Low Level Design Document

Electrical Engineering Senior Design

The Dream Team: Andrew Carter, Matthew Elliott, Andrew Harms, Ben Keller

Table of Contents

1 Introduction

Education is of paramount concern in society today. While homeworks, quizzes, and tests are effective means to measure learning, a real-time feedback system would offer another, possibly better, aid to the instructor. We propose a wireless classroom response system. To make the system more accessible to a wide range of students, our system will be based on open standards and open-source technology to reduce cost of the system. For the design, we must consider specific wireless and portable power technologies.

2 Problem Statement and Proposed Solution

Each group of students learns at a different pace and responds positively to different teaching methods. Measuring student understanding allows an educator to tailor his or her own style to a particular group of students, improving the learning experience for everyone involved. In the typical classroom, this measuring is accomplished through periodic homework assignments and targeting single students with questions. Utilizing these methods exclusively, several lectures could pass before the teacher notices a gap in student understanding.

Currently available devices that attempt to fill this need are flawed in key areas. A commonly used solution, einstruction(tm)'s classroom performance system (CPS), suffers from slow response, non-rechargeable batteries, and an expensive recurring subscription fee. With this hole in the market, it is possible for a small engineering team to develop a cheap and effective product, allowing the teacher to modify each lecture in real time, no matter the size of the audience.

The proposed system will have a response pad for each student as well as a response receiver. The response pads will have a user input that allows the student to enter the information that the teacher requests. Our user interface will also convey information concerning battery and unit state, such as "on" and "charging". The response pad will send information to the response receiver wirelessly. All wireless communication from the response pads will be routed to a computer through a wired connection by the response receiver. The response receiver will have a user interface in order to display configuration modes. The computer will collect, analyze, and display the information. In addition to wireless communication, the response pads will also connect to personal computers to register for classes. The response pads will be powered by rechargeable batteries. The response receiver will be powered by the wall outlet.

3 System Description and Block Diagram

The student response system contains two main system types: the response pad and response receiver. The response pad consists of a microcontroller, wireless transceiver chip, battery, battery charger, wired communication controller, and a user interface. The response receiver consists of a microcontroller, wireless transceiver chip, power device, wired communication controller, and a user interface. The response receiver and pads communicate with computers via wired communication. The response pads and receiver communicate with each other wirelessly. Each IC block (wireless transceiver, battery charger, etc.) within the response system devices must communicate with a microcontroller.

The microcontroller will manage all the components of the student response system. It orchestrates the radio transmitting and receiving performed by the wireless transceiver chips on the response pads and receiver. It will control the operation of the battery charger as well as wired data transfer from response pads and receiver to computer. The microcontroller also handles user input as well as user output transferred by the user interface. This block will also include writing a significant amount of software.

Wireless communication will be used to link each individual response pad to the response receiver.

A battery charger IC will be used to charge a battery pack that is built into each response pad. Power must also be provided to the response receiver.

Wired communication between each system device and computer will be implemented. Response pads will need to communicate with computers in order to register for classes in which the student response system is used. This interface will be used to power the battery charger. A wired interface will also be used by the response receiver to transmit student response data to the computer.

The user interface of the response pads will consist of two components. Users input information to be sent by the response pad to the response receiver and eventually to the instructor. The user interface will also display information back to the user such as information to be sent and modes of operation. Modes may include "on", "charging", "transmitting data" and "ready to transmit". User data will be entered via keypads. The response receiver will also utilize a simple user interface to output configuration modes such as "on" and "receiving data".

Inter-chip communication will be implemented in order that each IC can communicate with the microcontroller. In this way the microcontroller can manage all the chips in each response device. Also the battery charger will use inter-chip communication to interface with the wired communication controller.

Software must be written for the computer in order to interface with the response pads and response receiver through wired communication. The computer will act as a charging station for the response pads and allow them to register for classes online. The computer will be used to collect, analyze, and display student feedback.

See Figure 6 for a block diagram of the entire system.

4 System Requirements

- 4.1 Overall System

The final system will consist of multiple response pads, a response receiver, and accompanying computer software for classroom integration. Each of these pieces is integral to our overall system.

The response pads are the student interface to the system. The student will interact with the response pads via the LCD screen and the keypad. The LCD will display pertinent information and the keypad will allow input into the device, either answers to questions that are collected in class or information required for registering the device. The response pads will also interact with the rest of the system via the wireless link which will use the Zigbee protocol. The device will use USB to interface to a PC that can be used for powering the device and connecting to a computer for registration and firmware updates. Since the device also needs to be portable, it will include a rechargeable battery. Finally, the response pad will contain a micro-controller that acts as the brains of the device controlling the other components.

The response receiver is the central receiver that collects the data from the response pads, processes the information, and sends it to the teacher. The response receiver must have a wireless communication link that interfaces with the response pads, so it will also use the Zigbee protocol. A USB interface will act as both the connection to the computer and the power source for the device. Finally, the response receiver will need a microcontroller to control all components.

The accompanying computer software is the teacher interface into the system. The data is made available to the computer from the response receiver for viewing and/or presenting by the teacher through a USB interface. The software will also be able to interface into PowerPoint, or an equivalent program, for presentation to the class.

- 4.2 Subsystems

Microcontroller

The microcontroller orchestrates all the operation in our device. We need our microcontroller to have sufficient clock speed to manage all the devices and operations. Being a portable system, the response pad's microcontroller should be power efficient. Since the microcontroller must be programmed, it's language should be versatile yet comprehensible and intuitive. Because of multiple peripherals, we need sufficient input/output ports on our controller. Since the response pad could go into production, the microcontroller should be relatively inexpensive in bulk. Lastly, the type of packaging available for the chip is a concern, primarily in prototyping. We need to be able to easily troubleshoot and test the chip during the design process. Requirements for the response receiver are not as stringent since it will not use battery power.

