instead of simply accumulating months of data in the one place until the user puts aside enough time to initiate a physical connection with the system and download whatever has been recorded, incremental uploads of that same data can occur at much more frequent intervals. The benefits are obvious: system reliability is improved through data redundancy, the ability to make good decisions is improved through the availability of relevant and timely data, user responsiveness to system parameters is increased and system cost and complexity can be reduced as the requirement to store large volumes of data is eased.
An overview of the DataNode design has been given, describing its features and function at a relatively high level and presenting some of the ways in which the system has been applied to the task of collecting real world data. As with any system, interacting with the DataNode in non-trivial situations has highlighted several shortcomings in the Node. The opportunity was taken to discuss internally these limitations and to also set forth ideas for features that would be desirable in an upgraded Node. At the same time existing and prospective users were surveyed to gather more data, as well as considering a snapshot of the other data logging systems on the market in order to help verify what, if any, changes should be made to the DataNode system.

In light of this information, some recommendations can be made. Firstly, that in order to maintain the number of IO channels available to the user while improving the functionality of the system, the DataNode should increase the IO pins available on the MCU. Secondly, that developing the functionality of the Node to make better use of the technology available in 3G modems should be seen as a priority in order to improve the ability of the Node to report data to the user. This would require access to larger volumes of program storage and SRAM than the ATmega644PA can provide.

In a similar way, the inclusion of the Modbus protocol in the DataNode system would open up opportunities to interface more devices and to get into step with the other manufacturers. Again, this would require more program and RAM space.

Additionally the selection of external interfaces should be improved such that the existing desired usage cases can be accommodated. This would result in two RS232 ports being available to the user. An additional port providing an RS485 interface would be recommended, although being able to switch between RS232 and Zigbee for one of two ports and between RS232 and RS485 for the other would alleviate most issues. Further, it would be advisable to add USB capability for communicating with the Node, uploading firmware and even attaching storage devices. This would require both additional code and RAM storage space, but also IO pins on the MCU.

It can be seen that ATmega644 MCU can not satisfy the requirements for the updated DataNode: it lacks the necessary Flash space, RAM space and IO pins. Hence a replacement part will be investigated and a recommendation given.
Chapter 3

The DataNode in detail

3.1 Chapter Overview

This Chapter will explore the hardware and software design of the DataNode. A detailed overview of the DataNode system will be given. The hardware design will be discussed and the various components of the overall design will be broken down. This will be followed by a discussion of the software design, including flowcharts for the super loop architecture and function descriptions from the source files.

3.2 The DataNode Hardware

The DataNode system comprises a main PCB, connector PCB and a switch, housed in an IP67 rated enclosure. All of the connectors used are IP67 rated, along with the switch.

3.2.1 The ATmega644PA

The DataNode system is underpinned by the Atmel ATmega644PA Microcontroller. This 8 bit microcontroller hosts 64 kB of program memory and 4 kB of SRAM (Atmel 2012). It also features 2 kB of internal EEPROM, although this is unused in the DataNode. It provides two programmable USARTs, an SPI serial interface and an 8 channel,
3.2 The DataNode Hardware

Figure 3.1: DataNode Hardware Block Diagram

Figure 3.2: DataNode Connector Pinouts
3.2 The DataNode Hardware

Figure 3.3: DataNode Exterior

Figure 3.4: DataNode Hardware: Mainboard

Figure 3.5: DataNode Hardware: Connect Board
3.2 The DataNode Hardware

10 bit ADC. It also incorporates an on-chip analogue comparator and a programmable watch dog timer, which has a separate on-chip oscillator. It is initially programmed via an In System Programming port connecting to the SPI bus, although a bootloader is used to download the application once the part has been initialised.

The DataNode has been designed to log data to an Atmel AT45DB321D Flash chip, which can store 4 MB of data in 8192 pages of 512 B (Atmel 2011). The MCU addresses the Flash IC across the SPI bus, and it will enter a lower power mode when the chip select line input is deasserted.

Also attached to the SPI bus is a DS1390U real time clock and calendar. It counts year, month, date, day, hours, minutes, seconds and hundredths of seconds, while keeping track of days in the month and leap years. It has battery backup provided by a CR2032 coin cell battery (Maxim Integrated 2012).

The first of the ATmega644PA USARTs is connected to an RS232 transceiver IC, the Sipex SP3222E. Of the two available driver/receiver pairs in the SP3222E, only one pair is utilised. The MCU can cause the transceiver IC to enter a low power mode, whereby the IC will drop from a typical no load supply current of 0.3 mA down to a typical shutdown supply current of 1 µA (Sipex 2005). The RS232 port is the main means of communication with the DataNode, with application download occurring across the interface. Additionally, DataNodes equipped with a 3G modem or a GPS unit attach same via the RS232 port.

The second of the USARTs is connected to a quad single pole single throw switch, the 74HC4066 (NXP 2013). Each of the RX and TX pins connect to a pair of the switches. The switches are controlled by another two pins from the MCU, connected in such a way as to switch the USART between either of two connections; a ZigBee module or an SP3071E RS422 transceiver IC.

There are headers on the mainboard for an onboard ZigBee module. Digi XBee modules have been selected for use with the DataNode. In addition to the USART connections via the 74HC4066, the module has access to RTS, CTS and SLEEP_RQ lines on the MCU, which facilitate communications in the case of RTS and CTS and allow the MCU to put the module into a low power sleep mode in the case of SLEEP_RQ. Idle supply current for the XBee Pro module is around 30 mA, while sleep mode supply current is
The SP3071E IC is a full duplex RS422 transceiver, allowing the MCU to interface with RS422 devices. The RS422 port on the DataNode is intended for use with an external ZigBee module, allowing the module to be placed some distance away from the Node without impeding data transfer. The DataNode also accepts user communications via the RS422 port, which allows a user to interrogate the node without disturbing the modem or GPS unit attached to the RS232 port. The DataNode is set up so that it will not accept application download via the RS422 port. The SP3071E typically draws 0.8 mA and has no ability to enter a low power mode (Exar 2010).

The MCU controls a two colour LED for notifying the user. While the DataNode is in the start up routine the LED will flash with a red pulse for 10 ms each second. During normal operation, it gives a green ‘heartbeat’ flash for roughly 10 ms each second. When the unit is reading the attached sensors or performing a log operation, the MCU drives both diodes to give an orange pulse with the same 10 ms each second flash.

The MCU has the ability to provide power to different sections of the board. The nominal 12 V battery supply can be switched to the “analogue voltage” circuit, V_A, which can be found on pin 2 in each IO channel connector. Similarly, the MCU can switch the V_{422} circuit, which provides nominal 12 V battery supply to pin 1 of the RS422 connector. Finally, the V_{232} circuit and V_{SW5V} circuit can be switched, although these are switched together. V_{232} provides nominal 12 V battery supply to pin 1 of the RS232 connector, while V_{SW5V} provides a 5V rectified supply to pin 2 of the RS232 connector. These supplies are also available at various test points, as well as other places throughout the main and connect boards.

It is common for the original design of the power circuitry to be changed through board modifications in current DataNode implementations; for example, a modification might put switched 5 V on pin 2 of the IO connectors rather than the nominal 12 V battery supply, or a GPS or 3G modem might require 5 V rather than the nominal 12 V battery supply to be on pin 1 of the RS232 connector.
The connector board hosts a second, smaller, MCU; the Atmel ATTINY13. This 8 bit MCU hosts 1 kB of program memory and 64 B of SRAM (Atmel 2010a). It also features 64 B of internal EEPROM, although this is unused in the DataNode. With a maximum of 6 IO pins, it provides an SPI serial interface and a 4 channel, 10 bit ADC. It also incorporates an on-chip analogue comparator and a programmable watchdog timer, which has a separate on-chip oscillator. It is programmed via an In System Programming port connecting to the SPI bus.

This MCU is the controller for the DataNode power subsystem. The MCU samples the battery voltage regularly and controls a second two colour LED which flashes in a 10 ms per second ‘heartbeat’, giving feedback on the voltage level in the 12 V SLA battery. When the battery voltage is below roughly 11 V, the Power LED will flash red. Below 12 V the LED will flash orange, and below 13.4 V the LED will flash green. When the battery voltage is above 13.4 V the LED will cycle through green, orange, red, orange and then green, spending 10 ms on each, every second.

In addition, whenever the battery voltage is below 13.4 V the MCU will try to charge the battery by connecting $V_{CHG}$ to $V_{BATT}$. Usually this will be a solar panel, but in some cases where mains power is available it will be a 15 V plug pack. This is achieved by driving the base of a BC847 transistor, which then drives an FDN5618P MOSFET.

In a similar fashion, the MCU provides the nominal 12 V battery supply to the mainboard. The DataNode pushbutton switch must be depressed for up to 3 seconds when switching the unit on, which allows the ATTINY time to initialise and then to drive another BC847 and FDN5618P pair which connects $V_{BATT}$ to $V_{IN}$. This connection is maintained unless the battery voltage drops below 9.25 V, whereupon the ATTINY will open the connection and thereby cause the DataNode to turn off through power loss a short time after. This time is more than enough to log the power down state, but not long enough to send an SMS via an attached 3G modem.
3.2 The DataNode Hardware

3.2.3 The IO Channels

The DataNode interfaces with the real world via up to eight sensors attached to Port A on the MCU. The DataNode has been designed so that it will interface with 4-20 mA, 0-5 V, digital, frequency and interrupting sensors. IO1-IO4 are intended for use with digital or frequency sensors and require board modifications to use the other sensor types. IO5-IO8 can be set up for any of these types of sensors via jumpers on the mainboard. IO1 can also be set up to interface an interrupting sensor, such as a tipping bucket rain gauge. By performing a board modification, the DataNode is able to attach a second interrupting sensor, such as an anemometer, to any one of the remaining IO channels.

Additionally, any of the IO channels can be used as digital outputs by changing a jumper on the connector board and including a pull up resistor in the connector. The jumper brings a BC817 transistor into the circuit; when the output is driven HIGH, the transistor conducts, which results in a path to GND and the output will be LOW. When the output is driven LOW, the pull up resistor will cause the output to be HIGH.

These eight IO channels are brought out to the connector board in four pairs. Each pair goes to a separate connector, where pin 1 is a continuous voltage supply, pin 2 is a switched voltage supply, pins 3 and 4 are the IO channels and pins 5 and 6 are GND. See Figure 3.2 for further details.

3.2.4 The Power Subsystem

With the ATTINY providing supply to the mainboard by connecting $V_{BATT}$ to $V_{IN}$, the remainder of the power subsystem is as follows: an adjustable step down DC-DC converter, the Maxim MAX5033DASA, is set to provide a 5 V supply rail at up to 500 mA and up to 90% efficiency; $V_{5V}$. The IC has a programmable under voltage lockout mode which is unused in the existing design. This means that the low voltage drop out on the IC is 5.2 V (Maxim Integrated 2010). Following the MAX5033DASA is an LM1117 linear regulator which is used to provide a 3.3 V supply from the 5 V supply; $V_{3.3V}$ (Texas Instruments 2013).
3.3 The DataNode Firmware

3.2.5 Further Reading

A link to the datasheet for each of the ICs mentioned above may be found in the list of references. For those ICs where a specific datasheet page has been referenced, please find the datasheet attached in Appendix D.

3.3 The DataNode Firmware

The firmware for the DataNode utilises a Super Loop Architecture, with a timer interrupt waking the Node out of a low power sleep state to begin each iteration of the super loop. The source code is written in a hybrid of C and C++, which reflects the period of time over which development has taken place, the number of people involved in writing the code, the different solutions posed for each problem and the increasing complexity as time has passed.

The DataNode source code is separated into 13 source files and their corresponding header files. Each will be discussed in finer detail below.

3.3.1 Generic

The generic.c source file is the base on which the DataNode firmware stands. It contains the main() function, which initialises the Node on unit power up and then drops into the super loop. Once entered, this loop will run until the unit is powered down. It also contains the two functions for initialising the non-volatile system variables, which reside in the first page of Flash memory. One is called the very first time a Node is turned on, or when the Node is being reset to default values; it initialises all the variables to a specified initial value. The other is used to update the value of each of the system variables. One or the other of these functions will be called whenever the Node is powered up.
Figure 3.6: DataNode Main Function Flowchart
3.3 The DataNode Firmware

main()

The main() function provides the foundation for all of the work the DataNode performs, from setting the initial variables during the first power up through to managing the particular functions the Node performs each second. See main() flowchart, Figure 3.6 on page 28.

During the initialisation section, the Node will set both the USART for communicating at 9600 baud. The Node will give feedback throughout the boot sequence via RS232 predominantly, although certain messages will be sent to the RS422 port as well.

The Node will then read in flash page 0, which is where the system variables are stored. The final byte in the page is given a certain value whenever a page is written; by checking the read byte against this known value, it is possible to determine whether the node is booting for the first time or not. Based on this check, the Node will execute one of two functions: in the case of the byte matching the known value, the Node will call the function which reads the stored system variables. If the byte did not match, the Node will execute the function which sets those values to their default, or initial, values.

The real-time clock is the next to be initialised. A configuration function is called before the Node checks the RTC status; if the status shows an error, the Node will assign a default date and time to the RTC and then proceed; otherwise the Node will update the time structure with the current RTC date and time.

The Node sets up a structure in RAM which allows it to cache a flash page. This reduces the number of Flash write operations the Node will perform, which helps to extend the life of the Flash storage. The next step in initialising the Node is to synchronise the contents of this cached page with the contents from when the Node last shut down. This storage and retrieval process will be described in a section 3.3.3.

A flag is then set to signal that the Node is running. An initial read of the attached sensors is performed to give the Node a baseline from which to work. The number of alerts which are available to the Node is updated, then finally a set of loop counters are initialised and the super loop is begun.

As can be seen in the flowchart, Figure 3.6, the only way for the Node to exit the super
loop is to power down. Power down occurs when the power source is disconnected, the battery is run flat or the ATTINY13 on the connect board disconnects $V_{BATT}$ from $V_{IN}$.

The first action in the super loop is to determine whether the Node should execute the shutdown procedure. The test is against a flag, which is set when the ATmega644 internal analogue comparator triggers an interrupt due to supply voltage falling below a set point at 9.25 V. The shutdown procedure is described in section 3.3.8.

The next test is to ensure that the rest of the super loop is performed at maximum once per second. This allows the Node to spend as much time as possible in a low power mode. If the test fails, the Node will execute the low power functions and return to sleep.

In the event that the test is passed, then the Node first determines whether to perform a logging action. The logging interval is user programmable, and when the timer increments past that value a flag is set. It is this flag the Node tests against. The Node then checks whether a data download is currently in progress. Logging of new data is suppressed during a data download, since it causes the Node to become unresponsive and would cause issues with the download handler. If a download is flagged as in progress, the Node will check the download time against a time-out value and cancel the download if it has taken too long. Otherwise, no log will be performed; the log flag will remain set, eventually the log will occur. If no download is in progress, the log flag is reset and the log performed.

The DataNode system includes some measures of security. For example, the user is required to enter a PIN before certain commands can be performed. In the event of a user entering the wrong PIN 3 times in a row, the Node will enter a security lockdown mode whereby it will not accept interrogation from the user on any interface. The Node will continue to perform as per usual otherwise, logging and reading as appropriate, for example.

The Node next determines whether it is in lockdown by checking against a flag. If that flag is set and the user has sent a command to the Node, it will respond with a short message signifying the lockdown state. Otherwise, when the flag is not set, the Node will parse any commands it receives as per usual.
The current time is then read from the RTC, which allows the Node to perform actions based on the time of day.

The next test is against the log interval mentioned earlier; some sensors require a warm up or settling period, which can be defined by the user. The Node checks the current value of the log counter against the value which will allow the required settling time for the sensors. When that time is reached, the Node then attempts to turn the power on to the sensors; if the sensor power is already on, which could be due to the user performing a sensor read, for example, then the Node merely flags that it requires the sensors powered on. If the power is not turned on, it will turn the power on in addition to flagging that it requires the power on.

The DataNode system is capable of sending SMS alerts. The Node keeps track of how many alerts are available to it for the day with which to send alert SMSs in a simple variable. The user is able to request the Node refresh that number, which is done by setting a certain flag. The flag will also be set in the event that the user changes the number of alerts that the Node is allowed to send. If that flag has been set, the next step in the super loop is to reset the number of alerts available to the Node.

The next check is against the time read from the RTC and a flag which is set the first time the test is passed in the hour of 9 AM. In this way the section is only serviced once per day. The Node then checks whether it has been set up to send SMS. If it has been, the number of alerts the Node is able to send for the day will be reset. It may appear that this would repeat the action only recently undertaken, however in practice the earlier reset occurs extremely irregularly. Under normal operation, the number of alerts would not be changed and this particular instance of resetting the number allowed for the day would be the only time the value would be changed.

Once the number of alerts available has been reset, the next check is against the day on which the user has elected to receive a weekly status SMS. The user can select any of the weekdays and at 9AM that morning will receive an SMS to up to three mobile telephone numbers. In fact, it is possible to have the Node SMS a landline, as long as the landline has been set up to accept SMS.

So, if the current day matches the day which the user has set, the Node will read the system variable flash page into RAM. Then, for each of the three stored telephone
3.3 The DataNode Firmware

numbers the Node will check whether the number in question is set to receive SMS. If so, the Node calls the system status SMS function and passes the number with the function call. The Node then performs a delay before moving on to the next number, which allows the modem time to send the SMS.

The Node then performs any functions which happen once per day. An example of this would be for units with a tipping bucket rain gauge; once per day, the rainfall recorded for the day is written into the record for the previous day, and then reset to zero. Finally, a flag is set which signals to the Node that it has performed this path once today, which guarantees that the path will be performed a maximum of once per day.

In a similar fashion to the above, the next test is against the time read from the RTC and a flag which is set the first time the test is passed each hour. In this way the section is serviced only once per hour.

If the check is passed, the Node will attempt to send any outstanding alert SMS. First the Node tests against the flag which signals whether the Node is to send SMS alerts. If that passes, the Node checks that there are alerts available. If there are, the Node checks whether there are alert conditions currently in effect. If so, the Node will call the SMS alert handler function. If this function returns a value greater than zero, the alerts available to the Node will be decremented by one.

The Node then performs any functions which happen once per hour, or even only on the hour but not necessarily each hour. An example of the latter would be the recording of the 12 hour minimum and maximum SWL which occurs at 9 AM and 9 PM for Nodes equipped with SWL meters. Finally, a flag is set which signals to the Node that it has performed this path once this hour, which guarantees that the path will be performed a maximum of once per hour.

For the functions with a period of one minute, the test is the current minute against the flag which is set the first time the test is passed each minute. Again, this ensures that the section is serviced only once per minute.

The Node then performs any functions which occur once per minute. Such a function might store the sum of the minute’s worth of anemometer data, for example, in a Node with an attached wind speed sensor.
The Node then attempts to reduce power consumption. Firstly, it checks whether the sensors are currently being powered. If they are, it checks whether the user or the Node itself have set a flag signalling the power is required; in the event that both flags are cleared, the Node will turn the power to the sensors off. In the other cases, the Node will do nothing.

The super loop counters are then updated so that the Node will drop into the deeper loop sections only once per second. The Node then instigates a low power sleep mode, in which it remains until a timer interrupt awakens it. At that point, the Node will return to the top of the super loop to start again.

**default_rom()**

This function is called the first time a Node is started and also whenever the user sends the corresponding command to the Node. The function flushes the RAM copy of flash_page[0] before filling it with the default values for each of the variables. These variables and their locations are defined in the header file generic.h and will be discussed below.

**normal_startup_rom()**

This function is called only on system start up in Nodes which have had a page of values written into the first page of flash. With the previous function, values were assigned to variables. With this function, however, the variables are assigned values which are read from those stored in the first page of flash. Again, these variables and their locations are defined in the header file generic.h and will be discussed below.

**generic.h**

As is usual for header files, this file contains the function prototypes for the functions defined in the generic.c source file. In addition, a number of macros are defined; these cover such procedures as setting and clearing MCU pins and a variety of methods for reading and writing to the USARTs.
This header is also where the build target is specified for a firmware. There is a core set of functions and data structures which are common to all DataNode implementations, and there are also features, functions and data structures which are specific to a particular type of Node. For example, only the WeatherWatch Nodes need to assign memory to storing historic weather data to particular data structures, while only BoreWatch units address a Tyco flowmeter will require the functions through which to communicate with the flowmeter.

The header contains the majority of the preprocessor directives which are used to help the programmers in maintaining the code. One such example is shown in Listing 3.1, whereby the programmer is able to call the function led() for the particular colour they desire, rather than having to remember which hexadecimal value corresponds to that colour. In addition, if at any time the definitions for the different colours changes, the value of ORANGE need only be changed in one location, which can avoid bugs being introduced into the code through overlooking an instance of use when modifying all instances after changing the definition.

Listing 3.1: An example preprocessor directive and usage

```
#define ORANGE 0x03
...
// when logging or deleting data, notify user by flashing logging LED
  ORANGE
if ( recording_flag || delete_data_flag )
{
  LED( ORANGE );
}
```

This header also defines the way in which variables are stored into the 512 byte flash page zero. The starting byte for each variable is defined through a preprocessor directive, while the data type and number of bytes used are recorded in comment. The decimal and hexadecimal value for the starting location is also stored.

Finally, the sensors attached to the IO channels are defined by “channel type”, such as “digital” or “4-20 mA”, and “reading type”, such as “output” or “standing water level”, and these values are defined by preprocessor directive at the end of the header file.
3.3 The DataNode Firmware

3.3.2 Command

The command source file contains the majority of the functions for interrogating the Node and for the Node to display responses to the user. The most important function within the file, in terms of both lines of code and usefulness, is the function `parse_command()`.

`parse_command()`

After defining some storage for the command received, the Node determines which UART the command came in on. It then reads in the command from the appropriate buffer and sets the UART output selection value to match.

With the command read, the Node then performs a set of string comparisons to ascertain whether the command it has received is one that it has functionality to respond to. Additionally, a check is made against the Node’s security features such that the Node will fail to respond to a recognised command if the user has not established they have privileged access. As mentioned under the discussion of the super loop, the Node cannot access this function while it is in a security timeout.

Assuming that the Node matches the input command and that the user has the appropriate privileges to call that command, the Node will then perform the associated functionality. This may range from replying over the UART with the value of a particular variable, or setting such a variable, causing the Node to perform an action such as reading the attached sensors or switching an output, through to reading or writing a whole page of flash. In fact, in Nodes with attached 3G modems, certain commands will instigate an SMS response from the Node.

On the other hand, when the Node can not match the command with any it has knowledge of, it will respond with a short message which suggests the user sends the "#?" command, which will cause the Node to output the list of commands the user has access to under their current privilege level.
3.3 The DataNode Firmware

Other functions

The remaining functions in the source file can be divided into two groups. The first group comprises those functions which either directly or indirectly display results to the user. Examples of such functions would be the handlers for sending SMS or the function `display_calibrated_readings()` which outputs the readings from the sensors attached to the Node in a plain text format which is easily read by the user. The other group of functions change values within the Node, such as `set_alert_day()`.

3.3.3 Dataflash

The dataflash source contains the functions for interfacing with the Atmel dataflash ICs. In the original PIMS specification, the unit used a 2 MB flash IC. This was later changed to the current 4 MB IC. Atmel had designed the ICs in such a way that they are able to be easily swapped out for each other; the only major difference in terms of programming for them is to change the number of pages accessible. For the 2 MB IC this is 4092 while the 4 MB IC has access to 8192. Since the first page in flash is used by the DataNode to store non-volatile system variables, this means that the available storage space for logged data is one page less than the total available to each IC.

The dataflash ICs have two buffers which can store a single page worth of data each. These buffers can be used in conjunction with any read or write operations.

Dataflash functions

Since the dataflash IC is attached to the MCU on the SPI bus, it requires that certain SPI functions are available. The functions will be discussed later with the SPI source files.

The first function in the source is used to query the dataflash chip for the contents of its Status Register. The result gives useful information about the IC; the most used piece of information is the MSB of the result, bit 7, which indicates whether the IC is ready or busy.

The next function is called upon to read any given page from flash. It waits for the
3.3 The DataNode Firmware

flash status to indicate a ready state, then initiates a main memory page read for the page passed to the function. The result is the flash IC feeding the contents of that page across the SPI bus one byte at a time, whereupon the MCU writes each byte to the flash page data structure in RAM in order of arrival.

In a similar manner, the next function writes the data structure in RAM to a given page in flash. The final byte in the page is set to a particular value, in this case 0x5A, in order to signify to the Node when it is next read that the page is ‘good’. It is this mechanism that allows the Node to distinguish between a fresh, uninitialised, Node and a Node which has been set up for use. With the final byte set, the MCU then feeds the page across to the dataflash IC one byte at a time into the dataflash’s first buffer. When the MCU calls the SPI deselect function, discussed in a later section, the dataflash IC will erase the page by writing a logic one to each position in the page, then write the contents of the buffer to the page. A read of the status register will show the device to be busy until this process is complete.

There are three erase functions that the MCU can call, due to the storage hierarchy of the dataflash IC. The base unit of storage is the page, with a group of 8 pages forming a block and a group of NN blocks forming a sector. The MCU can therefore call on a page, block and sector erase function as required. Each erase function operates in a similar fashion; an opcode is sent to the dataflash IC, followed by a starting location. When the MCU deselects the dataflash IC, the erase command is begun and the status register will show the device busy until complete.

3.3.4 GPS

The functions for interpreting the output of the Garmin 18x GPS unit are contained in this source file. The GPS unit outputs NMEA sentences over RS232 at a 1 Hz rate. The Node receives the sentences via an interrupt routine on the UART which stores the sentence as a string. When the Node requires GPS data, it waits for up to a user defined period for a complete sentence to be stored; once there is a complete sentence a copy is made for manipulation. After ensuring that the received string is a GPS fix string, denoted by the sentence code $GPGGA, this copy is broken down into the relevant data fields for the Node to use.
3.3 The DataNode Firmware

3.3.5 I2C

At present the only devices utilising I\(^2\)C in the DataNode system are the temperature and relative humidity sensors used by the Davis weather sensors. As these sensors were added after the rest of the system had been defined, the ATmega644PA’s hardware I\(^2\)C pins had already been assigned to another task. With no unassigned IO pins on the ATmega644PA to dedicate to an I\(^2\)C interface, the decision was made to implement an interface using the last two IO channels, namely channels 7 and 8. As such a series of bit banging routines have been implemented in order to provide a software interface in the absence of a dedicated hardware interface.

Ideally this software interface would be replaced with a hardware interface in the upgraded Node; this would allow faster and more efficient communications through the use of either an interrupt driven or busy-wait scheme, (Ganesan 2003), while also freeing up those inputs for attaching other types of sensors.

3.3.6 Inputs

With the exception of the interrupting sensors described below, the routines for reading all other types of sensors the DataNode can attach are contained in this file. The entry point is through the function read_inputs(), which takes the current reading of each IO channel in series and stores the raw reading value for each channel in an array. These raw values are then converted, based on the settings for the channel, to give a human readable result. The sensors can be basically any combination of the following:

**Frequency** Over a period of one second the Node counts the high to low and low to high transitions on the IO channel, returning the count of the cycles which occurred during the period. The Node will count frequencies over the range of 1 Hz to 10 kHz. This type of sensor can be attached to any of the IO channels.

**Digital** Differentiates between the digital input levels of high or low. This type of sensor can be attached to any of the IO channels.

**0 - 5 Volts** The voltage applied to the input is passed through a voltage divider consisting of two 10 k\(\Omega\) resistors to rescale it for comparison against a 2.5 V reference
3.3 The DataNode Firmware

using the DataNode’s ADC. The ADC output is converted to give a result with mV units. This type of sensor can be attached to IO channels 5 - 8

4 - 20 mA The current applied to the input is passed through a 120Ω resistor to rescale it for comparison against a 2.5 V reference using the DataNode’s ADC. The ADC output is converted to give a result with μA units. This type of sensor can be attached to IO channels 5 - 8

Digital Temperature and Relative Humidity The Davis sensor is sent a command to read the current temperature and then the Node waits for the sensor to respond with the current data, which is stored into the raw readings array in the channel 7 position. Another command, this one to read the current Relative Humidity, is then sent. The Node waits for the response and stores the value into the channel 8 position in the raw readings array. This type of sensor can only be attached to IO channels 7 and 8

Additionally, the code for interfacing a Tyco flowmeter, which attaches via the RS422 port, is included here. The Node initiates a data dump from the Tyco meter, interrupting the output when the header line and the most recent data have been read. The relevant data bytes are then stored into the Tyco peak and off peak totals.

3.3.7 Interrupts

The interrupt service routines for each of the different interrupting sources are defined within this source file. These range from the ISR that the RTC triggers twice every second, where the various system counters are updated, to the routines called when data arrives on either UART. The Analog Comparator also has an ISR, which allows the Node to record when the voltage across the battery, read through a resistor divider, falls below 9.25 V. In the original design, the Node would then be able to power itself down; at some point in development this functionality has been changed and the Node is no longer able to power down. This can and does result in Nodes destroying SLA batteries through overdischarge and is one of the criteria for a system upgrade.

Additionally, this is where the input code for the remaining sensors resides. Both the tipping bucket rainfall gauge and anemometer make use of interrupts; their ISRs increment the appropriate count after confirming the input. There is a little extra code
3.3 The DataNode Firmware

within the superloop which calls helper functions to assign the recorded pulses to a certain window of time; these will be discussed in subsection 3.3.8.

3.3.8 Record

All of the functions that manipulate or format the strings of data that the Node writes to flash are contained in this file. Also present are the functions pertaining to the circular buffer, which allows the Node to overwrite the oldest data in the event that the flash memory becomes full. The main function within the file, however, is the function the Node calls whenever the logging interval is reached. The function gathers all the data needed to form the log string, formats it appropriately and logs it by calling another function. This function uses a data structure in RAM to buffer an entire page worth of data, writing to flash only when the buffer is full. This reduces the wear that the flash experiences.

Another function is called when the Node powers up which synchronises the memory; data that was in RAM when the Node was shut down will have been written to the next available flash page. Because this data may not have used the entire page worth of space, the page is read back into RAM from flash next time the Node is powered on. This reduces wasted space and helps to avoid errors in communication when downloading data from the Node.

The remaining helper functions for the anemometer, wind vane and tipping bucket reside here also. A record of each minute, split into twelve 5 second blocks, the last 15 minutes and the last hour is kept, along with the previous day’s rainfall and the rainfall since 9 AM. These functions roll the data through each of the arrays as appropriate.

The final function is the procedure that is called when the Analogue Comparator fails a comparison, which signifies that the battery voltage has fallen below the threshold. The Node determines the next available flash page, appends a shutdown message to the contents of the data page in RAM and writes the RAM page to flash. The Node will also attempt to send an SMS, warning the user of the power down, in Nodes that attach a modem. Once that is complete the Node will attempt to turn its power circuitry off; in the event that this does not succeed, the Node will resume operation while continuing to drain the battery below safe levels.
3.3 The DataNode Firmware

3.3.9 RTC

The RTC attaches to the MCU on the SPI interface and as such utilises the various SPI routines for communication. Considering the communication at a higher level shows that there need only be five functions for communicating with the RTC: one which configures it, one which sets the time, another to read the time and finally a function to read the RTC status register and another to clear it. These last two are used only on system start up, to determine whether the RTC has lost time.

3.3.10 SPI

Both the flash and RTC ICs use the SPI bus to interface with the MCU and these functions define how that interface works. Firstly, the Node must be able to select which device it intends to communicate with. Two functions help to govern this; the first defines the settings for the communications, since the two ICs require different clock speeds. The second determines which chip select pin should be driven in order to communicate with the appropriate device. The next function works to deselect either device and to return the interface to a default state until such time as it is needed again. Finally there are two functions that facilitate information exchange on the SPI interface: one exchanges a single byte and the second exchanges a word by formatting the result of two byte exchanges.

3.3.11 USART

While the interrupts source contains the code for dealing with data coming in on the UARTs, they also require further functions in order to work as intended. Firstly, there is an initialisation function for each UART including the ZigBee UART which defines the Baud rate and various settings such as number of data, stop and parity bits. Then there is a function for each UART which allows a character to be sent across the interface. The RS232 UART is equipped with a shutdown signal line, which enables the transceiver to enter a low power mode; there is a function to switch that signal on or off. Another function selects between the RS422 IC and the ZigBee module which both attach to UART1, although not concurrently. Finally there is a function which flushes the RX buffer for UART1 with NULLs and resets the index back to the initial
3.3 The DataNode Firmware

3.3.12 Utils

Utility functions reside in this source file. Firstly, init_mcu() is called upon each start up and initialises the various parts of the MCU, from clock prescalers to the timers and interrupts. Next, setup_channels() steps through each of the eight IO channels to determine how each bit of the port, data direction and digital input disable registers should be set. Once those are determined they are written to the appropriate location and the IO channels are ready to read their sensors. The ADC is initialised before the function completes.

These are followed by three functions which facilitate the reading of tank levels; a function to set the channel to a digital input for reading, then a pair of functions which set the channel to be a digital output driving the output either high or low. This is because it was discovered that leaving the channel as a digital input caused the sensors to fail to read after a short while. Driving the channel, as an output, either high or low fixes this issue.

The delay() function, used all throughout the firmware, is next to be defined, followed by three functions which switch the voltage on the IO channel connectors, RS232 connector and RS422 connector respectively. The function that drives the LED is then defined, in front of functions which convert a byte to a binary coded decimal and vice versa. The function that reads the ADC is the last one before a collection of functions, which convert raw sensor readings into useful outputs, apply a specific formula to other readings to generate further data or set the thresholds for high and low levels in a PumpWatch system that uses a 4 - 20 mA pressure transducer rather than digital sensors.

3.3.13 ZigBee

Finally there are the functions which define how the Digi XBee modules, a type of ZigBee module, are set up and function. Significant development has gone into these functions this year as they are intrinsically linked to another final year project currently
being undertaken by my colleague Christopher Johnstone. For a much more detailed investigation of these modules, please see his project.
3.4 Chapter Summary

The hardware and software design of the existing DataNode has been presented. Each of the major ICs present in the hardware design has been introduced and their function within the node has been described. Each of the major sections of the code has been introduced, with a flowchart for the super loop provided and a description of the major functions within each section described.
Chapter 4

The Method for System Development

4.1 Chapter Overview

The previous chapters have applied the research method chosen for this research project. In the same way a method for developing the system upgrade has been selected and will be put forth.

4.2 The Traditional Engineering Design and Development Cycle

The common method for an engineering project is what is known as the “Design and Development Cycle” (Leis & Phythian 2012), or similarly the “Engineering Design Process” (NASA 2008). This cycle can be described in 6 steps;

- Conception
- Specification
- Design
- Implementation
4.2 The Traditional Engineering Design and Development Cycle

- Testing
- Maintenance

4.2.1 Conception

The adage “necessity is the mother of invention” is often the basis for an engineering project. The Conception phase of the cycle is where ideas are put forward for discussion in response to a problem. A multitude of causes could result in the problem being posed, however common causes are changing technology or a request from the client.

The main objectives of the Conception phase are to identify the opportunity, or need, and to identify the required features, or objectives.

In the case of this project, the initial idea was put forward by Rimik that the feature set of the DataNode needed to be brought up to date. Apart from the impetus from the company to move in this direction, the results of the survey undertaken as part of the project highlighted certain deficiencies that it was decided should be addressed. As such, it has been determined that an upgrade to the DataNode should increase the amount of flash memory and RAM in the MCU, to increase the number of IO pins available in order to provide for more functionality without sacrificing the number of IO channels that are exposed to the user for attaching sensors and that an additional interface port be included such that the Node could be configured for the communications requirements that have been identified. Following on from the additional interface port, the upgrade should introduce the Modbus protocol and allow the Node to utilise an attached 3G modem for uploading data to FTP servers or the like. Finally, several shortcomings relating to the control of the solar panel and charging of the SLA battery will be addressed.

These criteria can be divided into two groups: the first group, encompassing the MCU flash and RAM requirements, number of IO pins, additional communications interface and charging circuit, are primarily hardware in nature and as such will require a physical prototype to be designed. The second group consists of the additional protocols, which will need to be implemented in software - this will not be possible until the hardware design has been completed.
4.2.2 Specification

If the theme for the previous phase was “necessity is the mother of invention”, then the proverb for the Specification phase could be “a stitch in time saves nine”. Specification relates to identifying and documenting the operational and contractual details pertaining to the project. The contractual side covers the legal, budgetary and scheduling aspects while the operational side covers the functions, processes and performance aspects. Performance characteristics may include specifying speeds, response times, accuracy and stability.

The other important undertaking of the Specification phase is to develop test plans that exhaustively define the ways in which the system will be tested to prove that it conforms with the design specifications, performance included.

The contractual details for the project were covered in the Preliminary Report submitted earlier in the year. This included risk assessments and timelines for the project schedule, as well as budget considerations. For the hardware side of this upgrade, the operational details are:

**MCU: Flash** The current limit of 64 kB flash will be increased to a minimum of double as much: the recommended MCU should have 128 kB flash or greater.

**MCU: RAM** The current limit of 4 kB RAM will be increased to a minimum of double as much: the recommended MCU should have 8 kB RAM or greater.

**MCU: IO pins** The ATmega644PA has no spare IO pins. The upgrade is expected to provide a USB port, an extra communications interface, an I\(^2\)C interface and be able to monitor battery voltage and ambient temperature. An additional two IO pins should be assigned to providing additional switched power rails. This means that the recommended MCU will be required to have a minimum of 10 more IO pins available to it than the existing MCU.

**MCU: Additional Communications** The current count of two UARTs will be increased to a minimum of three, plus USB connectivity.

**Power: guaranteed Node power down** The Node was originally designed to be able to shut itself down. This functionality was modified at some point by the
NCEA in order to change how the sensors were able to the powered. The upgrade should reinstate this functionality in addition to providing the sensor power configurations required.

**Power: panel charging battery while Node is off** The existing design has never been able to charge the battery if the Node has been powered down. This can result in damaged batteries due to plate sulfation in discharged batteries as well as preventing efficient use of the solar panel. The upgrade should address this by allowing the panel to efficiently charge the battery regardless of whether the Node is powered on.

**Power: prevent over discharging** An additional issue uncovered by the NCEA’s modification of the original design is that the Node will discharge a battery down to approximately 5.2 V in the event that it cannot turn itself off and the battery receives no charge. This is because the MAX5033DASA smitchmode voltage regulator will continue to operate until the input voltage falls below its under voltage dropout value. The battery manufacturers recommend that discharge should stop at around 10 V for the range of current that the Node will draw. Discharging below this level will result in plate sulfation inside the battery, rendering it useless. The Node upgrade is expected to resolve this issue separately from the ability of the Node to shut itself down.

**Power: prevent overcharging** In the existing design, the rudimentary solar panel control scheme employed by the ATtiny13 MCU results in battery terminal voltage rising towards the solar panel open circuit voltage, $V_{OC}$, which is 21 V for the panels used. As the battery terminal voltage rises above approximately 14.5 V, the fluid inside the battery begins to boil off. This gasification represents a permanent loss of battery capacity in the case of the SLA battery. The Node upgrade is expected to resolve this issue.

Similarly, for the software side:

**Modbus protocol** The existing design incorporates the hardware required to implement a Modbus interface but the Node does not have space to implement the interface in software. Following the addition of Flash and RAM capacity that the MCU upgrade would bring, it is expected that the upgrade would implement the Modbus protocol.
TCP/IP protocol The modem used in the existing design exposes an embedded TCP/IP stack to the MCU. However, lack of code space and RAM means that the Node cannot make use of the protocol to send data to offsite storage. As with the Modbus protocol, it is expected that the upgrade would implement the protocol.

As for the testing regime:

**MCU: Flash** The operational status of the flash can be tested by writing large firmware to the MCU and confirming that it performs as expected. Further time will need to be invested by Rimik in order to design a testing methodology for the flash.

**MCU: RAM** The RAM can be tested by various publicly available algorithms, such as those suggested by (Michael Barr 2000), to ensure correct function.

**MCU: IO pins** The number of pins can be easily verified, although verifying their function requires more effort. In general, the function depends not only on the physical connections but the firmware written to interface them also. As such, physical tests with a multimeter will be required to ensure that the pins connect to only where they are meant to, and tests in software will be required in order to prove that the pins function as required.

**MCU: Additional Communications** Similarly to the extra IO pins above, both hardware and software tests will be required to prove functionality. Since these are communications ports, it should be a trivial task to ensure that two way communication occurs over the ports in software, having previously determined the physical connection. Throughput tests will also be required in order to quantify what speeds are available.

**Power: guaranteed Node power down** The physical aspect of this test will be easily discernible; if the Node shuts down on command, the test will have been passed. Again, the software comes into play in writing sufficient functionality into the code for this switch to occur.

**Power: panel charging battery while Node is off** This test will be performed in three phases: with the Node not in the circuit, with the Node in the circuit and switched off, and with the Node in the circuit and switched on during the test.
4.2 The Traditional Engineering Design and Development Cycle

In this way it will be possible to ensure that the Node does not inhibit the ability of the solar panel controller to charge a battery.

**Power: prevent over discharging** The test for this will involve setting up a Node with a battery and no solar panel, and introducing a load on the battery through the system such that the battery is drained to the point of under voltage lock out. If the Node then powers down as the threshold is passed the test will have been successful; if the Node fails to power down the test will be deemed unsuccessful.

**Power: prevent overcharging** This test will require logging the battery voltage over a number of days with a positive difference between the amount of power supplied to the system by the solar panel and the amount of power drawn by the system. This surplus power will be stored in the battery; regular logs of the battery voltage will determine whether the battery voltage has exceeded the safe operating levels for the battery. For the SLA batteries used in the DataNode system this operating level is 14.5 V.

**Modbus protocol** Testing the Modbus protocol will require access to a Modbus enabled sensor. If it is possible to confirm that communication between the Node and sensor via the Modbus interface is successful then this test will pass, otherwise the test will fail.

**TCP/IP protocol** Similarly to the Modbus protocol above, success and failure with the TCP/IP protocol will depend on successful communication between the Node and the attached device. In the case of an FTP server, the ability to create files of logged data on the server will show that the test has been successful. The practicalities of testing for this success will be more difficult than most other tests since many more systems are involved in ensuring a success: the Node, the modem, the network, the FTP server, it’s interface and the account settings for accessing the server will all play a part in whether or not the tests succeeds.