Wireless Communication

Since our response pad is portable, wireless communication to the response receiver is a logical choice. The wireless protocol should be power efficient yet have sufficient range for our application. It should also be easy to interface to our microcontroller (this is more of a final part decision than protocol decision, however). IC Packaging is more of a concern here, since there are serious RF design decisions based around packaging. The protocol should be robust to interference, such as competing protocols on the same frequencies. Lastly, cost is a factor for major production.

Power

For portable power, a battery will be used. The battery should hold the requisite amount of power, along with current draw ratings on the order of what we need. A circuit will need to manage battery charging. It should be relatively easy to interface to our microcontroller.

User Interface

The user hardware interface must supply the end user with relevant information during usage. The response pad needs to have a power efficient user interface, while the response receiver can use more power if necessary. The form factor of the interface is important for the response pad, since it should be a hand held device.

We will write software to collect, analyze, and display student response data from the response receiver on a computer. Analysis will include totals of answer choices. Results will be displayed in text format.

Inter-chip Communication

Given the large number of subsystems, there should be an efficient inter-IC communication protocol. It should be fast enough to process student responses.

PC Communication

There needs to be communication to a computer, both on the response pad and response receiver. This PC communication should be easy to interface to our microcontroller. The protocol should also be widely available on computers. The speed of the interface is a concern, especially on the receiver end to process question responses.

- 4.3 Future Enhancement Requirements

One future enhancement is a direct interface to PowerPoint, or an equivalent piece of presentation software. This type of interface would make it easy for professors to integrate the feedback provided by our system into their lectures. Other enhancements include customizable analysis and graphical display software, and online response pad registration.

5 Low Level Design

The following is a detailed description of each major component of our design. See Figures 1 – 5 for a full schematic layout of each section.

- 5.1 Microcontroller

The microcontroller will orchestrate the functions of other chips. It's main loop consists of scanning the matrix keypad for input and updating the LCD screen. The USB and Zigbee use hardware interrupt lines on the microcontroller. The microcontroller will use the functions of the chips as described below. The microcontroller is programmed through the JTAG (Joint Test Action Group) interface. This is an in-circuit programming interface that gives us a simple way to reprogram the hardware after it is built. This is especially important for us, given the number of surface mount devices we have.

- 5.2 USB Circuitry

The USB chip requires three microcontroller functions for correct operation. A function to write to various entries in the USB register using SPI. This function will send an eight bit address followed by an eight bit entry to write to the register. A function to read from various entries in the USB register using SPI. This function will send an eight bit address and then receive the eight data bits contained at that address. An interrupt function to respond to the interrupt output pin on the USB chip. This function will call the previously defined read function to determine why the interrupt occurred and then respond by reading the newly available data from the chip or sending more data to the chip. The Zigbee chip is bundled with a wide variety of functions available that implement the IEEE 802.15.4 standard and the Atmel Zigbee stack included. These are available on the disks that came with the the chip and hardware development kit and are summarized in the Atmel IEEE 802.15.4 MAC User guide. A display function to be used with the LCD controller.

The USB chip includes four general-purpose I/O pins controlled by SPI. Using the output pins, we will control the battery chips two settings: 100mA or 500mA charging from USB, and suspend USB charging flag.

- 5.3 Wireless communication

The wireless communication will be accomplished using the Atmel IEEE 802.15.4 Radio Transceiver, the AT86RF230. The chip is based on the IEEE 802.15.4 standard and Zigbee stack protocol. The chip includes a 2.4GHz transceiver RF front-end with a 100 Ohm differential antenna input/output. It also includes the analog and digital circuitry to implement the Zigbee MAC based on the IEEE 802.15.4 standard and the interface to the microcontroller. The interface to the microcontroller is an SPI interface.

This Zigbee chip has a minimal list of outside components that it needs. The RF interface requires a balun and two capacitors to interface from the balanced, 100 Ohm line coming out of the Zigbee chip to the 50 Ohm unbalanced line that will be our antenna. The antenna will be a standard dipole antenna. We also need four capacitors to use as power supply decoupling capacitors. We need the decoupling capacitors because we actually have three different power and ground sources: RF, analog, and digital. Keeping the power for each of these as isolated as possible will help to ensure maximum performance from our system. The chip also needs an external oscillator running at 16MHz. We are using ceramic resonator for our oscillator, and the capacitors are included on the resonator component. Finally, the chip has eight pins which interface with the microcontroller. Five of these pins are used for the SPI interface. The other three pins are used to change particular states in the chip and as interrupt flags for the microcontroller.

The Zigbee chip we are using is built to handle most of the MAC functions required by the IEEE 802.15.4 standard. We still must set up a message structure and define how the data will be handled within our system. The information we are dealing with is minimal - i.e. not more than several characters. Our messages that will be sent from the nodes to the base station will consist of two parts: a 16-bit address unique to each device (MAC address) and an 8-bit piece of data. This data frame is sent to the Zigbee chip from the microcontroller, sent out by the chip, received by the base station, and sent back to the base station microcontroller so that the base station can identify who sent the message. For the MAC functions from the IEEE 802.14 standard, the microcontroller controls the Zigbee chip by sending requests to transmit or receive. After the microcontroller sends the piece of data to transmit, it sends the request to transmit and the chip takes care of the remaining MAC and PHY functionality. Likewise, when the Zigbee chip receives a piece of information, a flag is sent to the microcontroller signaling that a message is ready so the microcontroller can read the received message.

For testing on the Zigbee chip, we will be using the evaluation board and hardware development kit that we ordered from Atmel. The kit includes a suite of testing equipment, both hardware and programs.

- 5.4 Power

We will use the single cell Powerizer Polymer Li-Ion rechargeable battery to power our response pads. At 65x34x6 mm, this battery will meet our voltage (3.7 V) and current (maximum 2.8 A) requirements while conforming to the desired hand held form factor size limitations. The maximum charge rated at 1400 mAh should be more than sufficient. DC power will be supplied from the wall by a 5V AC adapter. USB will also provide 5V power from a computer.

Along with some supporting circuitry, Maxim's MAX8677A 1.5 A Dual-Input USB/AC Adapter Charger and Smart Power Selector will control the power for our units. The chip has a 40mOhm system to battery switch. We will design the supporting circuitry for separate USB and DC inputs, so that we may use either (DC takes precedence when both connected) to power the response pad or receiver. If a battery is connected, the MAX8677A will automatically charge the battery when external power is connected and then power a unit from the battery when external power is removed. Although unnecessary in the final product, DC power will be important for testing before we get the USB connection functioning correctly.

The supporting circuitry will include capacitors, resistors, and a voltage regulator chip. Decoupling capacitors are placed on the battery, DC input, USB input, SYS power output, VL output, as close as possible to the MAX8677A chip to eliminate any AC signal superimposed on the DC power (and logic) lines. A charging timer capacitor sets the fast charge and prequal fault timers which are involved in a safety function that will not let the battery charge for too long. Logic output pull-up resistors are used to ensure that inputs to logic systems settle at expected 3.3 V logic levels. A thermistor is used to monitor the battery or system temperature. Charging is suspended when the thermistor temperature is out of range. A pull up resistor equal to the thermistor resistance is also used. Resistors are used to set the DC input current limit as well as the fast charge current. The MCP1700 3-Pin SOT 23 low-dropout, low quiescent current, voltage regulator chip will be attached to the SYS output to drop and regulate the voltage at 3.3 V. The voltage regulator output will power the entire board. The voltage regulator circuit requires input and output bypass capacitors. All is shown in schematic.

Logic inputs from the microcontroller include charge enable, USB suspend, and maximum USB input current (either 100mA or 500mA). These logic inputs will be controlled by logic output pins on the USB chip (which are in turn controlled by the microcontroller through SPI). Logic outputs include active low fault (battery timer expires before charging is complete), USB power, DC power, charging, and charge complete indicators. Outputs will be indicated by LEDs.

The MAX8667A implements a number of safety features including thermal voltage regulation and monitoring for both chip and battery, input voltage limiting and over-current protection.

The chip will be controlled through the general purpose input and output pins on the USB controller chip to conserve microcontroller pins.

The MCP1700 voltage regulator will be tested first by measuring output voltages and currents in response to various input voltages ranging between 3 and 6 volts. The output should be a constant 3.3 V. The voltage regulator will be connected to the SYS output and testing of the MAX8766A will begin by measuring voltage and current from the SYS , logic, and voltage regulator outputs with 5 V DC power connected through an AC adapter but without a battery. A battery will then be connected along with DC power. The system will be monitored while the battery charges. SYS, logic, and voltage regulator output voltages and currents will be measured along with the battery voltage. Charging will be suspended using the logic input charge enable. After the battery is charged (checking automatic charge stop), DC power will be removed and the system voltages and currents will be monitored. Next, USB will be tested. All input power configurations will be tested including USB, USB and battery, USB and DC, and lastly USB, DC, and battery. All logic inputs will be tested. Before any power input is connected to the chip, the voltage will be measured. All results will be matched with the operating characteristics given in the specification sheet. After the system is built, microcontroller control of the logic inputs and power to other subsystems will be tested. Voltages and currents will be tested using an oscilloscope.

- 5.5 User Interface

Response pad user input will be accomplished using a 3x4 matrix keypad. The microcontroller will poll the keypad every 100 ms by setting each of the four horizontal lines high one by one and reading from the three vertical lines. This will detect any depressed key.

Response pad user feedback will be supplied via a 16x2 character LCD screen. The Hitachi HD44780U (LCDII) Dot Matrix LCD Controller/Driver will be used to control the LCD (Note: this is the same controller used in class for task 3). Two parallel 8-bit register interfaces (instruction and data) will be used between the microcontroller and the LCD controller. The LCD uses a parallel 16-bit logic pin to interface with the LCD controller and supply power. 8 pins are used for data transfer, 6 pins are used for power and control, and 2 pins are unused. The microcontroller will send information through the LCD controller to display data such as "ready to send" and "sent" statuses as well as user response to be sent to receiver on the LCD screen.

We can test our LCD and LCD controller using “Hyperterminal” in a similar manner to task 3. We can then connect both the LCD and keypad to the microcontroller. Final testing will consist of user input from keypad being transferred to the LCD through the microcontroller.

The collection, analysis, and display of data received from response receiver will be implemented by application software on a PC using a Linux OS. Our software will store the data received from the response receiver by checking the response pad ID number of the data packet and placing the response into its appropriate slot in an array. The length and ID numbers of this initial array will be predetermined through registration for the class. Questions will be in the form of multiple choice with four answer choices. The initial array will be separated into 4 new arrays (one for each answer choice) containing IDs corresponding to each student response unit. A text output will display the ID numbers in each array as well as the length of each array (answer totals). If the instructor chooses to supply a "correct" answer, a fourth array and array length will be displayed equal to one of the four previous arrays.