4.2.3 Design

The Design phase sees the Specification results transformed into a “System Design Document”, which is a detailed statement of implementation specifying how the system requirements are to be met.
4.2 The Traditional Engineering Design and Development Cycle

During this phase basic system schematics will be developed to show how it is envisaged that systems will be implemented and function specifications will be written to show how particular algorithms should work. This is not to say that these designs will be carried all the way through to completion; one of the main benefits of a comprehensive design phase is to identify insurmountable system specification flaws and requirements that cannot be met with available technology. It must be stressed that no change to the system design document as a result of an issue being identified should be made without a request to the original specifiers being authorised.

Building on the test plan from the Specification phase, during the Design phase test cases will be developed for each item, detailing the specific features which must be validated and exactly how they will be proved.

Much of the remainder of this dissertation is devoted to the Design phase. For example, Chapter 5 details the process of recommending an MCU to upgrade the Node with, working from the Specification phase to Design phase.

4.2.4 Implementation

The Implementation phase begins the task of making the Design a reality. The basic system schematics and functional descriptions are developed into detailed schematics and software modules as per the System Design Document. A common implementation strategy is break each section of the design up into modules according to dependencies and then implement these from the bottom up. In the case of a firmware task, such as implementing the Modbus protocol, this might look like developing a set of low level driver functions for the UART with which the protocol will be used, before developing the protocol functions themselves. Or for a hardware example, producing a schematic detailing the power circuitry which is then utilised in each of the other schematic modules.

It is common practice when using the “bottom up” implementation strategy to test that each module is functional before moving on to the next, but the detailed application of the test plan and test cases developed in the earlier phases generally wait until the following phase.
4.2.5 Testing

The Testing phase, also known as the Commissioning phase, is perhaps the most critical phase in the cycle. In fact it is usually this phase that defines the cycle; as the test cases are implemented and enacted, any failure in the system to meet the design requirements will result in returning to the Implementation or even Design phase to correct the issue. For this reason it is vitally important that the test cases be rigorous in order to catch any issues during production and well before the design is shipped as a final piece of equipment.

This testing is used to ensure that the system performance meets the requirements of the System Design Document, exhibiting appropriate stability and performance across a range of load cases. The testing should produce detailed reports which specify the level to which the system complies with the specification.

4.2.6 Maintenance

The Maintenance phase rounds out the cycle and comes into force once the design has reached manufacture. There are invariably further issues that are identified after the intensive Testing phase, often due to the user asking the design to perform in ways that the Conception phase planning did not even consider. As the system is developed to implement new features, or improve existing ones, the maintainers should use careful regression testing to ensure that modifications to the various sections of the system do not break the function of other parts of the system.

This is the phase in which the Rimik DataNode currently resides, illustrating how an idea can move with multiple projects through the Design and Development Cycle. As technology improves and new methods of performing tasks are refined, the idea of recording data remains essentially unchanged.

4.3 Software

The source code for the DataNode system is written in a mix of C and C++, with Atmel assembly used where appropriate. The source code has been written and maintained

AVR Studio is used with the AVRISP mkII to write initial code to the ATmega and ATTiny MCUs via the ISP header. The Priio bootloader is used for the ATmega; this allows new firmware to be sent to the Node via the RS232 port which allows for firmware upgrades without having to open the Node to access the mainboard.

The code has been modified and extended by at least 6 people from a starting point with the NCEA up to the current development at Rimik. When Rimik began developing the code it quickly became apparent that the rapid rate of development under the NCEA, in conjunction with differing coding styles and commenting habits across the maintainers, had resulted in a body of quite functional code that was badly in need of refactoring. The task of refactoring the source was undertaken partly in order to clean the code up and partly in order to gain an understanding of how and why the code was written in the first place. In the process, numerous peculiarities have been uncovered and resolved.
The Engineering Design and Development Cycle has been defined as the method which will be used to develop the upgrade to the DataNode. Each phase of the Cycle has been described and practical examples of how the method has been applied have been given where appropriate. The current situation of the development of the DataNode has been described, including the tools and processes used in maintaining the Node firmware. The Design method will be applied to the MCU selection in Chapter 5 and solar charger design in Chapter 6.
Chapter 5

MCU Selection

5.1 Chapter Overview

As has been previously discussed in Section 2.6, the DataNode’s existing MCU is no longer able to accommodate the demands that are being placed on it. In order for the DataNode platform to progress, it requires an MCU which provides more space for firmware, more RAM for storing data structures and more IO pins available, especially to interface communications protocols. These requirements were discussed in Section 4.2.2. This chapter will describe the process by which an MCU has been recommended.

5.2 MCU selection considerations

In investigating possible replacement MCUs, the existing environment at Rimik has been taken into account. Rimik has some experience with Atmel controllers, specifically the models used in the DataNode, and owns development tools (such as programming hardware) of small but not negligible cost. In addition Rimik has an existing and fully functional development environment for Atmel MCUs. Rimik has similar development tools and environments for Freescale MCUs because of other products in their portfolio. As such, the search for a replacement MCU has been limited to products from the families of controllers that these two manufacturers produce. This reduces the task of learning new architectures and removes the task of setting up and troubleshooting a
new development environment, both time intensive undertakings.

In order to select a MCU to replace the current DataNode MCU, each of the suitable families of MCUs was surveyed and appropriate MCUs were selected for comparison. Using a simple “advantageous, negligible, disadvantageous” system to rate each criterion, the following points of comparison have been selected:

**Architecture** An 8 bit architecture would be a negligible improvement, a 16 or 32 bit architecture would be considered advantageous since most of the current data structures use 16 bit data types at minimum. These architectures often include hardware divide functionality, also.

**Maximum Frequency** With lower clock speeds comes lower power consumption, in general, while on the other hand higher frequencies can reduce the amount of time to complete tasks, allowing the MCU to spend more time in sleep modes. There is a balance to be struck here.

**Size of flash** Double the existing, or better, would be considered advantageous. Equal or lower is considered disadvantageous, since this is a primary concern.

**Size of EEPROM** The EEPROM in the ATmega644 is currently unused. An equal or lower value would be considered negligible, a higher value would be considered advantageous.

**Size of RAM** In the same way as the flash criterion, double the existing, or better, would be considered advantageous. Equal or lower is considered disadvantageous, since this is a primary concern.

**Pin count** This criterion is tied closely with the following. More pins means, in general, more IO, but also means greater complexity in routing traces on the PCB. Values between the ATmega644 pin count and 100 pins will be considered advantageous, while values above 100 pins will be considered disadvantageous.

**Maximum IO pins** This is another primary concern for the upgrade, since the ATmega644PA does not have enough IO pins to perform all of the tasks required of it. Any value greater than the existing MCU’s will be considered advantageous, while and below or equal will be considered disadvantageous.
5.2 MCU selection considerations

ADC: number of channels, resolution  The 8 ADC channels available in the existing design are considered enough for the upgraded Node, provided that other IO are made available in addition. As such, a value greater than the existing will be deemed advantageous, while any below that will be considered disadvantageous.

Number of UART  This is another limitation of the existing design, as it impacts what sort of communications are available to a Node. Any value greater than the existing will be considered advantageous.

Number of SPI  The RTC and flash storage ICs attach on an SPI bus in the existing design. A value equal or greater than the existing would be considered advantageous, while less would be disadvantageous.

Number of TWI or I²C  Any value greater than the existing would be considered advantageous, less would be disadvantageous and equal would be negligible.

Number of timers  The existing design makes use of all the timers available on the ATmega644; hence a greater number would be considered advantageous and a lesser number disadvantageous.

Presence of WDT  In practice, all of the MCUs selected for comparison incorporate a watchdog timer of one sort or another, although not all the MCUs surveyed did. The absence would have been considered disadvantageous.

Presence of analogue comparator  This is considered advantageous.

USB connectivity  Again, since USB is a feature that the initial survey determined to be highly desirable, the presence of USB functionality would be considered advantageous and absence disadvantageous. Most MCUs selected included USB of one form or another, with some only offering Device, some Device and Host and a few offering the newer USB OTG as seen in most smartphones.

DAC: number of channels, resolution  The presence of a DAC would be seen as advantageous.

Power consumption guide  Bearing in mind that the numbers presented by the different manufacturers are, at best, only useful for comparing models within families of MCU, the power consumption for the different MCUs was included in the matrix. As such, little confidence can be placed in the numbers that Atmel and
5.2 MCU selection considerations

Freescale supply and instead real world testing would need to be used to determine what difference in power consumption exists between the MCUs selected for comparison. Therefore, since that sort of exhaustive testing is outside the scope of this project, these values have been afforded little weight in determining the outcome. As an aside, Atmel prides itself on its PicoPower technology and design philosophy and most of the Atmel MCUs selected for comparison feature this technology.

**Pricing from four major distributors** Availability and cost are another major factor to consider when selecting an MCU; there is little point specifying an MCU that cannot be sourced or increases the total system cost disproportionately. To this end four component distributors were used in order to determine pricing and availability: local suppliers Element14 (formerly Farnell) and RS Components along with DigiKey and Mouser out of the US. The different suppliers offer different pricing breakdowns for various order quantities and for this exercise the 20 or 25 off price point, depending on what levels the supplier offered, was used for the comparison.

5.2.1 Atmel

Atmel produces a comprehensive range of MCUs based on various ARM architectures, the 8051 architecture and their own AVR architecture. Given the requirements for program space and pin count, the tinyAVR family was ignored. With a focus on low power and small footprints, they top out at 8 KB flash and 32 pins. The 8051 architecture is seen as a legacy design with similar limitations to the existing megaAVR MCU and as such was not considered. This left the megaAVR, AVR XMEGA, AVR UC3 and ARM families.

Of the megaAVR family, only the ATmega1284P offers any improvement over the current 644P by doubling the amount of flash to 128 KB and quadrupling the amount of SRAM to 16 KB. Otherwise it is quite similar, having the same pin count and connectivity as the 644PA.

In the AVR XMEGA family, the other 8 bit AVR architecture, a total of four MCUs were chosen for comparison; two MCUs with 128 KB flash and 8 KB RAM and two
5.2 MCU selection considerations

MCUs with 192 KB flash and 16 KB RAM. Each MCU offers twice the number of ADC channels as the existing, while also increasing the resolution to 12 bits. They all offer USB Device connectivity. One MCU in each pair has the “C3” part number suffix, denoting a lower number of UARTs, SPI and I²C channels, whereas the other MCU in each pair has the “A3U” suffix which denotes that the device has 2 DAC channels, along with double the number of UARTs and SPI as the “C3” devices. The XMEGA MCUs improve on the megaAVR MCUs in most points of comparison.

The AVR UC3 family also yielded 4 suitable MCUs for comparison. Two MCUs have 128 KB RAM while the other two have 256 KB. Of each pair, one MCU has 48 pins and 16 KB RAM while the other has 64 pins, 32 KB RAM and a DAC channel. They otherwise have specifications in common, from number of ADC channels and their resolution to rated power consumption.

The final families considered were the ARM based MCUs. After limiting the options to those having at least 128 KB of flash, at least 8 ADC channels and Atmel’s proprietary picoPower technology, 4 MCUs based on the Cortex-M4 were added to the comparison. All have 32 KB RAM, 100 pins and 15 ADC channels with 12 bit resolution among other similarities. Similarly to the AVR UC3 family, two MCUs have 128 KB flash and the other two have 256 KB. The difference between each pair in this case is due to the more expensive MCU having USB Host and Device functionality where the cheaper MCU has only USB Device.

5.2.2 Freescale

Freescale has an extensive range of MCUs across 8, 16 and 32 bit architectures. After much research, a total of nine MCUs were added to the comparison; one from the 8 bit S08 family, five from the 32 bit ColdFire V1 family and three from the Kinetis family, based on the ARM Cortex-M0+.

The lone 8 bit MCU carries 128 KB flash and 12 KB RAM. It has 80 pins, but manages only 10 more IO pins than the existing MCU. It has 16 ADC channels with 16 bit resolution are available along with USB Device connectivity. In most other areas it is comparable to the ATmega644PA.
The five ColdFire MCUs come from three sub families under the ColdFire V1 architecture. One has USB Host, Device and OTG while the rest have no USB connectivity. Four of the five have 128 KB flash while the other has 256 KB; one has 8 KB RAM, three have 16 KB and the last has 32 KB. There are between 12 and 19 ADC channels across the group, with either 12 or 16 bit resolution. The two MCUs with 16 bit ADC resolution also integrate LCD controllers. These MCUs are also generally cheaper than the ATmega644PA.

The final three MCUs span the Kinetis L series with a KL15 part, KL25 part and a KL46 part. All have 128 KB of flash, 16 KB RAM and USB OTG connectivity. They are basically identical in all other points of comparison and compare favourably to the ATmega644PA. The KL15 part has 48 pins while the others have 64 and while the KL46 part is not well stocked across the suppliers surveyed the other MCUs are on the whole cheaper than the 644PA - in some cases significantly so.

5.2.3 The Data

The specifications for each MCU, including the ATmega644PA, is attached as Appendix B.4.

5.3 Analysis

As mentioned earlier, a simple method for discerning how the different MCUs compare to the existing ATmega644PA has been defined whereby, at each point of comparison, if an MCU had an advantage over the existing it received a ‘+’. If it had a disadvantage it received a ‘-’ and if there was no clear advantage it received a ‘.’. Note that the criteria for an MCU being better, worse or similar changed for each point of comparison. For example, an MCU with more IO pins available than the ATmega644PA may have received a ‘+’ since that would allow greater flexibility in the design, but that same MCU might have received a ‘-’ if the overall pin count was significantly higher than the existing part since that could make development, routing and production much more difficult.

The larger ATmega MCU, differing from the existing MCU only in the amount of flash,
EEPROM and RAM available, managed only those three positive scores. In all other areas it is identical to the ATmega644PA and hence received neutral scores, excepting the pricing scores where it was consistently more expensive - almost 3 times the cost from one distributor.

Of the ATxmega family the ATxmega128A3U was the stand out; the three negative marks were due to it being unavailable at one distributor and more expensive than the ATmega644PA at another as well as losing one of the power consumption comparisons. For its four neutral marks, one was because it is based on the same 8 bit architecture as the Atmega, another because it matches the ATmega644PA’s EEPROM amount, another because it includes a WDT and the final one because it equalled the ATmega644PA’s power consumption figure at one test point. In all other instances it earned a positive mark, accruing 16 in total. Importantly it offers double the number of ADC inputs while also allowing higher ADC resolution.

The 32 bit AVR MCUs were considered next. All four of these managed seven negative marks; four due to costliness and availability, two due to power comparisons against the existing MCU and the final one because they offer only one SPI channel in comparison to the ATmega644PA’s three. While the ADC resolution is increased to 12 bits over the ATmega644PA, these MCUs still offer only eight channels. Otherwise they compare favourably, with the 64 pin variants particularly so because of their high IO pin count.

The final Atmel family is based on the ARM Cortex-M4 core. With 100 pins they had the highest pin count of any MCU in the comparison, along with the highest IO pin count. Price and availability again conspired to limit the overall score with only one, the ATSAM4LS2C, managing to distinguish itself from its fellows.

The MC9S08MM128CLK, the only 8 bit MCU from Freescale to make the comparison, compared poorly to both the existing MCU and its competitors. Pricing again weighed heavily against it, but a general lack of connectivity relegated it to a low position despite boasting a 16 channel, 16 bit ADC.

Five MCUs based on the 32 bit Coldfire V1 architecture made the comparison, with a model from the 51JU subfamily joining two each from the 51EM and 51AG subfamilies. None scored particularly well although availability was good compared with the Atmel MCUs. The 51EM MCUs fared worst; with an integrated LCD controller these MCUs
have a high pin count in relation to their IO pin count and draw much more current than any other MCU in the comparison. 12 ADC channels at 16bit resolution are available although they provide no USB functionality.

The MCUs from the 51AG subfamily also fail to incorporate USB functionality. They have fewer pins while still providing at least as many IO pins as the 51EM MCUs. The ADC resolution is 12bits with 12 or 19 channels depending on the model. Apart from the increased flash and RAM they are otherwise quite comparable to the existing ATmega644PA.

The MCU from the 51JU family manages the best score of the 32 bit Coldfire MCUs because it includes USB OTG functionality. It is also the cheapest of the five despite also having, at 32 KB, more RAM than any of the others. In connectivity it compared favourably to the ATmega644PA, in particular more than doubling the number of ADC channels available while increasing the resolution to 12 bits and also including a single channel DAC.

The final Freescale family in the comparison was the ARM Cortex-M0+ based MKL family. These MCUs were quite similar; all three had the same amount of flash, EEPROM and RAM. All three offered the same connectivity options, including 14 ADC channels at 16bit resolution and a single channel DAC at 12bit resolution, excepting that the MKL15 does not offer USB OTG where the other two do and also has a smaller footprint than the other two. The MKL46 draws the most current of the three and is also less available than the other two. This leaves the MKL25 MCU as the highest rated of all the Freescale MCUs in the comparison as well as being arguably the second cheapest MCU in the entire comparison.

As shown in the attached spreadsheet on page 98, there are three main contenders in the decision. In order of descending score, these are the ATxmega128A3U, the MKL25Z128VLH4 and the ATSAM4LS2C.

5.4 The Top Three

Selecting between these top scoring MCUs became a more difficult task, finally coming down to practical considerations. The ARM Cortex-M4 based ATSAM4LS2C has the
highest pin count of the three, more than two and a quarter times the pin count of the MCU it would be replacing. This was deemed impractical because of the complexity of routing so many pins in the small form factor of the intended design, which could conceivably result in extra layers in the PCB and hence further additional cost. The MCU was therefore eliminated.

The remaining options, the 8 bit AVR based ATxmega128A3U and 32 bit ARM Cortex-M0+ based MKL25Z128VLH4, are relatively similar MCUs. They are both 64 pin ICs with 50 IO pins available, half again as many as the existing MCU. They both have 128 KB of flash, although with 16 KB of RAM the MKL25Z128VLH4 has twice as much as the ATxmega128A3U and four times as much as the ATmega644PA - both important considerations since development of the existing design is hampered by a lack of available resources in flash and RAM.

In terms of IO connectivity the two MCUs consistently improve upon the existing MCU. Both offer more ADC channels at higher resolution than the ATmega644PA. The number of UARTs is an area of distinction between the two MCUs; while the Freescale has one more than the ATmega644PA, the Atmel more than doubles that number. The Atmel has more SPI but fewer timers, while the Freescale’s USB OTG functionality trumps the Atmel’s USB Device only. Both have access to a 12 bit DAC, an improvement over the ATmega644PA, although the Atmel has an extra channel than the Freescale.

The power consumption figures do not allow for direct comparison between the Freescale and either Atmel MCU, since the tests were performed under different circumstances; the Freescale numbers were generated with the MCU clocked at rated speed, while the Atmel were downclocked. Comparing the numbers for the ATxmega128A3U and ATmega644PA appears reasonable since Atmel tested them under similar conditions. The published numbers would suggest similar power economy is available from both MCUs, contingent upon firmware programming.

Finally, the Freescale MCU is more widely available than the Atmel MCU as well as being less expensive than the ATmega644PA from three of the four surveyed distributors. Additionally Freescale offers a development board based on the MKL25Z128VLK4 MCU, which is the 80 pin variant of the MKL25Z128VLH4, that Element14 sells for $10.72 at the time of writing. On the other hand, the Atmel is stocked at only one of
the two Australian distributors surveyed and is more expensive than the ATmega644PA at that distributor while being markedly cheaper at the US based distributors. Atmel offers no development board based on an MCU similar to the ATXmega128A3U at the time of writing.
Suitable MCUs from two manufacturers have been compared and contrasted. An analysis of each MCU against the ATmega644PA, the DataNode’s current MCU, has been performed and each MCU has been given a rating so as to compare between MCUs. These ratings showed that three MCUs, namely the ATxmega128A3U, MKL25Z128VLH4 and ATSAM4LS2C, offered the best choice for the DataNode MCU moving forward.

The MKL25Z128VLH4 compared favourably due to its low cost, versatile USB functionality, high resolution ADC and DAC and amount of RAM. Access to an official development board weighed heavily in its favour also. The option to upgrade within the Freescale family is another bonus; an MCU with 256 KB Flash and 32 KB RAM in the same package and pinout is available, allowing a simplified upgrade path if and when required. As such, the MKL25Z128VLH4 was chosen for further development.
Chapter 6

Solar Controller and SLA Charger Design

6.1 Chapter Overview

Having selected an MCU in chapter 5, the design for the solar charger will be presented. This represents the second major area for upgrading the DataNode system for improved reliability and resilience to adverse conditions.

6.2 The Existing Design

The power subsystem requires overhaul because of several fundamental flaws:

1. Firstly, the Node cannot turn itself off to prevent damage to the battery due to draining it below safe levels.

2. The Node needs to be powered on in order to charge the battery, which renders the solar panel useless whenever the Node is off.

3. The charging system is a rudimentary design where when the solar panel output voltage is higher than the battery terminal voltage, and the terminal voltage is below a threshold set at a nominal 14.5 V, the ATtiny MCU connects the solar panel supply to the battery to charge it - the panels used are rated at 21 V open
6.2 The Existing Design

The existing design circuit, with maximum power at 16.8 V to 17.2 V depending on the panel. The state of charge is checked again after approximately two seconds.