Functions included in the application software include receive (forms initial array), parse (forms 4 arrays from initial array based on answer choice), count (counts entries in an array), and display (displays text output of value).

Testing of the PC application software can be accomplished by checking whether or not the answer arrays are filled correctly and the appropriate information is subsequently displayed after using the response system.

- 5.6 Interchip Communication

Interchip communication will be implemented with SPI.

Testing will consist of checking correct communication between microcontroller and USB or Zigbee chip.

- 5.7 PC Communication

Communication between the PC and the response pads and the PC and the response receiver will be implemented with USB. The response pads and receiver will be equipped with the MAXIM USB Peripheral Controller MAX3420EECJ+. This chip communicates with the microcontroller directly via an SPI interface. The USB chip automatically handles USB flow control, double buffering, low level USB signaling details, and all USB timing details. On the PC side, we will write a USB driver based on the USB human interface device class. The communication between the PC and the response pads is for device registration and is part of future enhancements.

The response receiver will operate in one of two modes, which will be determined by input from the PC. In standby mode, the response receiver will ignore all Zigbee packets and will not transmit over USB. In repeater mode, the response receiver will forward to the PC the 16 bit unique identifier and 8 bit data packet that it pulls from each Zigbee payload it receives. Each PC request to the response receiver to change operating modes will be acknowledged by the receiver when it enters the new mode by echoing the mode selection command.

To test the PC - response receiver connection, the PC will echo each USB packet payload to the terminal. The PC will then instruct the response receiver to switch between the two possible modes, and we will observe the terminal output to insure that the response receiver acknowledged the commands. We will then put the response receiver in repeater mode and have a response pad broadcast a packet and the PC terminal should display the 16-bit identifier of the response pad and its 8-bit data packet.

6 Preliminary Bill of Materials

The following is a list generated by EAGLE's scripting language. Unless otherwise noted in this document, parts are available from Digikey, along with free samples.

Partlist exported from /Users/mut3/Desktop/LLD/dreamteam_projectboard.sch

Part	Value
C1	.1uf
C2	.1uf
C3	.1uf
C4	.1uf
C5	1uf
C6	1uf
C7	1uf
C8	1uf
C9	1uf
C10	2.2pf
C11	22pf
C12	22pf
CB1	1uf
CB2	1uf
CB3	1uf
CB4	1uf
CBATC	4.7uF
CDCC	4.7uF
CERRES	Ceramic Res
CL	.1uF

CSYSC	10uF
CT	.068uf
CUSBC	4.7uF
IC1	MEGA64-A
JP1	Pins
JP2	Pins
JP3	Pins
JP4	Pins
JP5	Pins
LEDCHG	LED
LEDDOK	LED
LEDDONE	LED
LEDFLT	LED
LEDUOK	LED
MCP1700	Maxim IC
R1	33
R2	33
R3	10K
R4	680
R6	THERMIST
RCHG	150
RDOK	150
RDONE	150
RFLT	150
RISET	3k +/- 1
RPSET	1.5k +/-
RT	10k +/-
RUOK	150
S1	
TR1	
U$1	MAX3420E
U$2	12Mhz +-
U$3	MAX8677A
U$4	AT86RF23
U$7	16Mhz +-

7 Conclusions

Because student feedback is essential for an instructor to pass information to the students as efficiently as possible, many traditional methods have been employed for many years. These include homework assignments and answering specific questions from students. The technology available today also makes another solution possible. Real-time feedback of student understanding can be accomplished through a student response system. While solutions do exist, they have several drawbacks that have prevented widespread use. Some of these problems include cost to the students and the classroom interface for the instructor. Our solution solves the problem of creating a viable, efficient student response system while also reducing student cost and leaving the system open to future development and improvement. Open-source solutions can keep cost to a minimum by making the solution open to the public. It can also be improved by others who have free access to all documentation and plans.

8 References

Atmel, "8-bit AVR Microcontroller with 64K Bytes In-System Programmable Flash", August 2007

Atmel, "IEEE 802.15.4 MAC: User Guide", September 06.

Atmel, "AVR2005: Design Considerations for the AT86RF230", August 07.

Atmel, "ZigBee IEEE 802.15.4 Radio Transceiver: AT86RF230", May 22, 2007.

Maxim, "MAX8677A 1.5 A Dual-Input USB/AC Adapter Charger and Smart Power Selector"

Microchip, "MCP1700 Low Quiescent Current LDO Voltage Regulator"

www.sgbotic.com, "Keypad"

www.futurlec.com, "LCD"

www.batteryspace.com, "Li-Ion battery"

Hitachi, "HD44780U (LCDII) Dot Matrix Liquid Crystal Display Controller/Display"

Maxim 19-3781; Rev 2; 6/07

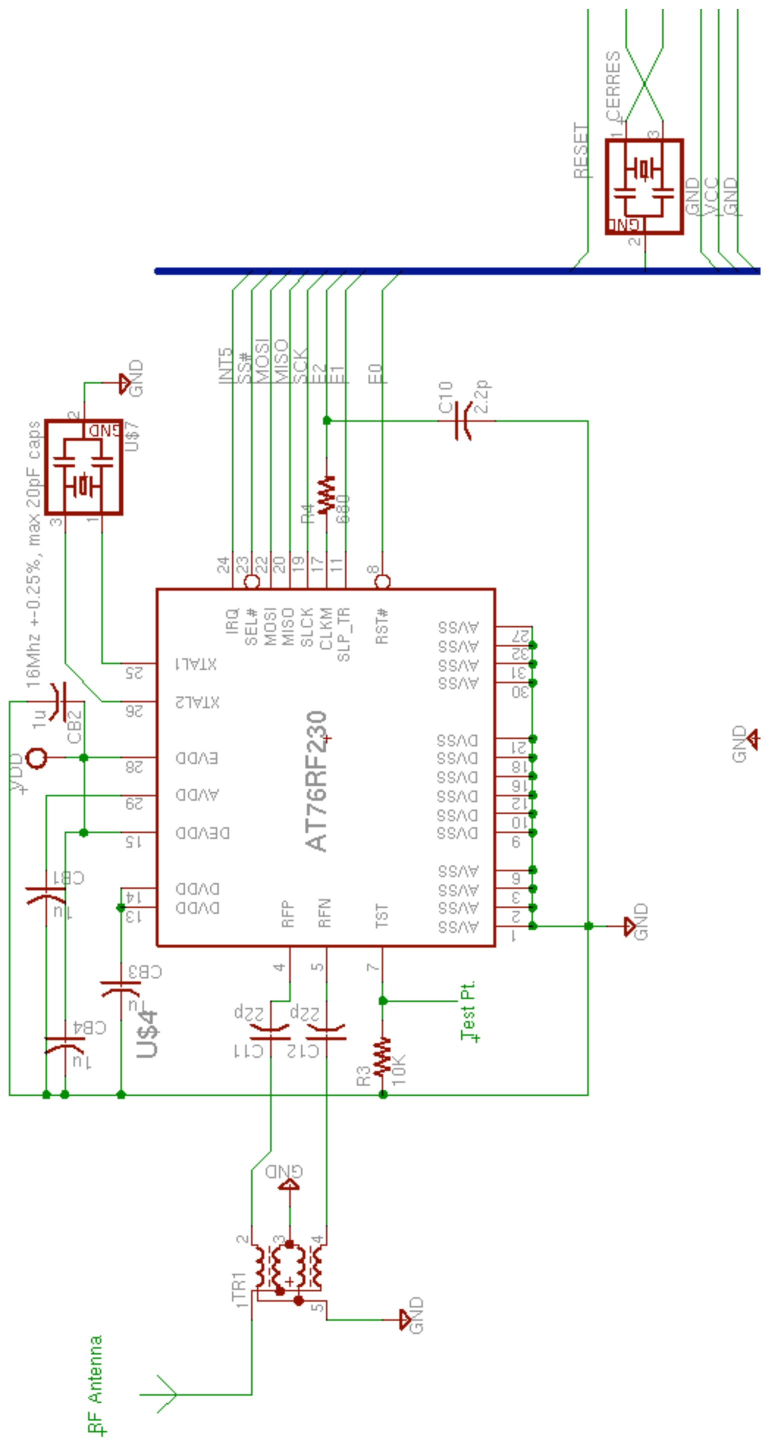

Figure 1: AT76RF230 (Zigbee Transceiver Chip) Schematic Layout

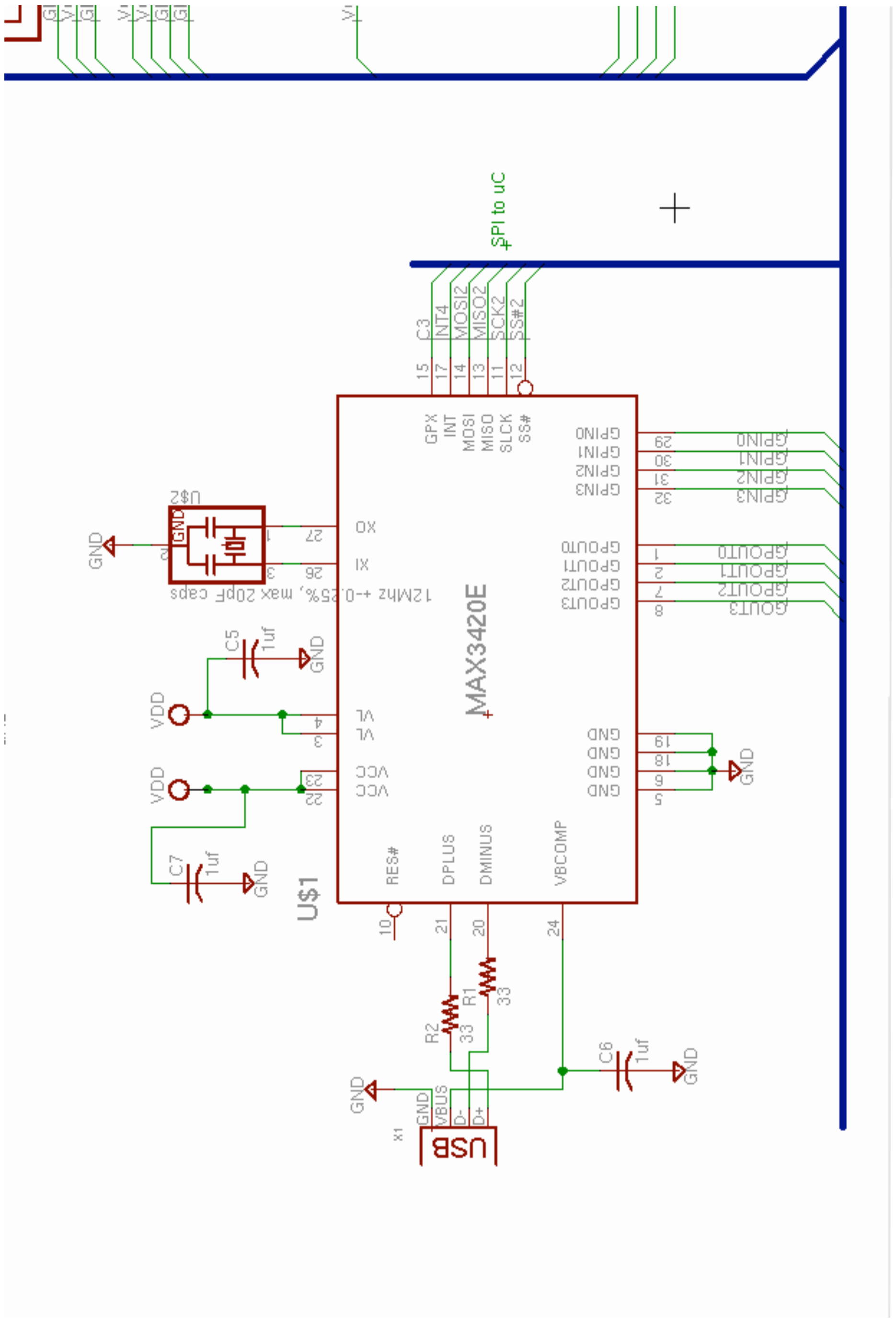