The typical two stage charger for SLA batteries comprises a bulk charge stage and a float charge stage. The bulk charge stage is commonly called the constant current mode, where a set level of current is supplied to the battery until the terminal voltage rises to a certain level. Once that level is reached the top-up stage, or constant voltage mode, is engaged. During this phase the terminal voltage is kept to a set level and the charging current drops away over time. Once the charging current falls below a set level the battery is declared fully charged. An optional third mode is also possible, which provides a float charge at a lower constant voltage level than the top-up. This float mode can be engaged indefinitely, maintaining the battery in a fully charged condition (PowerSonic 2011).

In practice, when the DataNode’s battery is discharged it will draw as much current as the solar panel can provide, often pulling the panel output voltage down to match the battery terminal voltage. As the battery stores more charge, however, the current drawn will decrease and so the panel output voltage will remain high. This, in turn, allows the battery terminal voltage to climb above the nominal 14.5 V cutoff that the ATtiny regulates - terminal voltages as high as 16.5 V have been observed. This overcharging produces hydrogen and oxygen gas via electrolysis of water; as these gases vent to the atmosphere, so the capacity for the battery to store charge is reduced. This means that the DataNode in its current form can destroy battery capacity through overcharging.

In addition, the panels are rated for a specific $V_{MP}$, the voltage at which maximum power is produced by the panel. There is also the corresponding $I_{MP}$ as well as $V_{OC}$ which specifies the open circuit voltage when no load is attached. In the existing design, as mentioned above, when the battery is in a discharged state any current flowing from the panel into the battery will force the panel output voltage to fall. Similarly, as the battery stores charge its terminal voltage will increase which in turn means that the panel output voltage will not fall as far. In fact, in the extreme case the battery terminal voltage will attempt to climb towards $V_{OC}$ - higher terminal voltages will increase the rate of electrolysis, however, which limits the practical terminal voltage limit - especially as the battery case begins to bulge. It can be seen that the ATtiny
cannot control the panel output in order to most efficiently derive power from the panel.

In terms of specifying desired outcomes, the DataNode should incorporate a charger that provides a constant current charging mode for bulk charge and a constant voltage charging mode for topping up the battery, while optionally allowing for a float mode also. Additionally the design should regulate the solar panel output in order to be able to make use of the maximum power available from the panel at any given time. The design should be able to charge the battery even if the Node is powered down. Finally, the charger should ensure that the battery terminal voltage never exceeds the float voltage specified by the battery manufacturer.

6.3 The Solar Charger Design

After reviewing the available options, it was decided that Linear Technology’s LT3652 “Power Tracking 2 A Battery Charger for Solar Power” IC would be used to form the basis for the design. The LT3652 is a step-down battery charger which takes input over the range of 4.95 to 32 V. It offers constant current, constant voltage and float charge modes, with charge current up to 2 A. Using an input voltage regulation loop allows the LT3652 to reduce the charge current when the input voltage falls below a set level, determined by an external resistor divider. In this way the LT3652 can maintain the peak output power of an attached solar panel. For further details, the datasheet has been attached in Appendix D on page 197.

Figure 6.1 shows the final circuit used in the prototypes. The charger is to be used with 5 or 10 W panels; the 5 W panel (Electus Distribution ZM9091) is rated for an open circuit voltage \(V_{OC}\) of 21 V, maximum power voltage \(V_{MP}\) of 16.8 V and a short circuit current \(I_{SC}\) of 0.38 A. The 10 W panel (Electus Distribution ZM9093) is rated for \(V_{OC}\) of 21 V, \(V_{MP}\) of 17 V and \(I_{SC}\) of 0.68 A. The SLA batteries to be used with the charger are 12 V batteries which utilise a float charge up to 14.4 V.

The process for specifying each component has been described in Appendix B.5 on page 100. The BOM is attached as Appendix B.6 on page 104. The components were ordered after checking the entire specification process for errors.
6.4 Production of the Prototype

With the components specified and ordered the next step was to capture the schematic. The EAGLE PCB design suite was chosen for this due to the author’s previous experience with the suite. After the process was complete it was learnt that access to Altium Designer was available through the University, although upon investigation the design process was no simpler in Designer since the University’s license had expired, requiring exotic components like the LT3652 to be entered in either suite.

With the schematic drawn, the PCB layout was completed. Before this was completed, a PCB manufacturer in Sydney offering 48hr turn around was chosen to produce the prototype PCBs. An enquiry into the University’s own capabilities showed that they were unable to provide the required PTH service. It was imperative to decide on a manufacturer before the design was finalised as the achievable tolerances, and therefore design parameters, are defined by the manufacturer. The final design can be seen in Figure 6.2.

The prototype PCBs arrived from the manufacturer within a day of the components arriving from the distributor. Initially two PCBs were populated and calibrated for use with the 10 W panel. A minor issue with the PCB design was discovered: from an early stage of the design the intention had been to use 0603 size LEDs for the circuit,
but the decision was later made to use 0805 LEDs. Although the BOM and schematic were modified to reflect this change, the footprint used in the PCB was not updated. Fortunately it is possible to mount an 0805 LED on 0603 pads with care, however this has been rectified in the PCB design and would not be an issue in production runs.

A slightly more problematic issue arose because the vias underneath the IC had been covered by solder mask. These vias were incorporated into the design because the LT3652 IC utilises an exposed ground pad on the underside of the package, in addition to the 12 pins along its sides. The idea behind the vias was to provide a path for heat to travel from one side of the board to the other, so that the entirety of the board could be hand soldered. Unfortunately there was no exposed pad on the underside of the PCB with which to transfer heat from the soldering iron to the other side of the board. Whether this issue would be changed for production runs depends on the method by which the PCBs will be populated. If populated by hand at Rimik, the design should be modified to simplify the production, but if populated via a pick and place machine and soldered with a reflow setup then it would be unnecessary to modify the design.

As a result of this issue, the LT3652 ICs were soldered to the prototype PCBs using a reflow technique performed with a hand held heat gun. The pads were given a light coating of solder and then the IC was tacked in place by soldering pins on diagonal corners, then flux was deposited on all of the pins. The PCB and IC were then brought up to temperature using the heat gun on a medium heat setting (6 on a scale of 10) for 90s. After this time the heat was increased to a high setting (9 on a scale of 10)
for 35 s in order to reflow the solder at the pins and underneath the IC. After this, the heat was reduced back to the medium setting and applied for 40 s. The heat was then removed and the PCB allowed to cool to room temperature. Each PCB was given a visual inspection under magnification to ensure that the solder had reflowed, then checked for shorted pins using a multimeter. 10 out of 10 PCBs were soldered with no electrical faults using this method.

To test the prototype chargers, a testbed was set up using a 10 W solar panel, a 12 V SLA battery and three multimeters. With a multimeter measuring the current through the solar panel and another measuring the current through the battery, the third multimeter was used to check the voltage across the battery and solar panel. The battery used was partly discharged before being attached to the charging circuit. The panel was attached to the charger, then voltages and currents were monitored to ensure that they were within expected bounds; the charger was trimmed to regulate the solar panel to 17 V, its $V_{MP}$. The current into the battery appeared to follow the expected curve, although maintaining a constant current with the solar panel proved impossible due to clouds moving across the face of the sun and shadows made by nearby trees and pedestrians. The battery voltage rose towards 14 V before dropping to approximately 13.5 V, while the current drawn also dropped away as the charger initiated its float charging phase.

The test was repeated with the second charger after slightly discharging the battery through a halogen globe. The second unit behaved in the same way as the first. Since both units gave promising initial results, the remaining 8 prototype boards were also populated. Please see Figure 6.3 for a picture of a completed charger PCB, and note the scale - the MSOP12 LT3652 is directly above the body of the platypus, below and to the right of the large aluminium capacitor.

### 6.5 Testing of the Prototype

In order to test the function of the solar panel controller and battery charger prototypes an interface board was designed so that the DataNode could be employed to log the voltage and current that the solar panel, battery and system were either using or providing, for a total of 6 data streams. For the voltage measurements, a simple
voltage divider was employed to scale the input voltages back to the range of 0 - 2.5 V as required by the Node’s ADC. A trimpot was employed to allow calibration of the voltage scale. An opamp employed as a voltage follower was used in order to keep the input impedance low enough for the Node to read.

For the current measurements, a high power shunt resistor was installed in the path to ground for the circuit being measured. The voltage potential across the shunt resistor was input into an opamp with trimpot adjustable gain, allowing calibration of the scale. A second voltage follower was used, again to match impedances.

The opamps used were the LM358D, a dual opamp IC in PDIP-8 package. The minimum supply voltage for the IC is 3 V, however the output voltage is $V_{CC} - 1.5$ V (ON Semiconductor 2013). Therefore, in order to achieve an output high voltage of 2.5 V, to match the ADC $V_{REF}$, the LM358D required a 4 V supply voltage. This was achieved using an LM317 adjustable voltage regulator, which is capable of supplying up to 1.5 A of current (Fairchild Semiconductor 2002). By using a trimpot, the output voltage is able to be adjusted to ensure that the LM358D opamps are producing the required 2.5
Four of these interface boards were made on prototyping stripboard, see Figure 6.4. This would allow each node to use six of its IO channels to log voltage and current data. The remaining two channels were set up as digital outputs in order that small incandescent globes could be attached to each Node in order to simulate loads. Each globe represents at least 100 mA or current load on the system, significantly more than a typical DataNode installation would constantly see.

Four Nodes were then set up in a testbed to measure how the solar controllers behave. Firstly, a Node was installed as a control: a 10 W panel and a 12 V 7 Ahr battery were connected to the interface board and then to the Node in the usual manner. This Node would exemplify the operation of the existing charging system. A second Node was set up in an identical manner, excepting that a prototype solar charger PCB was used instead of the Node’s charging circuit. It should be noted that the original charging circuit was not removed from the system, so any load that it represents will remain on the Node. The remaining Nodes were set up in the same way as the second 10 W Node, excepting that one used a 5 W panel and the other a 20 W panel.

Comparisons between the first two Nodes will provide insight into the differences between the solar panel control and battery charging implementations. Comparisons between the second, third and fourth Nodes will show the impact of the different sized panels on the Node’s ability to remain working under heavy load.

The Nodes were all set up in a similar manner: the solar panels all faced in the same direction and were positioned at the same angle with respect to the ground. The lamp on each Node was left lit during the entire test, and for one day the second lamp was lit in order to provide an even greater load. The Nodes, interface boards, batteries and prototype chargers were placed under a plastic cover to keep out dust, water and wildlife. The testbed was covered by a heavy plastic sheet overnight to keep the dew and rain off the units. The cover was removed at some point between 6 and 7 each morning and replaced between 5 and 6 each afternoon; reading the data that was being logged showed that the solar panels were providing no appreciable charge at those times.
6.6 Test Results

All the Nodes were able to record useful voltage data. However, the ability to record accurate current data was rendered basically void due to being unable to calibrate the circuit accurately. After a resistive load failed to provide enough accuracy to calibrate the opamp gain, a constant current source was devised and built. Its usefulness as a means of calibrating the current measuring circuits remains untested due to running out of time. The logged data from the Nodes can be found in Appendix B.8, beginning on page B.8.

Figure 6.5 shows the results of the first round of data analysis. For each column of data, representing the measured data streams recorded by each Node, the maximum and minimum values were found. For the control Node, the battery terminal voltage rose above 15 V over the course of testing and fell to 9.1 V, while over the same period of time the maximum voltage recorded from the solar panel was 16.2 V. During the night the solar panels produce no power, which is why the minimum in each case was 0 V.
These results prove two hypotheses Rimik had posed regarding the operation of the Node in its current form; firstly, that the control scheme does allow damagingly high battery terminal voltages to occur, which will definitely reduce the capacity of the battery due to the nature of the SLA battery. Secondly, that the control scheme cannot make the most efficient use of the solar panel - recall that the $V_{MP}$ rating for the panel used is 17 V.

Considering the three Nodes which used the prototype solar panel controller and battery charger, the most obvious data point is the maximum voltage that the Nodes recorded the solar panel reaching. Considering that the $V_{OC}$ rating for the panels used is 21 V, this means for at least part of the time across all three Nodes, the charger determined that there was insufficient current flowing to direct the power from the panel to the Node. Upon review of the logged data, it can be seen that through the mid morning each day the panel voltage rises fairly quickly to around $V_{OC}$, but that the battery voltage levels rise while the panel output voltage is down around 17 V and remain steady when the panel voltage is up around 21 V. This would appear to address the battery charger criterion of making efficient use of the solar panel, although it is not possible to state that the controller makes most efficient use of the panel in the absence of the current data. Thus further testing must be performed, once the calibration scheme has been finalised and tested.

Another reassuring data point is the maximum voltage recorded across the battery on all three Nodes: during the time that these Nodes were under test, not once did the terminal voltage rise above the manufacturer’s specification for the battery. This successfully addresses the specification that the controller not over charge the battery.

The data for the Node with 5 W panel shows conclusively that this panel and battery combination are not powerful enough to keep the Node working with the two incan-
6.6 Test Results

descent bulbs as load. No current installations of DataNodes uses this much power, although without the current measurements it is difficult to determine how much “this much” is. Another important result that the logger managed to record is exactly what can happen with a DataNode that cannot turn itself off: the second globe was lit at 5:50 on the 17th, draining the battery at a fast rate. The analogue comparator failed its comparison against 9.25 V between 22:58 and 23:00 that evening, although it passed the comparison the next time through even though it is clear from the logged data that the battery level was still falling - incorporating some hysteresis into this reading might resolve this issue, and this is also further justification for incorporating a dedicated battery voltage measurement in the upgrade specification, since the historical data could have been used to show the continuing downward trend.

Setting those considerations aside for the moment, it can be seen that the Node continued to operate for another two hours and reaching 6.5 V at 1:00AM. At this point the Node somehow managed to trigger at least another analogue comparator pass, followed by another fail. After the failed attempt, the Node should have switched off inside a few seconds as the input capacitors drained. However, since the Node has no ability to shut itself down, it continued to operate for at least another 4 minutes until the input voltage to the 5 V switchmode regulator passed below its under voltage lockout threshold.

There are two factors to note here; firstly, that the battery has sustained permanent damage long before the Node finally turned off. Secondly, that the specification that the Node upgrade prevent over discharge of the battery should not rely on a scheme with a single point of failure. It would seem a reasonable inference that for the existing design a decision was made during development that the MAX5033’s under voltage lockout feature would be ignored because the other hardware was designed in such a way as to provide a means for the Node to power itself down through code written into the firmware. The modification to the Node which breaks this functionality is as simple as cutting two traces and installing two jumper wires. The situation highlights the importance of the point made in Section 4.2.3; that no change should be made to the design without working the issue through from the original specification and specifiers. Regardless, this is a prime example of a case where, for the want of a resistor worth less than a cent, an embarrassing, potentially expensive and ultimately avoidable fault has passed through the many stages of system design into a product on the market. It
also serves as a timely reminder to question the assumptions that are made in design and to endeavour to anticipate the use (and misuse) of equipment.
The existing solar panel control and battery charging scheme was analysed in light of fundamental issues in its implementation. Specifications for desired operation were defined. A design was proposed for a charger which meets the specifications for efficient use of the attached panel; charging while the Node is not powered on; offers a multiple mode charging scheme covering constant current, constant voltage and float charging phases; ensures that battery over voltage conditions are avoided. 10 prototypes of the design were constructed and three of these were used in conjunction with the DataNode system, an analogue interface board, an SLA battery and a solar panel in comparative testing against a fourth Node equipped with the analogue interface board, SLA battery and 10 W solar panel.

The results from the comparative testing proved the issues with the existing design as well as confirming that the prototype design successfully meets the specifications put forth for an upgraded solar panel controller and battery charging scheme in regards to the efficient use of the solar panel and ensuring that the battery over voltage conditions are avoided. The recorded data was not conclusive in proving that the design charges the battery while the Node is powered off, however the initial testing of the first prototypes, performed without a Node attached, indicates that it will. Further testing and analysis is required.

It is worth noting that the design of the prototype lends itself to retrofit operations on existing Nodes. Although this would not solve issue that these Nodes may have in shutting themselves down, it would resolve the remaining issues.
Chapter 7

The DataNode Upgrade

7.1 Chapter Overview

With the MCU recommendation given and a solid prototype design presented, the remaining work is to describe the plan for designing the DataNode system upgrade and detailing how the system specifications outline in Section 4.2.2 will be addressed.

7.2 Remaining Specifications

The two sections of the specification which have not been related to either the MCU selection or the solar panel controller and battery charger prototype are the ‘power’ sections specifying that the Node be able to guarantee being able to power itself down and specifying that there is a lower limit for discharging the battery that must be adhered to. All of the remaining ‘MCU’ specifications flow on from the decision about which MCU to recommend, since the specific parameters, capabilities and pinouts are dependent on the MCU in question. As for the remaining ‘power’ specifications, they have been addressed by the prototype charger board.
7.3 Analysis

7.3.1 Low Voltage Shut Off

Now, regarding the requirement to ensure that there is a lower voltage limit such that the Node will switch off when the value is reached, without having to rely on the firmware to catch the scenario. The datasheet for the MAX5033 shows how the Under Voltage Lock Out can be programmed by connecting a resistor divider between \( V_+ \) and GND and connecting the ON/\( \overline{OFF} \) pin to the node between the resistors. The formula for determining the under voltage lockout threshold is given on page 9 of the datasheet, attached as Appendix C.1:

\[
V_{UVLO(TH)} = \left( 1 + \frac{R_1}{R_2} \right) \times 1.85 \tag{7.1}
\]

So, assuming a desired cut out at 9.25 V;

\[
\frac{V_{UVLO(TH)}}{1.85} - 1 = \frac{R_1}{R_2} \tag{7.2}
\]

\[
\frac{R_1}{R_2} = \frac{9.25}{1.85} - 1 \tag{7.3}
\]

\[
\frac{R_1}{R_2} = 4 \tag{7.4}
\]

\[
R_1 = 4 \times R_2 \tag{7.5}
\]

If E12 series resistors are available, choose for example 390 k\( \Omega \) for \( R_1 \), 100 k\( \Omega \) for \( R_2 \). This gives

\[
V_{UVLO(TH)} = \left( 1 + \frac{390k\Omega}{100k\Omega} \right) \times 1.85 \tag{7.6}
\]

\[
V_{UVLO(TH)} = 9.065 V \tag{7.7}
\]

The existing design already employs a 100 k\( \Omega \) resistor between \( V_+ \) and the ON/\( \overline{OFF} \) pin. Replacing this with a 390 k\( \Omega \) resistor and adding a board mod for the 100 k\( \Omega \) to GND would allow this modification to be made to the existing Nodes, raising the under voltage lock out threshold from 5.2 V to 9.065 V: this would reduce the permanent damage batteries suffer through over discharge, giving them a longer working life. It would also guarantee the Node can power off shortly after the analogue comparator
7.3 Analysis

test fails: considering the data from the charger testing, Node1 (Appendix B.9) took a mere two minutes to fall from 9.3 V to 9.1 V under the load of the two incandescent globes.

However, it is worth considering the datasheet for the battery, Appendix C.2, which suggests that for current draw up to 650 mA, battery discharge should stop at 10.5 V. R\textsubscript{14}, the analogue comparator voltage divider upper resistor, could be changed from 270 kΩ to 330 kΩ. This would result in the comparator failing its comparison when V\textsubscript{+} fell below 10.75 V. Choosing 470 kΩ for R\textsubscript{1} and 100 kΩ for R\textsubscript{2} gives

\[
V_{UVLO(TH)} = \left(1 + \frac{470k\Omega}{100k\Omega}\right) \times 1.85 \quad (7.8)
\]

\[
V_{UVLO(TH)} = 10.545 V \quad (7.9)
\]

Comparing these values with the data collected on Node2 in the testbed, attached as Appendix B.14, it can be seen that with both incandescent globes providing load to the system, the Node measured battery voltage of 10.746 V at 0:16AM on the 19th. At 1:00AM on the 19th, the Node measured the voltage falling to 10.541 V - 44 minutes to pass between both the suggested new analogue comparator threshold and the suggested under voltage lock out threshold for the MAX5033, a difference of 0.2 V. Interestingly, the Node stayed on for only and extra 36 minutes after passing 10.541 V, shutting off at 1:36AM and 10.008 V - 36 minutes to lose 0.54 V across the battery, while the load remained the same. This suggests that at this point in the battery’s charge cycle it is rapidly approaching the “dead flat” state.

With this in mind, it is recommended that both the modifications to the analogue comparator resistor divider and the extra resistor added between the ON/OFF pin and GND be implemented in the system redesign. This would provide the least damaging environment for the battery, provided it was implemented in conjunction with the prototype solar charger design.