Figure 2: MAX3240E (SPI to USB Controller chip) Schematic Layout

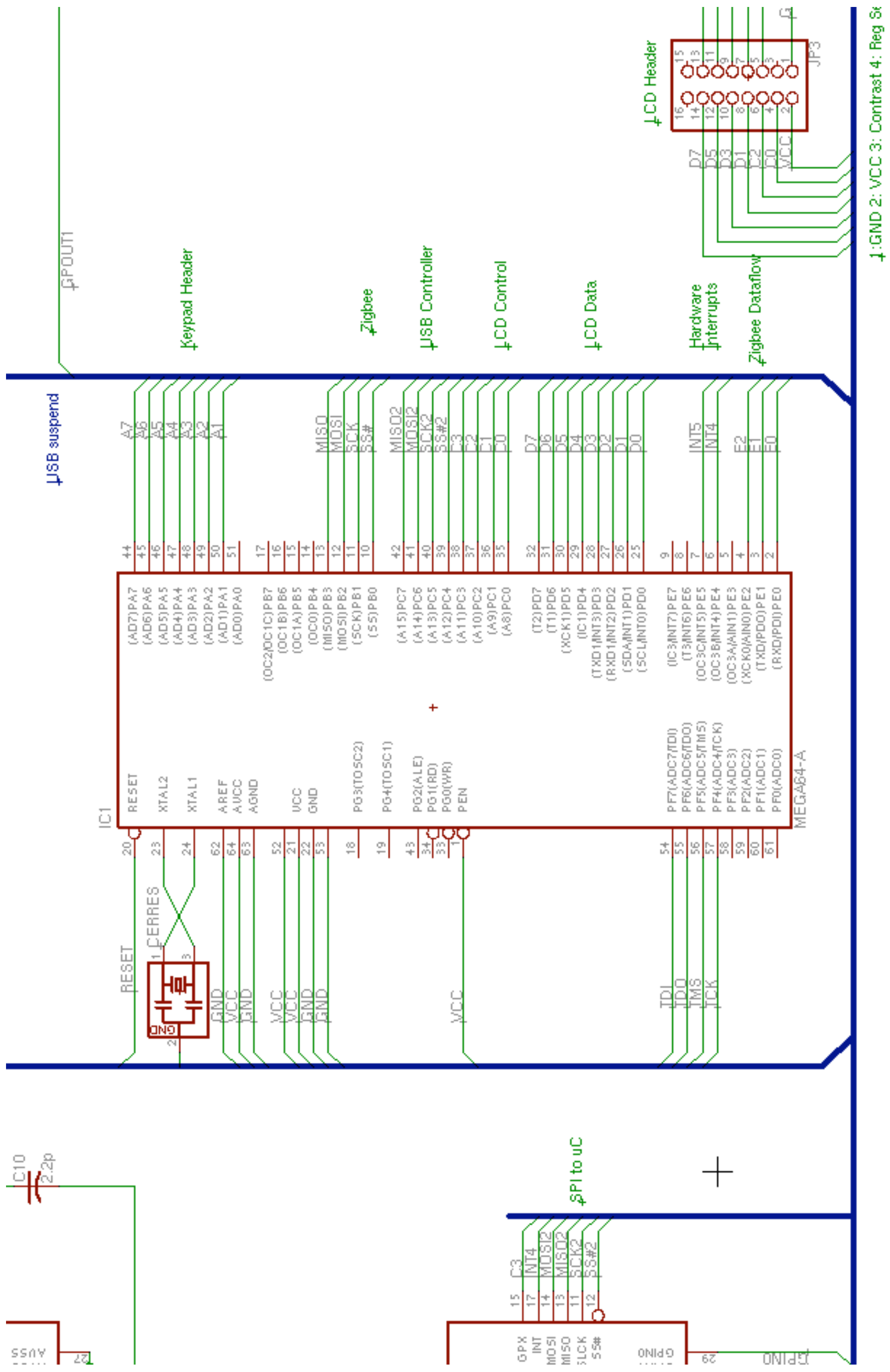

Figure 3: ATMEGA64 (Atmel Mega 64 Microcontroller) Schematic Layout

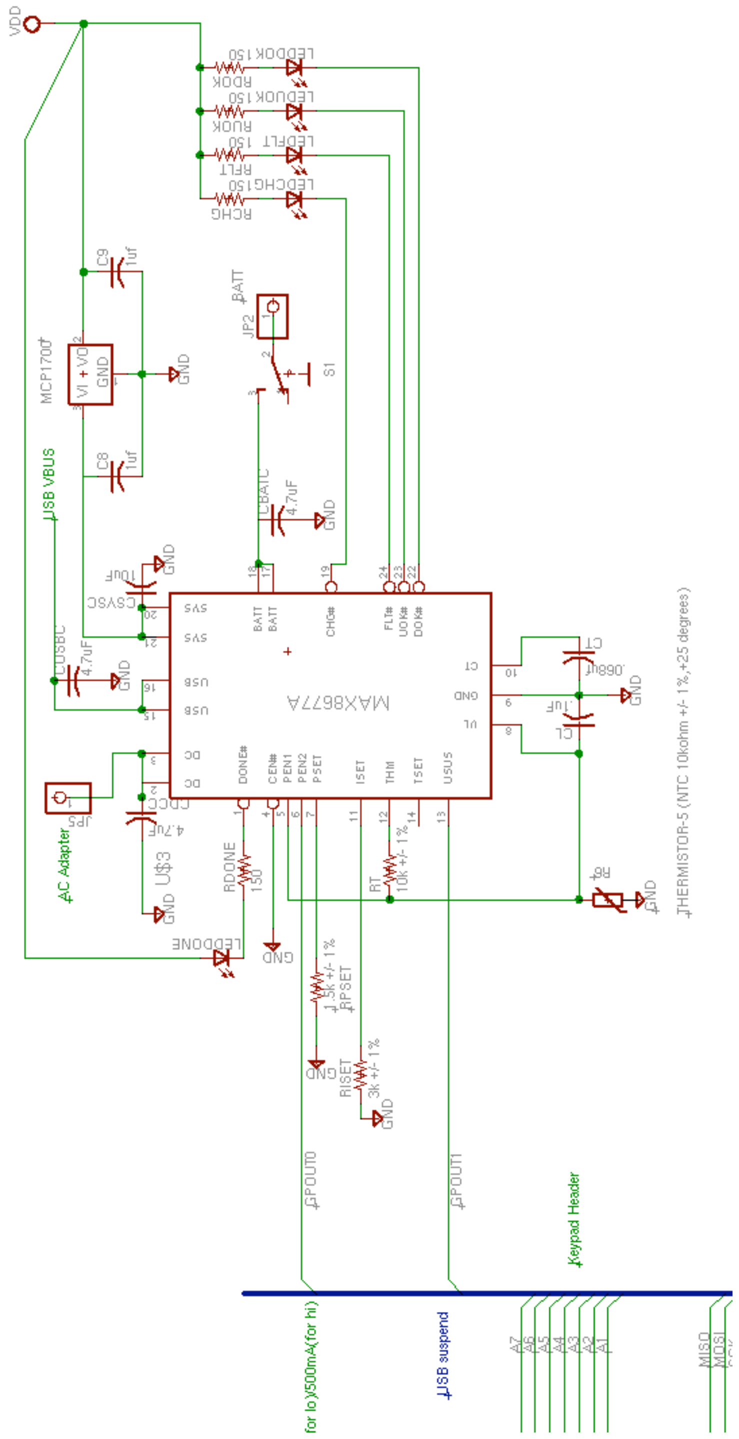

Figure 4: MAX8677A (Li-ion charging chip) Schematic Layout