7.3.2 Node Initiated Shutdown

Further reading of the under voltage lock out section of the MAX5033 datasheet shows that bringing the ON/OFF pin down to GND potential will put the device into a shutdown mode. This, however, is no improvement over the current situation; the lim-
7.3 Analysis

Iteration on the Node being able to shut itself off is because there are insufficient IO pins for additional control. As such, in order to resolve the final outstanding system design specification an MCU with additional IO pins is required. Therefore, this requirement will not be able to be addressed until the rest of the system design is undertaken.
It has been demonstrated how, through modifying the existing circuitry by swapping
one resistor for a different value and adding another resistor as a board mod, one of the
outstanding system design specifications could be satisfactorily addressed. With the
design for the solution complete, it remains to apply the chosen engineering method:
namely, to implement the solution and test.
Chapter 8

Conclusions and Further Work

The aim of the project was to design a suitable upgrade to the existing DataNode system. To this end, a traditional Engineering Design and Development Cycle was chosen as the method by which the project would be performed. The Conception phase encompassed the survey performed and the investigation of the capabilities of similar systems, highlighting the flash and RAM capacities, number of IO pins and number of communications interfaces as the hardware impetus for upgrading the MCU, while implementing further communications protocols, namely Modbus and TCP/IP, were recognised as the software impetus for upgrade. In addition, several deficiencies in the way the existing design managed the solar panel, battery and power rails within the Node provided further momentum.

A selection of MCUs were analysed in order to provide a recommendation for an upgraded controller for the Node. The Freescale MKL25Z128VLH4 was chosen, having performed well against its competition and the specification criteria.

A prototype solar panel controller and SLA battery charger was developed in response to the specification criteria. Initial testing showed that it was capable of charging slightly discharged batteries and, importantly, not subjecting them to over charge conditions. On the back of those positive initial results an extended testbed was devised using a specially designed analogue interface board in order to measure the voltage and current levels of the Node, battery and solar panel. 3 Nodes, attaching a prototype charger along with their 5 W, 10 W and 20 W panel along with their battery, analogue interface board and Node were pitted against a control Node, with solar panel, bat-
tery and analogue interface board to determine how, if at all, the solar charge board improved over the Control.

The results from the logged data were promising, although only the voltage measurements proved useful - two methods of calibrating the current circuits were devised, with the first failing to provide any accuracy or repeatability. The second one remains untested due to halting testing in order to finalise the project writing.

The data conclusively shows how the DataNode can destroy SLA batteries, validating one aspect of the prototype design straight away: with the prototype charger installed, battery voltages rose to a maximum of just under 14 V, which is within the manufacturer’s specifications. On the control Node, battery voltages reached 15.75 V, well above the manufacturer’s specification.

The data additionally showed that the solar panel controller worked to improve the efficiency of the charging circuit by regulating the output from the solar panel in order to operate close to the panel’s peak power point.

Of the two system design specifications that had not been addressed, analysis of the existing design has resulted in a design to be implemented in order to solve one. The Node currently has no low voltage drop out capability; this results in the Node running batteries completely flat, down to around 5.2 V, and destroying them. By adding a single resistor as a board mod, it should be possible to resolve this issue by programming the 5 V regulator’s under voltage lock out threshold to a higher voltage than the default 5.2 V. This way, as soon as the voltage drops below the threshold the Node will shut down, saving the battery.

At the same time, the recommendation has been made that the Node’s analogue comparator set up be changed to reflect the battery manufacturer’s guidelines for acceptable discharge levels in its batteries. Modifying the analogue comparator threshold to 10.75 V from 9.25 V would require changing the value of one resistor. Raising the comparator threshold to this value would allow the voltage regulator’s threshold to be lifted to just over 10.5 V Performing this pair of small changes would result in better service life out of the batteries.
8.1 Achievement of Project Objectives

The following objectives have been addressed:

Chapter 2 Perform a survey of existing users and define the attributes of the system that are appreciated, and what areas for improvement exist; Survey the state of the market to determine what competitors exist and what they offer; Evaluate the DataNode platform, in light of the completed surveys

Chapter 4 Develop selection criteria which an upgrade to the microprocessor and associated hardware, if deemed necessary, would need to address

Chapter 5 Based on the selection criteria, investigate available MCU families with the view of recommending a MCU to underpin the upgraded node

Chapter 6 If time permits, prepare a prototype for testing

8.2 Further Work

In the short term, the solution to the battery over discharge and Node shutdown issues suggested in Chapter 7 can be implemented and tested to discern whether or not it solves the problems. If it can, this solution would be incorporated into the design of the upgraded DataNode.

Similarly, testing will continue for the solar panel controller and battery charger prototype designed and implemented in Chapter 6. The initial results look promising, with the prototype devices providing much better control over the battery than the control Node.

Finally, the major work yet to be completed is the design of the upgraded Node system, utilising the recommended MCU and applying the suggested system specifications. This will require schematic capture, PCB design and verification, then PCB prototypes to be made and assembled. With prototypes on hand, the software design aspect will come to the fore as device drivers are written and the code base migrated to the new system. Once that process begins, physical tests of the system can proceed, working
8.2 Further Work

to demonstrate that the design fulfils the specifications set forth and cycling through
the Engineering Design and Development Cycle as necessary.
References


Ganesan, P. (2003), ‘Efficiently adding secure communications to networked low-end embedded systems using software thread integration’.


Appendix A

Project Specification
ENG 4111/2 (or ENG8002) Research Project

Project Specification

For:  STEPHEN ELLIS
Topic:  UPGRADE OF DATALOGGER PLATFORM
Supervisors:  H. ZHOU
Sponsorship:  Rimik Pty Ltd

Project Aim:  To determine under what circumstances an upgrade to the existing DataNode system would be justified. Following on from that determination, to provide a recommendation for the microcontroller such an upgrade would be based on.

Program:

1. Perform a survey of existing users and define the attributes of the system which are appreciated, and what areas for improvement exist.

2. Survey the state of the market to determine what competitors exist and what they offer.

3. Evaluate the DataNode platform in light of the completed surveys

4. Develop selection criteria which an upgrade to the platform would need to address

5. Based on the selection criteria, recommend a suitable base for the upgrade

As time and resources permit:

1. Prepare a prototype for testing

Agreed:

Student Name:  Stephen Ellis
Date:  20 March 2013

Supervisor Name:  Hong Zhou
Date:  20 March 2013

Examiner/Co-Examiner:
Date:
Appendix B

Some Supporting Information
B.1 Survey Results

Rimik’s existing customers were asked a short series of questions in order to help gain an understanding of what users are looking for in a data logging unit. In addition, during this year’s AgShow and FarmFest field days these same questions were put to attendees expressing interest in the DataNode system. The responses have been collated and are attached below; please see the discussion in section 2.3 on page 7.
Internal Discussion Results:

Limitations identified in the existing node:
- Limited code space for complex functions – cannot fit all current code.
- Limited data space for storing data structures such as historical weather data – cannot fit all current structures.
- Limited to 8 I/O channels
  - 4 non configurable without hardware modes
  - 4 configurable via jumpers
  - Using \textit{I}^2\textit{C} for connecting sensors must be done in software, rather than hardware, and ties up two IO channels
- Current battery charging system destroys batteries; batteries end up over charged and are able to be run dead flat.

Desirables for upgraded node:
- USB/SD connectivity for data download
- Display options; 16x2 character LCD or much larger eg 160x128 pixels, graphical LCD with touch, etc
- Ability for the node to power itself down and up again
- Need to be able to handle battery voltage logging and control of unit via set points
- Bypass charging (always on charge functionality)
- Improved/finer control over power supply rails for both 5V and 12V to individual channels
- Capacity to perform analogue output calculations based on analogue input or predetermined set points (rate control)
- Reduce cost of connectors by changing to screw connection
- Move away from dedicated 3G modems towards integrating a 3G module
- All inputs configurable for digital, frequency, 0-5Vand 4-20mA
- All outputs capable of digital output
- Some or all outputs capable of analogue output
- Dual RS232 channels
- Wireless connectivity (meshed) at the same time as IP Modem
- Ethernet connectivity
- Bluetooth connectivity?
Questions asked of existing and prospective customers:

What tasks in your business which can be monitored and controlled remotely would return the highest value?

What timely data is required in your business to help you make decisions?

What tasks require the most time and data analysis in order to make a correct decision?

Responses:

Limitations:
- Current remote access via modem at 9600 Baud rate is expensive to operate
- No wireless access - could use Bluetooth, radio, or internet
- Difficult download methodology - could use USB stick or wireless access instead of RS232

Desirables:
- Alert systems for:
  - Tanks (Level set points)
  - Crop spraying (Delta T and wind speed/direction set points)
  - Lucerne management (Dew Point limits)
  - Soil Moisture (low set points for irrigation requirement)
  - Via - SMS, Radio, Internet
- Control Systems
  - Tank level control coupled with remote pump control
  - Channel level control and steam flow management
  - Pump control
  - Stream flow management
  - Proportional valve control in addition to existing digital output
  - Temperature control systems for refrigeration units and heating chambers
  - Rate controllers (variable input and variable output) to replace old Rilmik systems
- SCADA type screen display on PC with regular updates of values - need local network
- Want easier access to setup and data provision
  - Radio and Internet
- Phone App would be good for checking current via internet
- Access to Node via mobile phone app through direct connection
- Touch Screen with graphical display of logged data
- Easier menu for navigation via touch screen
- Valve control
- Dual RS232 for input of GPS and other function such as Radar Distance measurement (both NMEA output) or EM unit
B.3 Selection Matrix

Attached below is the matrix that was generated in order to select an appropriate MCU to replace the existing ATmega644PA MCU. The methodology behind how each point of comparison was weighted has been discussed in section 5.3 on page 60.

The information in the table was drawn directly from the appropriate datasheets; links to these may be found in the references. (Atmel 2010b), (Atmel 2013c), (Atmel 2013b), (Atmel 2013a), (Atmel 2013d), (Freescale 2010a), (Freescale 2012c), (Freescale 2010c), (Freescale 2010b), (Freescale 2013), (Freescale 2012b), (Freescale 2012a).
<table>
<thead>
<tr>
<th>MCU</th>
<th>ATmega644PA-AU</th>
<th>ATmega1284P-AU</th>
<th>ATmega128C3U</th>
<th>ATmega128A3U</th>
<th>ATmega192C3U</th>
<th>ATmega192A3U</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>8 bit AVR</td>
<td>8 bit AVR</td>
<td>8 bit AVR</td>
<td>8 bit AVR</td>
<td>8 bit AVR</td>
<td>8 bit AVR</td>
</tr>
<tr>
<td>Max Frequency (MHz)</td>
<td>20</td>
<td>20</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>Flash (KB)</td>
<td>64</td>
<td>128 +</td>
<td>128 +</td>
<td>128 +</td>
<td>192 +</td>
<td>192 +</td>
</tr>
<tr>
<td>EEPROM (KB)</td>
<td>2</td>
<td>4 +</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>SRAM (KB)</td>
<td>4</td>
<td>16 +</td>
<td>8 +</td>
<td>8 +</td>
<td>16 +</td>
<td>16 +</td>
</tr>
<tr>
<td>Pin count</td>
<td>44</td>
<td>44</td>
<td>64 +</td>
<td>64 +</td>
<td>64 +</td>
<td>64 +</td>
</tr>
<tr>
<td>Max I/O pins</td>
<td>32 +</td>
<td>32</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
</tr>
<tr>
<td>ADC (channels, resolution [bits])</td>
<td>8, 10</td>
<td>8, 10</td>
<td>16, 12</td>
<td>16, 12</td>
<td>16, 12</td>
<td>16, 12</td>
</tr>
<tr>
<td>UART</td>
<td>2</td>
<td>2</td>
<td>3 +</td>
<td>3 +</td>
<td>3 +</td>
<td>3 +</td>
</tr>
<tr>
<td>SPI</td>
<td>3</td>
<td>3</td>
<td>5 +</td>
<td>10 +</td>
<td>5 +</td>
<td>10 +</td>
</tr>
<tr>
<td>TWI / I2C</td>
<td>1</td>
<td>1</td>
<td>2 +</td>
<td>2 +</td>
<td>2 +</td>
<td>2 +</td>
</tr>
<tr>
<td>Timers</td>
<td>3</td>
<td>3</td>
<td>5 +</td>
<td>7 +</td>
<td>5 +</td>
<td>7 +</td>
</tr>
<tr>
<td>WDT</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Analogue comparator</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Power supply</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>USB</td>
<td>no</td>
<td>no</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Power consumption</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
</tr>
<tr>
<td>* Active (mA)</td>
<td>0.4</td>
<td>0.4</td>
<td>0.25 +</td>
<td>0.25 +</td>
<td>0.25 +</td>
<td>0.25 +</td>
</tr>
<tr>
<td>* Power down (µA)</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>* Power save (µA)</td>
<td>0.6</td>
<td>0.6</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>pricing [*20% off, comparative]</td>
<td>4.64</td>
<td>5.36</td>
<td>2.62 +</td>
<td>5 +</td>
<td>3.28 +</td>
<td>5 +</td>
</tr>
<tr>
<td>* Digiby</td>
<td>5</td>
<td>4.64</td>
<td>5.36</td>
<td>2.62 +</td>
<td>5 +</td>
<td>3.28 +</td>
</tr>
<tr>
<td>* Element</td>
<td>3</td>
<td>2.07</td>
<td>12.86</td>
<td>not stocked</td>
<td>3</td>
<td>10.90</td>
</tr>
<tr>
<td>* Mouser</td>
<td>5</td>
<td>6.26</td>
<td>6.80</td>
<td>not stocked</td>
<td>5 +</td>
<td>3.93</td>
</tr>
<tr>
<td>* RS Components</td>
<td>5</td>
<td>5.85</td>
<td>17.50</td>
<td>not stocked</td>
<td>5 +</td>
<td>5.64 +</td>
</tr>
</tbody>
</table>

Tallies:
- [ >14 highlighted ]
- [ >10 highlighted ]
- [ >6 highlighted ]

### B.4 MCU Selection Matrix

<table>
<thead>
<tr>
<th>Atmel</th>
<th>ATUC128L4U</th>
<th>ATUC128L3U</th>
<th>ATUC256L4U</th>
<th>ATUC256L3U</th>
<th>ATSAM4L2C</th>
<th>ATSAM4L2C</th>
<th>ATSAM4L2C</th>
</tr>
</thead>
<tbody>
<tr>
<td>32 bit AVR</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
<td>50 +</td>
</tr>
<tr>
<td>Max Frequency (MHz)</td>
<td>128 +</td>
<td>128 +</td>
<td>256 +</td>
<td>256 +</td>
<td>128 +</td>
<td>128 +</td>
<td>128 +</td>
</tr>
<tr>
<td>Flash (KB)</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 MHz 1.8V 25°C</td>
<td>48 +</td>
<td>48 +</td>
<td>48 +</td>
<td>48 +</td>
<td>48 +</td>
<td>100 +</td>
<td>100 +</td>
</tr>
<tr>
<td>RAM (KB)</td>
<td>5 +</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>1 MHz 1.8V 25°C</td>
<td>4 +</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>USB</td>
<td>no</td>
<td>no</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Power consumption</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
<td>1 MHz 1.8V 25°C</td>
</tr>
<tr>
<td>* Active (mA)</td>
<td>0.1 +</td>
<td>0.1 +</td>
<td>0.1 +</td>
<td>0.1 +</td>
<td>0.1 +</td>
<td>0.1 +</td>
<td>0.1 +</td>
</tr>
<tr>
<td>* Power down (µA)</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
</tr>
<tr>
<td>* Power save (µA)</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>pricing [*20% off, comparative]</td>
<td>5.93</td>
<td>6.64</td>
<td>6.99</td>
<td>7.28</td>
<td>4.10 +</td>
<td>5.95 +</td>
<td>3.92 +</td>
</tr>
<tr>
<td>* Digiby</td>
<td>5 +</td>
<td>5 +</td>
<td>6.40 +</td>
<td>6.99</td>
<td>7.28 +</td>
<td>4.10 +</td>
<td>5.95 +</td>
</tr>
<tr>
<td>* Element</td>
<td>5 +</td>
<td>5 +</td>
<td>11.47 +</td>
<td>11.47 +</td>
<td>14.68 +</td>
<td>14.68 +</td>
<td>14.68 +</td>
</tr>
<tr>
<td>* Mouser</td>
<td>5 +</td>
<td>5 +</td>
<td>8.68 +</td>
<td>9.81 +</td>
<td>9.81 +</td>
<td>9.81 +</td>
<td>9.81 +</td>
</tr>
<tr>
<td>* RS Components</td>
<td>5 +</td>
<td>5 +</td>
<td>13.60 +</td>
<td>14.68 +</td>
<td>14.68 +</td>
<td>14.68 +</td>
<td>14.68 +</td>
</tr>
</tbody>
</table>

Tallies:
- [ >14 highlighted ]
- [ >10 highlighted ]
- [ >6 highlighted ]

### Notes
- **RS Components**: RS Components pricing (~20% off, comparative)
- **Power save (µA)**: Power save (µA)
- **Device**: Device
- **WDT**: WDT
- **SPI**: SPI
- **USB**: USB
- **Max IO pins**: Max I/O pins
- **ADC**: ADC (channels, resolution [bits])
- **SRAM**: SRAM (KB)
- **EEPROM**: EEPROM (KB)
- **Flash**: Flash (KB)
- **Pin count**: Pin count
- **Max Frequency (MHz)**: Max Frequency (MHz)
- **Architecture**: Architecture

### Pricing
- **$**: Price
- **$** $**: Price (25% off, comparative)
- **$** $**: Price (50% off, comparative)
- **$** $**: Price (75% off, comparative)
- **$** $**: Price (90% off, comparative)

### Power Consumption
- **USB**: USB
- **Analogue comparator**: Analogue comparator
- **WDT**: WDT
- **SPI**: SPI
- **TIMERS**: Timers
- **ADC**: ADC (channels, resolution [bits])
### B.4 MCU Selection Matrix

#### Freescale

<table>
<thead>
<tr>
<th>Device</th>
<th>128 x 64 LCD Controller</th>
<th>MCF51G125CLH</th>
<th>MCF51G125CLF</th>
<th>MCF51G128CLH</th>
<th>MCF51G128CLF</th>
<th>MCF51G128CXL</th>
<th>MCF51G128CHF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
</tr>
</tbody>
</table>

#### ARM Cortex M0+

<table>
<thead>
<tr>
<th>Device</th>
<th>128 x 64 LCD Controller</th>
<th>MCF51G125CLH</th>
<th>MCF51G125CLF</th>
<th>MCF51G128CLH</th>
<th>MCF51G128CLF</th>
<th>MCF51G128CXL</th>
<th>MCF51G128CHF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
</tr>
</tbody>
</table>

#### 32-bit ColdFire V1

<table>
<thead>
<tr>
<th>Device</th>
<th>128 x 64 LCD Controller</th>
<th>MCF51G125CLH</th>
<th>MCF51G125CLF</th>
<th>MCF51G128CLH</th>
<th>MCF51G128CLF</th>
<th>MCF51G128CXL</th>
<th>MCF51G128CHF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
</tr>
</tbody>
</table>

#### 8-bit S08

<table>
<thead>
<tr>
<th>Device</th>
<th>128 x 64 LCD Controller</th>
<th>MCF51G125CLH</th>
<th>MCF51G125CLF</th>
<th>MCF51G128CLH</th>
<th>MCF51G128CLF</th>
<th>MCF51G128CXL</th>
<th>MCF51G128CHF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
</tr>
</tbody>
</table>

#### 128x64 LCD Controller

<table>
<thead>
<tr>
<th>Device</th>
<th>128 x 64 LCD Controller</th>
<th>MCF51G125CLH</th>
<th>MCF51G125CLF</th>
<th>MCF51G128CLH</th>
<th>MCF51G128CLF</th>
<th>MCF51G128CXL</th>
<th>MCF51G128CHF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
<td>32 bit ColdFire V1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
<td>ARM Cortex M0+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
<td>not stocked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
<td>$3.54</td>
</tr>
</tbody>
</table>
The method for specifying each component will be described. The equations shown are from Linear Technologies’ LT3652 datasheet. The solar panel $V_{OC}$ will be used for the input voltage, $V_{IN}$, at 21 V. $V_{BAT}$ will be 14.4 V. $I_{CRG(MAX)}$ will be 1 A.

### B.5.1 C\textsubscript{1}

The input capacitor, $C_1$, decouples the switched supply current and therefore needs sufficient ripple current rating, where

$$I_{C1(RMS)} \cong I_{CHG(MAX)} \times \frac{V_{BAT}}{V_{IN}} \times \left( \frac{V_{IN}}{V_{BAT}} + 1 \right)^{\frac{1}{2}} \quad (B.1)$$

$$= 1 \times \left( \frac{14.4}{21} \right) \times \left( \frac{21}{14.4} - 1 \right)^{\frac{1}{2}} \quad (B.2)$$

$$= 0.464 \text{ A} \quad (B.3)$$

The bulk capacitance of $C_1$ determines the level of ripple on the input voltage, $\Delta V_{IN}$, where the capacitance in $\mu$F is given by

$$C_{IN(BULK)} = \frac{I_{CHG(MAX)} \times \frac{V_{BAT}}{V_{IN}}}{\Delta V_{IN}} \quad (B.4)$$

$$= \frac{1 \times 14.4}{21} \frac{1}{\Delta V_{IN}} \quad (B.5)$$

Linear Technology recommends that input ripple be kept less than 0.1 V. So the minimum capacitance for $C_1$ would be

$$C_{IN(BULK)} = \frac{14.4}{21} \frac{1}{0.1} \quad (B.6)$$

$$= 6.86 \mu F \quad (B.7)$$

Choosing a 10 $\mu$F capacitor will give input voltage ripple

$$\Delta V_{IN} = \frac{14.4}{C_{IN(BULK)}} \quad (B.8)$$

$$= \frac{14.4}{21} \frac{1}{10} \quad (B.9)$$

$$= 68.6 \text{ mV} \quad (B.10)$$
This level of ripple is within the acceptable range.

A low ESR aluminium electrolytic Nichicon RSS1E100MCN1GS was chosen for $C_1$.

### B.5.2 $R_3$

The charge current is determined by inductor sense resistor $R_3$; the voltage drop across the sense resistor at the desired current will be 100 mV. So given $I_{CRG(MAX)}$ will be 1 A,

$$R_3 = \frac{0.1}{I_{CHG(MAX)}}$$  \hspace{1cm} (B.11)

$$= \frac{0.1}{1}$$  \hspace{1cm} (B.12)

$$= 0.1 \Omega$$  \hspace{1cm} (B.13)

### B.5.3 $C_2$, $D_3$ and $D_4$

The boost supply rail is bootstrapped with $C_2$, $D_3$ and $D_4$ and operates of the range of 0 V to 8.5 V. A $1 \mu$F capacitor was chosen for $C_2$, while a 1N914BTR small signal diode was chosen for $D_3$. Because the battery voltage will be higher than the upper operating range voltage of the BOOST pin on the LT3652, a 6V2 Zener diode was chosen for $D_4$ to reduce the operating range to within 8.5 V.

### B.5.4 $D_1$

The input diode.

### B.5.5 $C_3$ and $C_4$

Linear recommend a $10 \mu$F ceramic to decouple the BAT pin, and for long cables to the battery a $100 \mu$F chip tantalum, with a voltage rating that exceeds $V_{BAT}$. 
B.5.6 \( L_1 \)

A 15\( \mu \)H inductor with saturation current above 1.15 A and RMS current above 1 A was chosen.

B.5.7 \( D_2 \)

The rectifier diode needed to exceed a current rating of 0.5s A.

B.5.8 \( R_4 \) and \( R_5 \)

Either two or three resistors could be specified for the feedback network, with the preference being for two resistors. As such working through the proscribed equations yield values of 1.1 M\( \Omega \) for \( R_4 \) and 324 k\( \Omega \) for \( R_5 \).

B.5.9 \( R_1 \) and \( R_2 \)

The input resistors are used to define \( V_{MP} \) for the solar panel. In order to account for 5 W, 10=,W and 20 W panels, \( R_1 \) was chosen to be made up of a 510 k\( \Omega \) fixed resistor and a 47 k\( \Omega \) trimpot, while 100 k\( \Omega \) was chosen for \( R_2 \).

B.5.10 \( R_7 \) and \( R_8 \), \( D_5 \) and \( D_6 \)

Simple yellow SMT LEDs were chosen in an 0805 package. Accounting for the voltage drop across the resistors, the meant a 1.9 k\( \Omega \) resistor would be used for each LED.

B.5.11 \( C_5 \)

By specifying a capacitor here, the IC knows to use the Timer charge termination method. As such the value of the capacitor sets the time out period: this design called for a 4.7 \( \mu \)F.
Finally, using a 1 kΩ resistor in line with the battery temp thermistor, a 10 kΩ, $B=3380$ NTC thermistor, allows the charger to charge the battery up to 45°C.
The below is spreadsheet for the Solar Charger design bill of materials. Component prices have been included as a guide only, with no attempt to optimise the cost. It is the author’s experience that Element14 is generally more expensive than competing suppliers such as R S Components or X-ON and it is expected that such would be used for production runs.
<table>
<thead>
<tr>
<th>Item Name/Num</th>
<th>PCB Location</th>
<th>Description</th>
<th>Part#</th>
<th>Supplier</th>
<th>Qty</th>
<th>U/M</th>
<th>Cost</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>ON Semi MBRS140T3G Schottky 1A 40V DO-214AA</td>
<td>9557237</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.55</td>
<td>$0.55</td>
<td>y y y</td>
</tr>
<tr>
<td>D2</td>
<td>ON Semi NSR0320MW2T1G Schottky rectifier 1A 23V SOD-323</td>
<td>1714826</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.24</td>
<td>$0.24</td>
<td>y y y</td>
</tr>
<tr>
<td>D3</td>
<td>Fairchild 1N914BTR small signal 100V 300mA</td>
<td>1782849</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.03</td>
<td>$0.03</td>
<td>y y y</td>
</tr>
<tr>
<td>D4</td>
<td>Multicomp BZX84C6V2 Zener SOT-23</td>
<td>1466607</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.13</td>
<td>$0.13</td>
<td>y y y</td>
</tr>
<tr>
<td>LED1,2</td>
<td>Multicomp yellow LED 0805</td>
<td>2112117</td>
<td>Element14</td>
<td>2 ea</td>
<td>ea</td>
<td>$0.07</td>
<td>$0.13</td>
<td>n y y</td>
</tr>
<tr>
<td>C1</td>
<td>Nichicon 10μF 25V Aluminium Low ESR</td>
<td>2148334</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$1.79</td>
<td>$1.79</td>
<td>y y y</td>
</tr>
<tr>
<td>C2</td>
<td>Multicomp 1μF 10V 0805</td>
<td>9406301</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.03</td>
<td>$0.03</td>
<td>n y y</td>
</tr>
<tr>
<td>C3</td>
<td>Murata 10μF 16V 0805 ceramic</td>
<td>1845747</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.24</td>
<td>$0.24</td>
<td>n y y</td>
</tr>
<tr>
<td>C4</td>
<td>Vishay 100μF 16V 2917 tantalum</td>
<td>1754095</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.28</td>
<td>$0.28</td>
<td>y y y</td>
</tr>
<tr>
<td>C5</td>
<td>Multicomp 4.7μF 6V3 0805</td>
<td>1759475</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.03</td>
<td>$0.03</td>
<td>n y y</td>
</tr>
<tr>
<td>L1</td>
<td>Bourns 15μH 9A SMD Inductor</td>
<td>9315306</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.47</td>
<td>$0.47</td>
<td>y y y</td>
</tr>
<tr>
<td>T1</td>
<td>Murata 10k 3380K thermistor 0603</td>
<td>9528105</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.33</td>
<td>$0.33</td>
<td>y y y</td>
</tr>
<tr>
<td>U1</td>
<td>Linear Technology LT3652HVIMSE. MSOP12</td>
<td>2295371</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$7.59</td>
<td>$7.59</td>
<td>y y y</td>
</tr>
<tr>
<td>R1F</td>
<td>Multicomp 511k 1% 0.1W 0805</td>
<td>2141937</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.01</td>
<td>$0.01</td>
<td>n y y</td>
</tr>
<tr>
<td>R1V</td>
<td>Murata 47k trimpot single turn SMD</td>
<td>2219392</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.21</td>
<td>$0.21</td>
<td>y y y</td>
</tr>
<tr>
<td>R2</td>
<td>Multicomp 100k 5% 0.125W 0805</td>
<td>2074341</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.01</td>
<td>$0.01</td>
<td>n y y</td>
</tr>
<tr>
<td>R3</td>
<td>Welwyn 0R1 1% 0805 [sense resistor]</td>
<td>1506142</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.38</td>
<td>$0.38</td>
<td>n y y</td>
</tr>
<tr>
<td>R4</td>
<td>Multicomp 324k 1% 0.125W 0805</td>
<td>2139067</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.01</td>
<td>$0.01</td>
<td>n y y</td>
</tr>
<tr>
<td>R5</td>
<td>Vishay 324k 1% 0.125W 0805</td>
<td>2139067</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.01</td>
<td>$0.01</td>
<td>n y y</td>
</tr>
<tr>
<td>R6</td>
<td>Yageo 1k 1% 0.125W 0805</td>
<td>9237496</td>
<td>Element14</td>
<td>1 ea</td>
<td>ea</td>
<td>$0.01</td>
<td>$0.01</td>
<td>n y y</td>
</tr>
<tr>
<td>R7,8</td>
<td>Panasonic 1k91 0.1% 0.25W 1206</td>
<td>2094979</td>
<td>Element14</td>
<td>2 ea</td>
<td>ea</td>
<td>$0.35</td>
<td>$0.70</td>
<td>n y y</td>
</tr>
<tr>
<td>CON1,2,3,4</td>
<td>Phoenix Contact PCB screw down terminal block 2 way</td>
<td>3041440</td>
<td>Element14</td>
<td>4 ea</td>
<td>ea</td>
<td>$1.70</td>
<td>$6.80</td>
<td>n y y</td>
</tr>
<tr>
<td>PCB</td>
<td>CHRG_PCB - [?manu?]</td>
<td>???</td>
<td>???</td>
<td>???</td>
<td>???</td>
<td>???</td>
<td>???</td>
<td>y y y</td>
</tr>
</tbody>
</table>

$19.98

Schematic

STE ENG4111/4112 Project
B.7 Solar Charger BOM

Total Cost: $19.98
Attached below are the data logs from the four Nodes used in testing the solar panel controller and battery charger prototype.
### B.9 Solar Charger Data: Node 1

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Sys V</th>
<th>Sys A</th>
<th>Batt V</th>
<th>Batt A</th>
<th>Panel V</th>
<th>Panel A</th>
<th>Output</th>
<th>O</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 16:00</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:10</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:20</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:30</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:40</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:50</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:00</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:10</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
</tbody>
</table>

**Manufacturer's SN:** STE_PROJ_10W_Cntrl
**Unit ID:** 1

Date/Time | Sys V | Sys A | Batt V | Batt A | Panel V | Panel A | Output | O | D |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 16:40</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:50</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:00</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:10</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
</tbody>
</table>

**User's SN:** Node1

**Unit ID:** 1

**Manufacturer's SN:** STE_PROJ_10W_Cntrl
**User's SN:** Node1

Date/Time | Sys V | Sys A | Batt V | Batt A | Panel V | Panel A | Output | O | D |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 16:40</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 16:50</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:00</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 17:10</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
</tbody>
</table>

Date/Time | Sys V | Sys A | Batt V | Batt A | Panel V | Panel A | Output | O | D |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 18:45</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 18:55</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 19:05</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 19:15</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
</tbody>
</table>

**User's SN:** Node1

**Unit ID:** 1

**Manufacturer's SN:** STE_PROJ_10W_Cntrl
**User's SN:** Node1

Date/Time | Sys V | Sys A | Batt V | Batt A | Panel V | Panel A | Output | O | D |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 19:25</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 19:35</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 19:45</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
<tr>
<td>16/10/2013 19:55</td>
<td>12.345</td>
<td>0.123</td>
<td>12.654</td>
<td>0.023</td>
<td>12.476</td>
<td>0.023</td>
<td>12.546</td>
<td>0.234</td>
<td>12.345</td>
</tr>
</tbody>
</table>
Time Series data for the period 5.00am 17/10/13 to 5.00 am 18/10/13

B.10 Solar Charger Data: Node 1: 17 & 18
Time Series data for the period 5.00am 18/10/13 to 5.00 am 19/10/13
B.12 Solar Charger Data: Node 1: 19 & 20

Time Series data for the period 5:00am 19/10/13 to 5:00 am 20/10/13
B.13 Solar Charger Data: Node 1: 20 & 21

Time Series data for the period 5.00am 20/10/13 to 5.00 am 21/10/13
<table>
<thead>
<tr>
<th>Time</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
<th>Value 6</th>
<th>Value 7</th>
<th>Value 8</th>
<th>Value 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013 22:39</td>
<td>12.807</td>
<td>0.278</td>
<td>12.467</td>
<td>0.124</td>
<td>0</td>
<td>0.111</td>
<td>0</td>
<td>0</td>
<td>0.114</td>
</tr>
<tr>
<td>16/10/2013 23:09</td>
<td>12.807</td>
<td>0.278</td>
<td>12.387</td>
<td>0.114</td>
<td>0</td>
<td>0.111</td>
<td>0</td>
<td>0</td>
<td>0.114</td>
</tr>
<tr>
<td>16/10/2013 23:10</td>
<td>12.807</td>
<td>0.278</td>
<td>12.387</td>
<td>0.114</td>
<td>0</td>
<td>0.109</td>
<td>0</td>
<td>0</td>
<td>0.114</td>
</tr>
<tr>
<td>16/10/2013 23:11</td>
<td>12.807</td>
<td>0.278</td>
<td>12.387</td>
<td>0.114</td>
<td>0</td>
<td>0.111</td>
<td>0</td>
<td>0</td>
<td>0.114</td>
</tr>
</tbody>
</table>

B.14 Solar Charger Data: Node 2

131
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
<th>Value 6</th>
<th>Value 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>17/10/2013</td>
<td>11:21</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:22</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:23</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:24</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:25</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:26</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:27</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:28</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:29</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:30</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:31</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:32</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:33</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:34</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:35</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:36</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:37</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:38</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:39</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:40</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:41</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>11:42</td>
<td>12.612</td>
<td>0.287</td>
<td>12.612</td>
<td>0.118</td>
<td>20.364</td>
<td>0.125</td>
<td>0.1</td>
</tr>
</tbody>
</table>

**B.14 Solar Charger Data: Node 2**

- **Value 1**: 135
- **Value 2**: 0.012
- **Value 3**: 0.118
- **Value 4**: 0.028
- **Value 5**: 0.028
- **Value 6**: 0.03
- **Value 7**: 0.03
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>User Id 1</th>
<th>User Id 2</th>
<th>User Id 3</th>
<th>User Id 4</th>
<th>User Id 5</th>
<th>User Id 6</th>
<th>User Id 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>17/10/2013 16:1</td>
<td>12:21:05.571</td>
<td>12:161:187</td>
<td>0</td>
<td>0.251</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 16:4</td>
<td>12:21:05.571</td>
<td>12:161:187</td>
<td>0</td>
<td>0.251</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:6</td>
<td>12:21:05.571</td>
<td>12:161:187</td>
<td>0</td>
<td>0.251</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:8</td>
<td>12:21:05.571</td>
<td>12:161:187</td>
<td>0</td>
<td>0.251</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 21:3</td>
<td>12:21:05.571</td>
<td>12:161:187</td>
<td>0</td>
<td>0.251</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B.14 Solar Charger Data: Node 2

136
B.14 Solar Charger Data: Node 2

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Node</th>
<th>Voltage</th>
<th>Current</th>
<th>Temperature</th>
<th>Power</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>21/10/2013 08:08</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:00</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 11:31</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:27</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 13:51</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:26</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 16:22</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:00</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 18:50</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:00</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 21:19</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:00</td>
<td>12:00</td>
<td>12:00</td>
</tr>
<tr>
<td>21/10/2013 23:44</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>12:00</td>
<td>11:00</td>
<td>12:00</td>
<td>12:00</td>
</tr>
</tbody>
</table>
Time Series data for the period 5.00am 17/10/13 to 5.00 am 18/10/13
Time Series data for the period 5.00am 18/10/13 to 5.00 am 19/10/13
B.17 Solar Charger Data: Node 1: 19 & 20

Time Series data for the period 5.00am 19/10/13 to 5.00 am 20/10/13
B.18 Solar Charger Data: Node 1: 20 & 21

Time Series data for the period 5.00am 20/10/13 to 5.00 am 21/10/13
<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Unit ID</th>
<th>mV</th>
<th>mV</th>
<th>mV</th>
<th>mV</th>
<th>mV</th>
<th>mV</th>
<th>mV</th>
</tr>
</thead>
<tbody>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.833</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.833</td>
<td>0.163</td>
<td>11.854</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.792</td>
<td>0.162</td>
<td>11.874</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.977</td>
<td>0.166</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.997</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.956</td>
<td>0.162</td>
<td>12.018</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.895</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.915</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16/10/2013</td>
<td>01:18</td>
<td>11.874</td>
<td>0.162</td>
<td>11.936</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Date/Time</td>
<td>Node</td>
<td>Voltage</td>
<td>Current</td>
<td>Power</td>
<td>Status</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>------</td>
<td>---------</td>
<td>---------</td>
<td>-------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:12</td>
<td>3</td>
<td>11.82</td>
<td>0.29</td>
<td>11.56</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:29</td>
<td>3</td>
<td>11.38</td>
<td>0.27</td>
<td>11.51</td>
<td>0.08</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 20:36</td>
<td>3</td>
<td>11.20</td>
<td>0.26</td>
<td>11.46</td>
<td>0.10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 21:42</td>
<td>3</td>
<td>11.00</td>
<td>0.25</td>
<td>11.30</td>
<td>0.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 22:48</td>
<td>3</td>
<td>11.40</td>
<td>0.28</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 23:02</td>
<td>3</td>
<td>11.38</td>
<td>0.28</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:14</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:16</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:18</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:20</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:22</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:24</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:26</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:28</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:30</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:32</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:34</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:36</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:38</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:40</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:42</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:44</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:46</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:48</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:50</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:52</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:54</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:56</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 18:58</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:00</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:02</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:04</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:06</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:08</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:10</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:12</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:14</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:16</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:18</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:20</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:22</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:24</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:26</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:28</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17/10/2013 19:30</td>
<td>3</td>
<td>11.60</td>
<td>0.29</td>
<td>11.71</td>
<td>0.16</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
B.20 Solar Charger Data: Node 1: 17 & 18

Time Series data for the period 5.00am 17/10/13 to 5.00 am 18/10/13
B.21 Solar Charger Data: Node 1: 18 & 19

Time Series data for the period 5.00am 18/10/13 to 5.00 am 19/10/13
Time Series data for the period 5.00am 19/10/13 to 5.00 am 20/10/13
B.23 Solar Charger Data: Node 1: 20 & 21

Time Series data for the period 5.00am 20/10/13 to 5.00 am 21/10/13

![Chart showing voltage and current data over time, with specific times and values indicated.]
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>17/10/2013</td>
<td>0:15</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:16</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:17</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:18</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:19</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:20</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:21</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:22</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:23</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:24</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:25</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:26</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:27</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:28</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:29</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:30</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:31</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:32</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:33</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:34</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:35</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:36</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:37</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:38</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:39</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:40</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:41</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:42</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:43</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:44</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:45</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:46</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:47</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:48</td>
<td>12.12</td>
</tr>
<tr>
<td>17/10/2013</td>
<td>0:49</td>
<td>12.12</td>
</tr>
</tbody>
</table>

B.24 Solar Charger Data: Node 4

169
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Voltage</th>
<th>Current</th>
<th>Temperature</th>
<th>Power</th>
<th>Charge Status</th>
<th>Battery Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>21/10/2013</td>
<td>6:27</td>
<td>11.156</td>
<td>0.246</td>
<td>11.125</td>
<td>0.138</td>
<td>15.637</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:29</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.816</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:31</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.832</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:33</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.914</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:35</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.258</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:37</td>
<td>11.136</td>
<td>0.248</td>
<td>11.125</td>
<td>0.118</td>
<td>15.135</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:41</td>
<td>11.136</td>
<td>0.246</td>
<td>11.125</td>
<td>0.118</td>
<td>15.832</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:43</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.094</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:45</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.832</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:47</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>15.832</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:49</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:51</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:53</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:55</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>6:57</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:01</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:03</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:05</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:07</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:09</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:11</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:13</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:15</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:17</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:19</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:21</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:23</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
<tr>
<td>21/10/2013</td>
<td>7:25</td>
<td>11.136</td>
<td>0.244</td>
<td>11.125</td>
<td>0.118</td>
<td>16.365</td>
<td>0.1</td>
</tr>
</tbody>
</table>

---

B.24 Solar Charger Data: Node 4

185
Time Series data for the period 5.00am 17/10/13 to 5.00 am 18/10/13

- Sys V
- Batt V
- Panel V
- Sys A
- Batt A
- Panel A

B.25 Solar Charger Data: Node 1: 17 & 18
Time Series data for the period 5.00am 18/10/13 to 5.00 am 19/10/13
Time Series data for the period 5.00am 19/10/13 to 5.00 am 20/10/13
B.28 Solar Charger Data: Node 1: 20 & 21

Time Series data for the period 5.00am 20/10/13 to 5.00 am 21/10/13
Appendix C

Data Sheets
MAX5033
500mA, 76V, High-Efficiency, MAXPower Step-Down DC-DC Converter

**General Description**

The MAX5033 step-down, high-efficiency, high-voltage, step-down DC-DC converter (DC-DC) operates from 5.5V to 76V and can source up to 500mA current. This monolithic, fixed-frequency, current-mode DC-DC converter incorporates a high-voltage bootstrap circuit to drive the high-side MOSFET, a high-voltage feedback comparator, and a low-voltage controller. The MAX5033 is optimized for automotive and rugged industrial applications where high current output is required. Open-loop efficiency at high input voltages, high efficiency at light loads. The MAX5033 includes thermal shutdown and short-circuit current limit to prevent permanent damage to the device. The MAX5033 delivers up to 500mA output current. The output current may be limited by the maximum output voltage of 24V. This device features on-package output short-circuit protection, and thermal shutdown.

The MAX5033 is available in space-saving, small SO and leadless SIP packages and operates over the automotive -40°C to +125°C temperature range. The MAX5033 is available in both leaded (Pb) and lead (Pb)-free packages: To order the lead (Pb)-free package, add a + after the part number.

**Applications**

Automotive, Consumer Electronics, Industrial, Distributed Power

**Typical Operating Circuit**

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim’s website at www.maximintegrated.com.