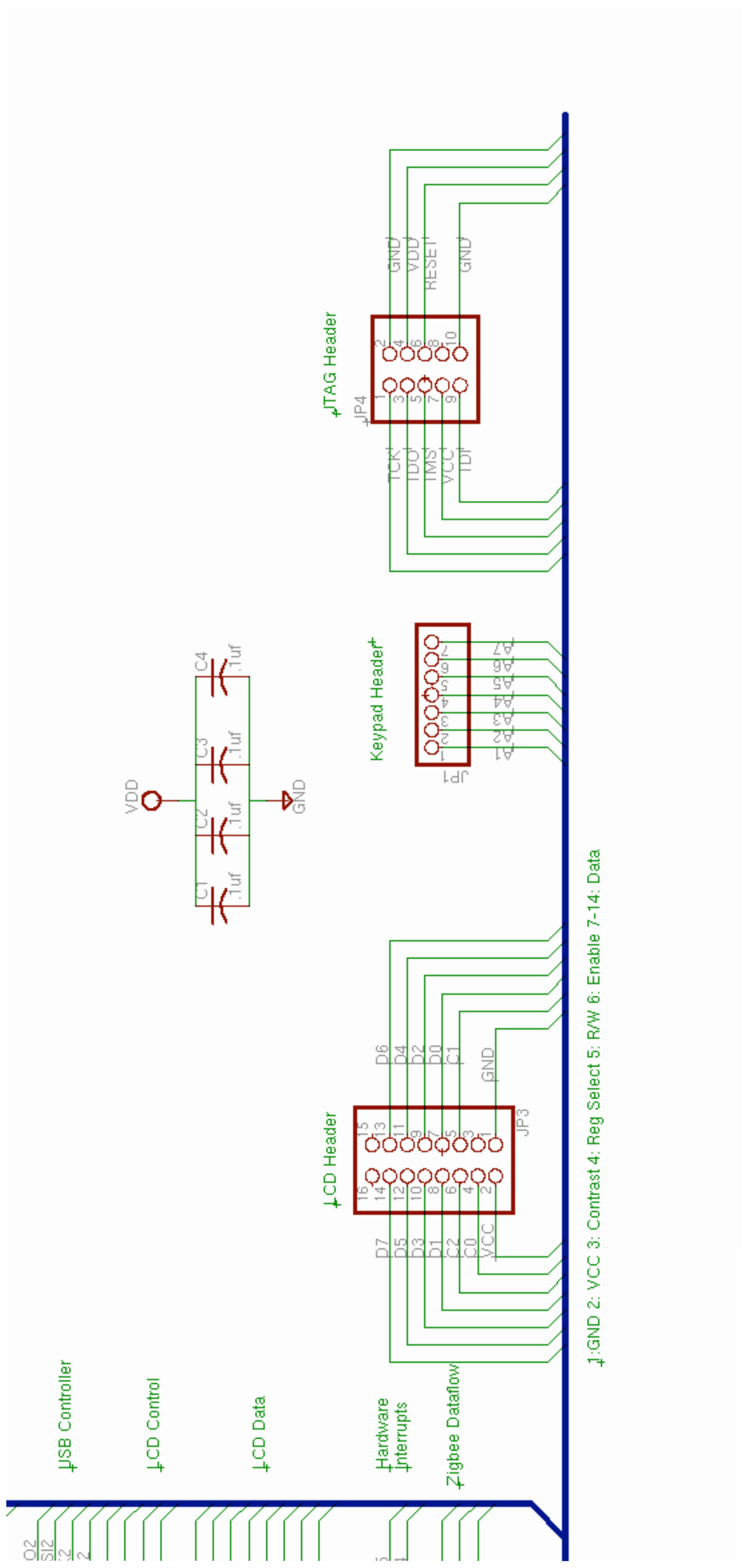

Figure 5: Pin headers for off-board hardware, Schematic Layout

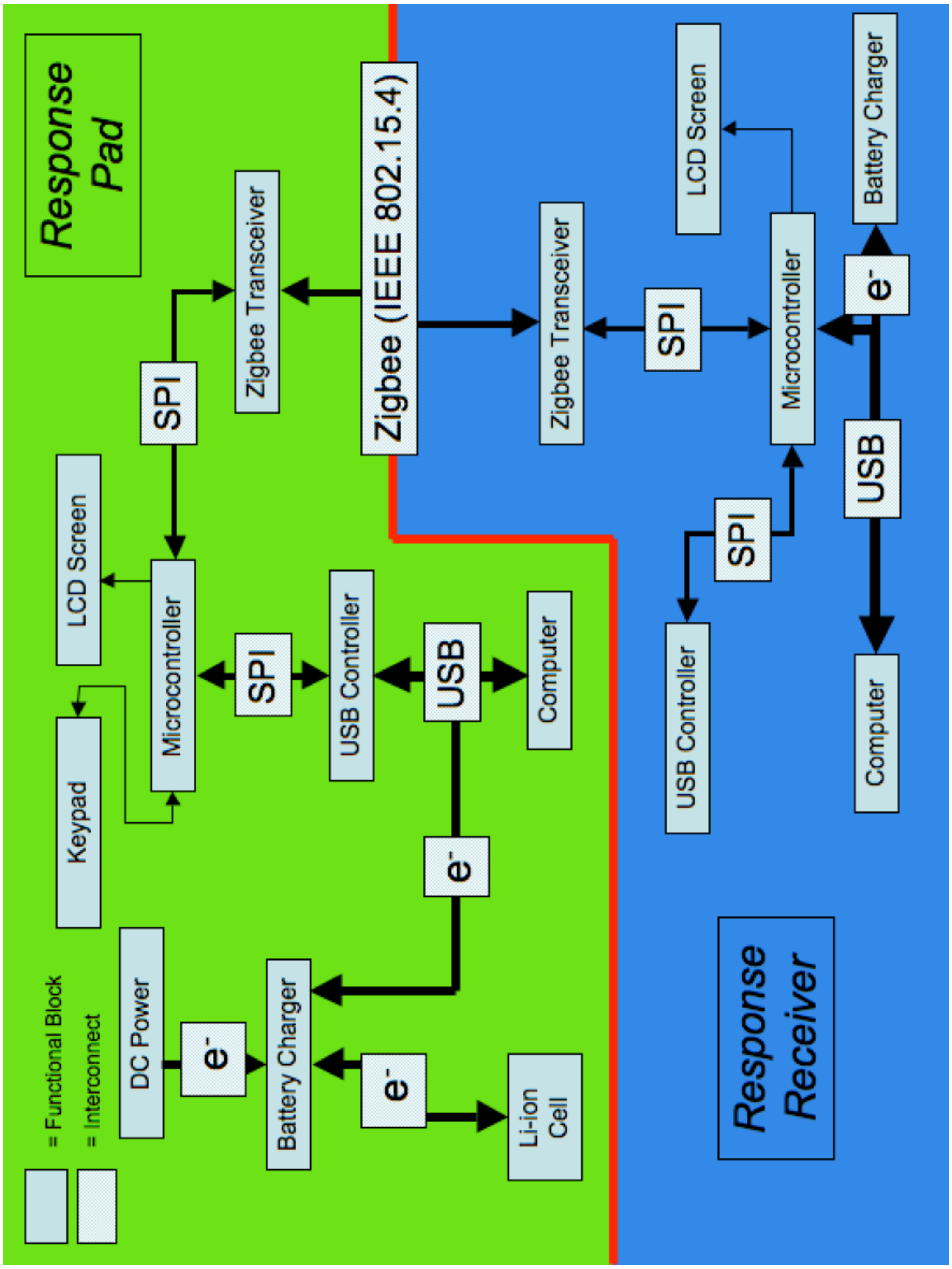

Figure 6: System Block Diagram