---

**Features**

- 500mA output current
- Adjustable output voltage from 3.3V, 5V, and 12V
- Low dropout voltage of 7.5V
- Fixed 125kHz switching frequency
- High efficiency at light loads
- Fixed 125kHz switching frequency
- Thermal shutdown and short-circuit current limit

**ABSOLUTE MAXIMUM RATINGS**

- Input Voltage: 5.5V to 76V
- Output Voltage: 0V to VIN
- Operating Junction Temperature: -40°C to +125°C
- Storage Temperature: -65°C to +150°C

**ELECTRICAL CHARACTERISTICS**

- **Maxim Integrated**
  - 
  - 

---

**Ordering Information**

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim’s website at www.maximintegrated.com.
Step-Down DC-DC Converter
500mA, 76V, High-Efficiency, MAXPower

Typical Operating Characteristics

VIN = 12V, V ON/OFF CURRENT (µA)

(TEMPERATURE (°C))

MAX5033

Pin Description

Pin | Name | Function
--- | --- | ---
1 | V IN | Input Voltage. Bypass VIN to GND with a low-ESR capacitor as close to the device as possible.
2 | GND | Ground
3 | 5 | ON/OFF = VIN
4 | 6 | BST Boost Capacitor Connection. Connect a 0.1µF ceramic capacitor from BST to LX.
5 | Pin 7 to 8 | PEAK SWITCH CURRENT LIMIT (A)
6 | Pin 8 | Source Connection of Internal High-Side Switch

Simplified Block Diagram

MAX5033

MAX5033

Maxim Integrated
MAX5033 500mA, 76V, High-Efficiency, MAXPower Step-Down DC-DC Converter

The MAX5033 features internal compensation for optimum on-chip compensation bandwidth and phase margin. With the precise compensation, the device easily meets transient response requirements for low ESR output capacitors. Use a 10µF tantalum capacitor and a 150µF electrolytic capacitor in parallel to achieve a fast transient response and low output noise. Set the output capacitance according to the application tolerances and output ripple requirements.

**Thermal-Overload Protection**
The MAX5033 features thermal overload protection. If the junction temperature rises above 160°C, the device will shut down and will turn off the output to protect against destruction. The maximum junction temperature is 150°C.

**Overvoltage Protection**
The MAX5033 has an internal soft-start time (tSS) of 400µs. It is important to reset the output voltage at startup before the soft-start timer times out. The soft-start timer is reset by the output current exceeding the soft-start current limit. The soft-start timer times out after 400µs. It is important to keep the output rise time at tSS to avoid output overshoot. The output rise time is directly proportional to the output capacitor.

**Undervoltage Lockout (UVLO)**
The undervoltage lockout (UVLO) is an internal circuit that turns off the device when the output voltage drops below the UVLO threshold. The UVLO threshold is 6.5V, 7.5V, and 12V, respectively. The MAX5033 also features an adjustable UVLO threshold. The output capacitor and its ESR form a zero, which must be equal or greater than the maximum capacitance value times the desired zero frequency. The output capacitor is determined by the application requirements and output ripple requirements.

**VDD Pin Protection**
The VDD pin has 100mV hysteresis. If the VDD pin voltage decreases below the UVLO threshold, the device will turn off. The VDD pin voltage increases above the UVLO threshold, the device will turn on. The UVLO threshold is 5.0V, 7.5V, and 12V, respectively. Connect FB to the preset voltage to turn on the device.

**Applications Information**

**Selecting a Rectifier**
Choose a rectifier with a continuous current rating equal to the maximum required current. Choose a rectifier with a maximum voltage rating greater than the maximum input voltage. The rectifier should have 10µF tantalum capacitors in parallel to achieve a fast transient response and low output noise. Set the output capacitance according to the application tolerances and output ripple requirements.

**Selecting an Output Capacitor**
The output capacitance and its ESR form a zero, which must be equal or greater than the maximum capacitance value times the desired zero frequency. The output capacitor is determined by the application requirements and output ripple requirements.

**Thermal Considerations**
The thermal resistance of the device is 120°C/W. This means that for every 1°C increase in the junction temperature, the device will heat up by 120°C. Use the following equations to calculate the input currents and output current requirements:

\[
\text{I}_{\text{IN}} = \frac{\text{V}_{\text{IN}} - \text{V}_{\text{OUT}} - \text{V}_{\text{DS}}}{\text{R}_{\text{DS} \text{(ON)}}}
\]

where \(\text{V}_{\text{IN}}\) is the maximum input voltage and \(\text{V}_{\text{OUT}}\) is the minimum output voltage from VIN to GND with the center node to GND.

**Inductor Selection**
Choose an inductor with a peak inductor current greater than the maximum load current and a peak inductor current equal to the maximum load current. Use the following equations to calculate the peak inductor current:

\[
\text{I}_{\text{PRMS}} = \frac{\text{V}_{\text{IN}} \times \text{I}_{\text{OUT}}}{\text{R}_{\text{L}}}
\]

where \(\text{V}_{\text{IN}}\) is the maximum input voltage, \(\text{I}_{\text{OUT}}\) is the minimum output current, and \(\text{R}_{\text{L}}\) is the load resistance.

**PCB Layout Considerations**
Proper PCB layout is critical for optimal performance. Ensure a ground plane is used on the PCB. Use the following equations to calculate the ground plane area:

\[
\text{Ground Plane Area} = \frac{\text{V}_{\text{IN}} \times \text{I}_{\text{OUT}}}{\text{R}_{\text{GND}}}
\]

where \(\text{R}_{\text{GND}}\) is the ground plane resistance.

**Conclusion**
The MAX5033 is a high-efficiency, low-voltage, step-down DC-DC converter suitable for a wide range of applications. It features internal compensation, thermal overload protection, undervoltage lockout, and output short-circuit protection. The device also features an adjustable UVLO threshold. Use a 10µF tantalum capacitor and a 150µF electrolytic capacitor in parallel to achieve a fast transient response and low output noise. Set the output capacitance according to the application tolerances and output ripple requirements.
Table 2. Typical External Components Selection (Circuit of Figure 2) (continued)

<table>
<thead>
<tr>
<th>VIN (V)</th>
<th>VOUT (V)</th>
<th>IOUT (A)</th>
<th>EXTERNAL COMPONENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3</td>
<td>0.5</td>
<td></td>
<td>CIN = 100µF, Panasonic, EEVFK1H101P</td>
</tr>
<tr>
<td>5V</td>
<td>0.5</td>
<td></td>
<td>CBST = 0.1µF, 0805</td>
</tr>
<tr>
<td>12V</td>
<td>0.5</td>
<td></td>
<td>CIN = 100µF, Panasonic, EEVFK1H101P</td>
</tr>
</tbody>
</table>

Table 3. Component Suppliers

<table>
<thead>
<tr>
<th>SUPPLIER</th>
<th>PHONE</th>
<th>FAX</th>
<th>WEBSITE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vishay</td>
<td>402-563-6866</td>
<td>402-563-6296</td>
<td><a href="http://www.vishay.com">www.vishay.com</a></td>
</tr>
<tr>
<td>TDK</td>
<td>847-803-6100</td>
<td>847-390-4405</td>
<td><a href="http://www.component.tdk.com">www.component.tdk.com</a></td>
</tr>
<tr>
<td>Panasonic</td>
<td>714-373-7366</td>
<td>714-737-7323</td>
<td><a href="http://www.panasonic.com">www.panasonic.com</a></td>
</tr>
<tr>
<td>Nichicon</td>
<td>858-824-1515</td>
<td>858-824-1525</td>
<td><a href="http://www.nichicon.com">www.nichicon.com</a></td>
</tr>
<tr>
<td>Diodes Incorporated</td>
<td>805-446-4800</td>
<td>805-446-4850</td>
<td><a href="http://www.diodes.com">www.diodes.com</a></td>
</tr>
<tr>
<td>Coilcraft</td>
<td>847-639-6400</td>
<td>847-639-1469</td>
<td><a href="http://www.coilcraft.com">www.coilcraft.com</a></td>
</tr>
<tr>
<td>AVX</td>
<td>843-946-0238</td>
<td>843-626-3123</td>
<td><a href="http://www.avxcorp.com">www.avxcorp.com</a></td>
</tr>
</tbody>
</table>

Figure 3. Load Temperature Monitoring with ON/OFF (Requires Accurate VIN)

Figure 4. Dual-Sequenced DC-DC Converters (Startup Delay Determined by R1/R1', Ct/Ct' and Rt/Rt')

Chip Information

PROCESS: BiCMOS

MAX5033

500mA, 76V, High-Efficiency, MAXPower Step-Down DC-DC Converter

MAX5033

500mA, 76V, High-Efficiency, MAXPower Step-Down DC-DC Converter
### MAX5033

**500mA, 76V, High-Efficiency, MAXPower Step-Down DC-DC Converter**

**Revision History**

<table>
<thead>
<tr>
<th>REVISION NUMBER</th>
<th>REVISION DATE</th>
<th>DESCRIPTION</th>
<th>PAGE CHANGED</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>9/03</td>
<td>Initial release</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>5/04</td>
<td>New product update</td>
<td>1–7, 10</td>
</tr>
<tr>
<td>2</td>
<td>6/04</td>
<td>Removed future product asterisk and made specification changes</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>3</td>
<td>1/07</td>
<td>Modified Absolute Maximum Ratings specifications</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4/10</td>
<td>Corrected inconsistencies in Absolute Maximum Ratings and Electrical Characteristics tables</td>
<td>1, 2, 3, 4, 17</td>
</tr>
</tbody>
</table>

Maxim Integrated  160 Rio Robles, San Jose, CA 95134 USA  1-408-601-1000
Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied.
Maxim reserves the right to change the circuitry and specifications without notice at any time. The parametric values (min and max limits) shown in the Electrical Characteristics table are guaranteed. Other parametric values quoted in this data sheet are provided for guidance.

© 2010 Maxim Integrated The Maxim logo and Maxim Integrated are trademarks of Maxim Integrated Products, Inc.
Features:
- Absorbent Glass Mat (AGM) technology for superior performance.
- Valve-regulated, spill-proof construction allows safe operation in any position.
- Power/weight ratio yielding unrivaled energy density.
- Rugged impact resistant ABS case and cover (UL94-HB).
- U.L. recognized under file number E1231665.

Performance Specifications:
- **Nominal Voltage**: 12 volts (10.5 volts).
- **Rated Capacity**
  - 30 hrs: (88.8Ah to 10.50 volts) 7.00 Ah
  - 30 hrs: (65.0Ah to 10.50 volts) 6.50 Ah
  - 30 hrs: (35.0Ah to 10.50 volts) 3.50 Ah
- **Approximate Weight**: 4.25 lbs (1.93 kg).
- **Energy Density**: 2.50 W-hr/lb (5.55 W-hr/kg).
- **Internal Resistance**: (approx.) 0.100 ohms.
- **Max Discharge Current**: 7.00 amperes.
- **Max Short-Duration Discharge Current**: (10 Sec.) 9.00 amperes.
- **Cycle Applications**:
  - **Float** or **Stand-By** Service:
    - Recharge at a rate of 0.25C up to full capacity.
    - Recharge at a rate of 0.1C until 80% charged.
    - Once fully charged, allow the battery to float until next charge. Failure to float the battery will result in lower performance.
  - **Deep** or **Full** Discharge:
    - Recharge at a rate of 0.25C up to 80% charged.
    - Recharge at a rate of 0.1C until 50% charged.
    - If cell voltage drops below 1.75 volts, recharge at a rate of 0.1C for 10 hours.

Technical Data:
- **Physical Dimensions**:
  - **Length**: 5.95 in (151 mm)
  - **Width**: 3.70 in (94 mm)
  - **Height**: 3.25 in (83 mm)
  - **Case**: ABS Plastic

Further Information:
For more information about our products, please visit www.power-sonic.com.
LT3652

Power Tracking 2A Battery Charger for Solar Power

**FEATURES**

- Input Supply Voltage Regulation Long for Peak Power Tracking (MPPT) Solar Applications
- Wide Input Voltage Range: 2.5V to 50V (Unipolar)
- Programmable Charge Rate Up to 2A
- Optional Termination Timer: T25 for On-Board Termination
- Rectangular Programmable Final Voltage Up to 14.4V
- Accommodates Li-Ion Polymer, LiFePO4, SLA CHEM Battery
- No VIN Blocking Diode Required for Battery Voltages > 4.2V
- 1MHz Fixed Frequency
- 6% Fixed Voltage Reference Accuracy
- ±2% Cristiano Current Accuracy
- ±2% C/10 Discrete Accuracy
- Binary-Coded Open-Collector I/O (IO)
- Three-State (ON) to MSOP-12 Packages

**APPLICATIONS**

- Solar Powered Applications
- Renewable Monitoring Stations
- LiFePO4 (Lithium Phosphate) Applications
- Portable Handheld Instruments
- 12V to 24V Automotive Systems

**ABSOLUTE MAXIMUM RATINGS**

- Temperature Range: –40°C to 125°C
- VIN = 20V, Boost – SW = 4V
- Note 1: The temperature grade is identified by a label on the shipping container.

**DESCRIPTION**

The LT®3652 is a complete monolithic step-down battery charger that operates over a 4.05V to 31V input voltage range. The LT3652 provides a constant-current/constant-voltage charge characteristic, with maximum charge current externally programmable up to 2A. The charger employs a 3.3V float voltage feedback reference, so any desired battery float voltage up to 14.4V can be programmed with a resistor divider.

The LT3652 includes internal voltage regulators which reduce charge current if the input voltage falls below a programmed level, set with a resistor divider. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak input power.

The LT3652 can be configured to terminate charging when charge current falls below 1% of the programmed maximum (C/10). Once charging is terminated, the LT3652 enters a non-current (IDLE) standby mode. An audio-charge feature starts a new charging cycle of the battery voltage at 2% below the programmed float voltage. The LT3652 also contains a programmable safety timer, which can be used to terminate charging after a desired time is reached.

The LT3652 also contains a programmable charge rate up to 2A. For solar panel applications, the input regulation loop tracks the maximum power point voltage, optimizing charge efficiency.

**APPLICATIONS**

- The LT3652 can be configured to terminate charging when charge current falls below 1% of the programmed maximum (C/10). Once charging is terminated, the LT3652 enters a non-current (IDLE) standby mode. An audio-charge feature starts a new charging cycle of the battery voltage at 2% below the programmed float voltage.

**ORDER INFORMATION**

- **PACKAGE**
- **ORDERING INFORMATION**

**ELECTRICAL CHARACTERISTICS**

- **Description:**
  - The table contains the specifications which apply over the full operating temperature range and alternate specifications at ±2°C, ±2V, AVS (0.01%), –10°C, 0°C, 50°C, 150°C, C/2, C/10, 25°C, 40°C.

**TYPICAL APPLICATION**

- 2A Solar Panel Power Manager with 7.2V LiFePO4 Battery and 12V Peak Power Tracking

**PIN CONFIGURATION**

- **TOP VIEW**
- **BOTTOM VIEW**

**ELECTRICAL CHARACTERISTICS**

- **Description:**
  - The table contains the specifications which apply over the full operating temperature range and alternate specifications at ±2°C, ±2V, AVS (0.01%), –10°C, 0°C, 50°C, 150°C, C/2, C/10, 25°C, 40°C.
PIN FUNCTIONS

VIN_REG (Pin 2): Input Voltage Regulation Reference. Maximum charge current can be reduced from this pin by pulling it below 2.7V. Connecting a resistor divider from VIN to VIN_REG enables programmed minimum operational VIN voltage. This is typically used to program the peak input voltage for a solar panel. The LT3652 versus the minimum charge current requirement to maintain the operational VIN voltage. Through maintaining the voltage on VIN and VIN_REG at or above 2.7V, if this voltage regulation feature is not desired, connect the pin to VIN.

SHDN (Pin 3): Precision Threshold Shutdown Pin. The enable threshold is 1.2V (mV), with 100Ω of input resistance. When shutdown mode is engaged, all charging functions are disabled. The precision threshold allows use of the internal SHDN pin to operate at 50Ω, (Pin 3). In shutdown, the SHDN pin is pulled below 1.2V. The internal shutdown resistor is 250k, so if this function is not used, connect the pin to VIN.

CHRG (Pin 6): Open-Collector Charger Status Output, typically pulled up through a resistor to a reference voltage. This status pin can be pulled up to a voltage as high as VIN when disabled, and can sink currents up to 10mA when enabled. During battery charging cycle, if programmed threshold voltage remains above 2.7V, the status pin remains high-impedance.

FAULT (Pin 8): Battery Temperature Monitor Pin. When a temperature fault occurs, this pin is pulled low. If no fault conditions exist, the FAULT pin remains high-impedance.

SW (Pin 12): Switch Forward Drop (V_IN – VSW) vs Temperature

VFB (Pin 7): Battery Float Voltage Feedback Reference. The VFB pin is programmed to achieve a final float voltage of 3.7V on this pin. Output battery float voltage (VFBREF) is programmed using a resistor divider (as in the following equation: VFBREF = VFB(TH) x R1 / (R1 + R2)), and is also the reference for the current sense voltage. Once a charge cycle is terminated, the input bias current of the BAT pin is reduced to 0.8μA, to minimize battery discharge while the charger remains connected.

SENSE (Pin 10): Battery Bias Current, Efficiency, and Power Dissipation Data (VSENSE – VFB) vs Temperature

C/10 Threshold (VSENSE – VFB) vs Temperature

VSENSE (DC) (mV)

CHRG (Pin 6) is connected to BAT to VIN, and R2 is connected from VIN to ground.

Rated resistor values to program desired (VSENSE) below the equation:

R1 = (VFB(TH) x 2.5 x VFB/3) Ω
R2 = (0.2 x 2.5 x VFB/3) Ω

A 0.1μF capacitor is connected between VIN and VFB to filter noise or application requirements. Voltage sense of charging state is being regulated at 0.8μA, and the VFB pin is pulled low. If no fault conditions exist, the internal timer is used to disable this function. If the internal timer is used to disable this function, the timer is pulsed, and the charging cycle is terminated when the voltage on VIN_REG falls below the 2.7V threshold, at or above 2.7V, if this function is not used, connect the pin to VIN.

CHRG (Pin 6): Open-Collector Charger Status Output, typically pulled up through a resistor to a reference voltage. This status pin can be pulled up to a voltage as high as VIN when disabled, and can sink currents up to 10mA when enabled. During battery charging cycle, if programmed threshold voltage remains above 2.7V, the status pin remains high-impedance.

SW (Pin 12): Switch Forward Drop (V_IN – VSW) vs Temperature

VFB (Pin 7): Battery Float Voltage Feedback Reference. The VFB pin is programmed to achieve a final float voltage of 3.7V on this pin. Output battery float voltage (VFBREF) is programmed using a resistor divider (as in the following equation: VFBREF = VFB(TH) x R1 / (R1 + R2)), and is also the reference for the current sense voltage. Once a charge cycle is terminated, the input bias current of the BAT pin is reduced to 0.8μA, to minimize battery discharge while the charger remains connected.

SENSE (Pin 10): Battery Bias Current, Efficiency, and Power Dissipation Data (VSENSE – VFB) vs Temperature

C/10 Threshold (VSENSE – VFB) vs Temperature

VSENSE (DC) (mV)

CHRG (Pin 6) is connected to BAT to VIN, and R2 is connected from VIN to ground.

Rated resistor values to program desired (VSENSE) below the equation:

R1 = (VFB(TH) x 2.5 x VFB/3) Ω
R2 = (0.2 x 2.5 x VFB/3) Ω

A 0.1μF capacitor is connected between VIN and VFB to filter noise or application requirements. Voltage sense of charging state is being regulated at 0.8μA, and the VFB pin is pulled low. If no fault conditions exist, the internal timer is used to disable this function. If the internal timer is used to disable this function, the timer is pulsed, and the charging cycle is terminated when the voltage on VIN_REG falls below the 2.7V threshold, at or above 2.7V, if this function is not used, connect the pin to VIN.
APPLICATIONS INFORMATION

The LT3652 is a complete monolithic, mid-power multichemistry battery charger designed for input voltage applications with solutions that require minimum external components. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch. The LT3652 incorporates a 2A switch that is driven by a timer-based control to minimize dynamic performance during charging cycles. Wide input range allows operation to full float voltage as long as the IC remains powered and the sensed battery voltage becomes low enough. The LT3652 employs an input voltage regulation loop, which reduces charge current if the monitored input voltage falls below a programmed level. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak output power.

The LT3652 automatically enters a battery protection mode when the sensed battery voltage goes below a programmed level, and the current delivered while the charger is operating is constant-current (CC) mode. When the voltage on the V FB pin is below 2.3V (V FB(PRE)), the LT3652 switches to constant-voltage (CV) mode, which corresponds to 100mV effective voltage range of ITH. A clamp limits this voltage to a maximum of 3.7V. As this occurs, the ITH voltage falls from the limit clamp voltage to a value that corresponds to 100mV. The LT3652 also supports a low-current standby mode, in which the input supply voltage is reduced to 0V, and the LT3652 is biased off.

The LT3652 operates on a wide input voltage range of 0V to 8.5V, as referenced to the BOOST pin. The BOOST pin is used for battery float voltages below 4.2V and is derived from the charger input supply through a resistor divider. The LT3652 is biased directly from the charger input supply (VIN) and does not include an external biasing pin. The LT3652 contains a battery temperature monitoring circuit. This feature monitors battery temperature using a thermistor during the charging cycle. If the battery temperature becomes outside a safe charging range of 0°C to 40°C, the IC suspends charging and initiates a fault condition until the temperature returns to the safe charging range.

The LT3652 contains two digital open-collector outputs, which provide charger status and signal fault conditions. These binary-codes pins signal for battery charging, standby shutdown, and battery faults. The LT3652 is a complete monolithic, mid-power, multi-chemistry battery charger designed for input voltage applications with solutions that require minimum external components. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch.

The LT3652 is biased directly from the charger input supply through the VIN pin. This supply provides large switched currents, so high-quality, low ESR electrolytic capacitors are recommended to minimize voltage glitches on the VIN. The IC is connected to the BOOST pin through a resistor divider. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch. The LT3652 incorporates a 2A switch that is driven by a timer-based control to minimize dynamic performance during charging cycles. Wide input range allows operation to full float voltage as long as the IC remains powered and the sensed battery voltage becomes low enough. The LT3652 employs an input voltage regulation loop, which reduces charge current if the monitored input voltage falls below a programmed level. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak output power.

The LT3652 automatically enters a battery protection mode when the sensed battery voltage goes below a programmed level, and the current delivered while the charger is operating is constant-current (CC) mode. When the voltage on the V FB pin is below 2.3V (V FB(PRE)), the LT3652 switches to constant-voltage (CV) mode, which corresponds to 100mV effective voltage range of ITH. A clamp limits this voltage to a maximum of 3.7V. As this occurs, the ITH voltage falls from the limit clamp voltage to a value that corresponds to 100mV. The LT3652 also supports a low-current standby mode, in which the input supply voltage is reduced to 0V, and the LT3652 is biased off.

The LT3652 operates on a wide input voltage range of 0V to 8.5V, as referenced to the BOOST pin. The BOOST pin is used for battery float voltages below 4.2V and is derived from the charger input supply through a resistor divider. The LT3652 is biased directly from the charger input supply (VIN) and does not include an external biasing pin. The LT3652 contains a battery temperature monitoring circuit. This feature monitors battery temperature using a thermistor during the charging cycle. If the battery temperature becomes outside a safe charging range of 0°C to 40°C, the IC suspends charging and initiates a fault condition until the temperature returns to the safe charging range.

The LT3652 contains two digital open-collector outputs, which provide charger status and signal fault conditions. These binary-coded pins signal for battery charging, standby shutdown, and battery faults.

The LT3652 is biased directly from the charger input supply through the VIN pin. This supply provides large switched currents, so high-quality, low ESR electrolytic capacitors are recommended to minimize voltage glitches on the VIN. The IC is connected to the BOOST pin through a resistor divider. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch. The LT3652 incorporates a 2A switch that is driven by a timer-based control to minimize dynamic performance during charging cycles. Wide input range allows operation to full float voltage as long as the IC remains powered and the sensed battery voltage becomes low enough. The LT3652 employs an input voltage regulation loop, which reduces charge current if the monitored input voltage falls below a programmed level. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak output power.

The LT3652 automatically enters a battery protection mode when the sensed battery voltage goes below a programmed level, and the current delivered while the charger is operating is constant-current (CC) mode. When the voltage on the V FB pin is below 2.3V (V FB(PRE)), the LT3652 switches to constant-voltage (CV) mode, which corresponds to 100mV effective voltage range of ITH. A clamp limits this voltage to a maximum of 3.7V. As this occurs, the ITH voltage falls from the limit clamp voltage to a value that corresponds to 100mV. The LT3652 also supports a low-current standby mode, in which the input supply voltage is reduced to 0V, and the LT3652 is biased off.

The LT3652 operates on a wide input voltage range of 0V to 8.5V, as referenced to the BOOST pin. The BOOST pin is used for battery float voltages below 4.2V and is derived from the charger input supply through a resistor divider. The LT3652 is biased directly from the charger input supply (VIN) and does not include an external biasing pin. The LT3652 contains a battery temperature monitoring circuit. This feature monitors battery temperature using a thermistor during the charging cycle. If the battery temperature becomes outside a safe charging range of 0°C to 40°C, the IC suspends charging and initiates a fault condition until the temperature returns to the safe charging range.

The LT3652 contains two digital open-collector outputs, which provide charger status and signal fault conditions. These binary-coded pins signal for battery charging, standby shutdown, and battery faults.

The LT3652 is biased directly from the charger input supply through the VIN pin. This supply provides large switched currents, so high-quality, low ESR electrolytic capacitors are recommended to minimize voltage glitches on the VIN. The IC is connected to the BOOST pin through a resistor divider. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch. The LT3652 incorporates a 2A switch that is driven by a timer-based control to minimize dynamic performance during charging cycles. Wide input range allows operation to full float voltage as long as the IC remains powered and the sensed battery voltage becomes low enough. The LT3652 employs an input voltage regulation loop, which reduces charge current if the monitored input voltage falls below a programmed level. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak output power.

The LT3652 automatically enters a battery protection mode when the sensed battery voltage goes below a programmed level, and the current delivered while the charger is operating is constant-current (CC) mode. When the voltage on the V FB pin is below 2.3V (V FB(PRE)), the LT3652 switches to constant-voltage (CV) mode, which corresponds to 100mV effective voltage range of ITH. A clamp limits this voltage to a maximum of 3.7V. As this occurs, the ITH voltage falls from the limit clamp voltage to a value that corresponds to 100mV. The LT3652 also supports a low-current standby mode, in which the input supply voltage is reduced to 0V, and the LT3652 is biased off.

The LT3652 operates on a wide input voltage range of 0V to 8.5V, as referenced to the BOOST pin. The BOOST pin is used for battery float voltages below 4.2V and is derived from the charger input supply through a resistor divider. The LT3652 is biased directly from the charger input supply (VIN) and does not include an external biasing pin. The LT3652 contains a battery temperature monitoring circuit. This feature monitors battery temperature using a thermistor during the charging cycle. If the battery temperature becomes outside a safe charging range of 0°C to 40°C, the IC suspends charging and initiates a fault condition until the temperature returns to the safe charging range.

The LT3652 contains two digital open-collector outputs, which provide charger status and signal fault conditions. These binary-coded pins signal for battery charging, standby shutdown, and battery faults.

The LT3652 is biased directly from the charger input supply through the VIN pin. This supply provides large switched currents, so high-quality, low ESR electrolytic capacitors are recommended to minimize voltage glitches on the VIN. The IC is connected to the BOOST pin through a resistor divider. The LT3652 is a timer-based, current-sensing, average-current mode step-down switch. The LT3652 incorporates a 2A switch that is driven by a timer-based control to minimize dynamic performance during charging cycles. Wide input range allows operation to full float voltage as long as the IC remains powered and the sensed battery voltage becomes low enough. The LT3652 employs an input voltage regulation loop, which reduces charge current if the monitored input voltage falls below a programmed level. When the LT3652 is powered by a solar panel, the input regulation loop is used to maintain the panel at peak output power.

The LT3652 automatically enters a battery protection mode when the sensed battery voltage goes below a programmed level, and the current delivered while the charger is operating is constant-current (CC) mode. When the voltage on the V FB pin is below 2.3V (V FB(PRE)), the LT3652 switches to constant-voltage (CV) mode, which corresponds to 100mV effective voltage range of ITH. A clamp limits this voltage to a maximum of 3.7V. As this occurs, the ITH voltage falls from the limit clamp voltage to a value that corresponds to 100mV. The LT3652 also supports a low-current standby mode, in which the input supply voltage is reduced to 0V, and the LT3652 is biased off.

The LT3652 operates on a wide input voltage range of 0V to 8.5V, as referenced to the BOOST pin. The BOOST pin is used for battery float voltages below 4.2V and is derived from the charger input supply through a resistor divider. The LT3652 is biased directly from the charger input supply (VIN) and does not include an external biasing pin. The LT3652 contains a battery temperature monitoring circuit. This feature monitors battery temperature using a thermistor during the charging cycle. If the battery temperature becomes outside a safe charging range of 0°C to 40°C, the IC suspends charging and initiates a fault condition until the temperature returns to the safe charging range.

The LT3652 contains two digital open-collector outputs, which provide charger status and signal fault conditions. These binary-coded pins signal for battery charging, standby shutdown, and battery faults.
C.3 LT3652

APPLICATIONS INFORMATION

Using a resistor divider with an equivalent input resistance at the VFB pin of 25k for input bias current error is recommended. For 25k, RFB1 = 3.3k and RFB2 = 10μA yields:

\[ \text{RFB1} = 0.25 \times \frac{\text{RFB2}}{3.3} \]

The charge function operates to achieve the final float voltage of 3.3V on the VFB pin. The auto-reset feature includes a new charging cycle when the voltage at the VFB pin falls below 2.5V before this float voltage.

Because the battery voltage is across the VIN_REG pin, a voltage reference can be added in parallel with the 10μF ceramic bypass capacitor. This results in a 3-stage regulator, as shown in Figure 5, to ease component selection and increase output voltage precision. The output of additional current through the feedback divider.

Larger additional bypass capacitances may be desired for visual indication in a non-battery condition (see the Block Pin section).

It is desired to operate a system lead from the LT3652 charger output when the battery is disconnected. Additional bypass capacitances are required in this type of application, where the application has a paralleled battery charge or a higher additional output bulk capacitance. For these applications, 10μF/±5% ceramic capacitors are typically used. These capacitors have low ESR, which can couple external signals onto the pin, which can provide unnecessary output transients or noise. Effects of parasitic capacitances can be typically reduced by adding a small value (10kΩ to 10μF) in parallel with the 10μF ceramic bypass capacitor. This additional bypass capacitance may also be required in systems where the battery is connected to the charger with a long wiring length. The voltage rating of CFB must exceed the battery float voltage.

Inductor Selection

The primary criteria for inductor value selection is an LT3652 charger the required current of the inductor. Because the inductor current is determined, the inductor must also have a saturation current equal to or exceeding the maximum peak current in the inductor. An inductor value is selected given the desired amount of ripple current (μA) can be approximated using the relation:

\[ L = \frac{10 \times \text{I}_{\text{FB}} \times \text{V}_{\text{BAT}} \times (1 - \frac{V_{\text{BAT}}}{V_{\text{IN}}(\text{MAX})})}{(1 - \frac{V_{\text{BAT}}}{V_{\text{IN}}(\text{MAX})})^2} \]

The minimum average diode current rating (\( I_{\text{DIODE(MAX)}} \)) for the peak power point is similar to that of \( V_{\text{OC}} \). For \( V_{\text{BAT}}(\text{FL T}) = 3.6V \):

\[ I_{\text{DIODE(MAX)}} > 2 \times (25 - 0.7(7.2)) / 25 \]

The minimum average diode current rating (\( I_{\text{DIODE(MAX)}} \)) can couple external signals onto the pin, which can provide unnecessary output transients or noise. Effects of parasitic capacitances can be typically reduced by adding a small value (10kΩ to 10μF) in parallel with the 10μF ceramic bypass capacitor. This additional bypass capacitance may also be required in systems where the battery is connected to the charger with a long wiring length. The voltage rating of CFB must exceed the battery float voltage.

Battery Float Voltage Programming

The output voltage that \( V_{\text{BAT}}(\text{FL T}) \) is programmed to ensure that the battery output is programmed by connecting a resistor divider from the BAT pin to VFB. \( V_{\text{BAT}}(\text{FL T}) \) can be programmed up to 14.4V.

Reference Selection

The reference selection for the LT3652 charger is the ripple current created in that inductor. Ripple current is typically set within a range of 25% to 29% to allow for a reliable output. Ripple current is typically set within a range of 25% to 29% to allow for a reliable output. Ripple current is typically set within a range of 25% to 29% to allow for a reliable output. Ripple current is typically set within a range of 25% to 29% to allow for a reliable output.
As the temperature coefficient for $V_{oc}$ is similar to that of $V_{mp}$, for specified temperature coefficient for $V_{oc}$ ($TC_{oc}$ = –19.8 mV/°C) and the specified open circuit voltage ($V_{oc}$) at 77°F is not to be inserted into the equations to calculate the appropriate resistor values for the temperature compensation network in Figure 8. With $R_{chrg}$ equal to 10kΩ, then:

$$R_{chrg} = \frac{\Delta V_{oc}}{\Delta T} = \frac{0.7 \text{ V}}{1000 \text{ °C}} = 0.7 \text{ mΩ/°C}$$

Battery Voltage Temperature Compensation

Some battery chemistries have charge voltage requirements that vary with temperature. Lead-acid batteries in particular experience a significant change in charge voltage requirements as temperature changes. For example, for a lead acid battery, the 12-cell nominal voltage is a fixed charge of 2.25V/cell at 100°F. This battery float voltage, however, has a temperature coefficient which is typically specified as ±3.9 mV/°C per cell.

In a manner similar to the $V_{mp}$ temperature correction outlined previously, implementation of battery voltage temperature compensation can be accomplished by incorporating an LT3644 into the output feedback network.

For example, a 12-cell lead acid battery has a float charge voltage that is commonly specified at 2.25V/cell at 100°F, or 13.5V at –3.3 mV/°C per cell temperature coefficient.

$$R_{fb1} = -3.9 \text{ mΩ} \cdot \text{deg C} \cdot 12 \text{ cells} = -46.8 \text{ mΩ}$$

$$R_{fb2} = -46.8 \text{ mΩ} \cdot \frac{10 \text{ Ω}}{1 \text{ kΩ}} = -4.68 \text{ kΩ}$$

$$R_{fb3} = -4.68 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{3 \text{ kΩ}} = -1.56 \text{ kΩ}$$

$$R_{fb4} = -1.56 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{10 \text{ kΩ}} = -0.156 \text{ kΩ}$$

The NTC pin sources 50μA, and monitors the voltage across $R_{FB1}$. The LT3652 enters a current limit as $R_{FB1}$ increases. This current limit reduces maximum charger output current if the IC junction temperature approaches 125°C. In most cases, when excessive temperature conditions are relieved with only slight reductions in maximum charger current.

$$R_{fb1} = -3.9 \text{ mΩ} \cdot \text{deg C} \cdot 12 \text{ cells} = -46.8 \text{ mΩ}$$

$$R_{fb2} = -46.8 \text{ mΩ} \cdot \frac{10 \text{ Ω}}{1 \text{ kΩ}} = -4.68 \text{ kΩ}$$

$$R_{fb3} = -4.68 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{3 \text{ kΩ}} = -1.56 \text{ kΩ}$$

$$R_{fb4} = -1.56 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{10 \text{ kΩ}} = -0.156 \text{ kΩ}$$

The NTC pin sources 50μA, and monitors the voltage across $R_{FB1}$. The LT3652 enters a current limit as $R_{FB1}$ increases. This current limit reduces maximum charger output current if the IC junction temperature approaches 125°C. In most cases, when excessive temperature conditions are relieved with only slight reductions in maximum charger current.

$$R_{fb1} = -3.9 \text{ mΩ} \cdot \text{deg C} \cdot 12 \text{ cells} = -46.8 \text{ mΩ}$$

$$R_{fb2} = -46.8 \text{ mΩ} \cdot \frac{10 \text{ Ω}}{1 \text{ kΩ}} = -4.68 \text{ kΩ}$$

$$R_{fb3} = -4.68 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{3 \text{ kΩ}} = -1.56 \text{ kΩ}$$

$$R_{fb4} = -1.56 \text{ kΩ} \cdot \frac{1 \text{ kΩ}}{10 \text{ kΩ}} = -0.156 \text{ kΩ}$$

The NTC pin sources 50μA, and monitors the voltage across $R_{FB1}$. The LT3652 enters a current limit as $R_{FB1}$ increases. This current limit reduces maximum charger output current if the IC junction temperature approaches 125°C. In most cases, when excessive temperature conditions are relieved with only slight reductions in maximum charger current.
The switch node (Pin SW) trace should be kept as short typically less than 10nS to maximize conversion efficiency.

Layout Considerations

The LT3652 switch node has rise and fall times that are typically less than 30ns to maximize conversion efficiency. The switch node (Pin SW) trace should be kept as short as possible to minimize high frequency noise. The input capacitor (C1) should be placed directly in the hi/Lo terminal of the switch node to reduce switching noise. Short, wide traces on these nodes also help to avoid voltage drops from inductive ringing. The IC's decoupling capacitor should also be in close proximity to the IC to minimize inductive ringing. The SEN and BAT traces should be routed together and close to the IC to minimize inductive ringing. Short, wide traces on these nodes also help to avoid voltage drops from inductive ringing. The IC's decoupling capacitor should also be in close proximity to the IC to minimize inductive ringing.

High current paths and transients should be kept isolated from battery ground to ensure an accurate output voltage reference. Effective grounding can be achieved by considering switched current in the ground plane, and careful component placement and orientation can effectively steer these high currents such that the battery reference does not get corrupted. Figure 11 illustrates an effective grounding scheme using component placement to control ground currents. When the switch is enabled (loop #1), current flows from the input bypass capacitor (C1) through the switch and inductor to the battery positive terminal. When the switch is disabled (loop #2), the current to the battery positive terminal is provided through the freewheeling Schottky diode (D1). In both cases, these switch currents return to ground via the output bypass capacitor (CBAT).

The LT3652 packaging has been designed to efficiently reduce heat from the IC, via the exposed pad on the backside of the package, which is soldered to a copper footprint on the PCB. This footprint should be made as large as possible to reduce the thermal resistance of the IC case to ambient air.
C.3 LT3652

**REVISION HISTORY**

<table>
<thead>
<tr>
<th>REV</th>
<th>DATE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>C4</td>
<td>12/12</td>
<td>Removed reference to Nickel cell charging capability</td>
</tr>
<tr>
<td>C5</td>
<td>10/08</td>
<td>Corrected</td>
</tr>
</tbody>
</table>

**RELATED PARTS**

<table>
<thead>
<tr>
<th>PART NUMBER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTC4001/LTC4001-1</td>
<td>Monolithic 2A Switch Mode Synchronous Lithium-Ion Battery Charger</td>
</tr>
<tr>
<td>LTC4006</td>
<td>Small, High Efficiency, Fixed Voltage, Lithium-Ion Battery Charger with PowerPath™ Control, Constant-Current/Constant-Voltage Switching Regulator, Resistor Voltage/Current Programming, AC Adapter Current Limit and Thermistor Sensor and Indicator Outputs</td>
</tr>
<tr>
<td>LTC4002</td>
<td>Standalone, 4.7V ≤ VIN ≤ 24V, 500kHz Frequency, 3 Hour Charge Termination, 16V PEAK POWER VOLTAGE</td>
</tr>
<tr>
<td>LTC4008</td>
<td>4A, High Efficiency, Multi-Chemistry Lithium-Ion Battery Charger with PowerPath™ Control, Constant-Current/Constant-Voltage Switching Regulator, Resistor Voltage/Current Programming, AC Adapter Current Limit and Thermistor Sensor and Indicator Outputs</td>
</tr>
<tr>
<td>LT3652-8.2/LT3652-8.4</td>
<td>Monolithic 2A Switch Mode 2-Cell Li-Ion Battery Charger with Programmable Charge Current, Timer or C/10 Termination, Small and Few External Components, 4mm × 4mm QFN-16 Package –1 for 4.1V Float Voltage Batteries, 16V PEAK POWER VOLTAGE</td>
</tr>
</tbody>
</table>

**TYPICAL APPLICATION**

5A Solar Panel Powered 3 Stage 12V Lead-Acid Fast/Float Charger: 5A Charger Fast Charges with CC/CV Characteristics up to 14.4V; When Charge Current Falls to 0.1A Charger Switches to 13.5V Float Charge Mode; 1A Solar Panel Powered 3-Stage 12V Lead-Acid Fast/Float Charger; 1A Charger Fast Charges with CC/CV Complete Charger for 3- or 4-Cell Li-Ion Batteries, AC Adapter Current Limit, Programmable Charge Current, Timer or C/10 Termination, Small and Few External Components, 4mm × 4mm QFN-16 Package | 10μF 4.7μF

**PART NUMBER DESCRIPTION COMMENTS**

#### PowerPath is a trademark of Linear Technology Corporation.

**Characteristics**

- Up to 14.4V
- When Charge Current Falls to 0.1A Charger Switches to 13.5V Float Charge Mode
- 1A Solar Panel Powered 3-Stage 12V Lead-Acid Fast/Float Charger
- 1A Charger Fast Charges with CC/CV Complete Charger for 3- or 4-Cell Li-Ion Batteries, AC Adapter Current Limit, Programmable Charge Current, Timer or C/10 Termination, Small and Few External Components, 4mm × 4mm QFN-16 Package

**Components**

- 10μF
- 4.7μF
- 3652 TA04
- MBRS140
- 1μF 1N914 BZX84C6V2L
- 100k
- 499k
- 10μF
- 3652 TA04
- 100μF
- 10k
- 22μH MBRS340
- 0.1
- NCP18XH103
- WURTH
- B = 3380
- muRata
- 10k
- 309k
- 910
- 7447779122
- LINEAR TECHNOLOGY CORPORATION 2010

**Linear Technology**

- 1630 McCarthy Blvd., Milpitas, CA 95035-7417
- 1-800-729-4433
- www.linear.com