SECURE SOLAR POWER SOURCE

A Design Project Report

Presented to the Engineering Division of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering (Electrical)

By

Vladimir Kozitsky

Project Advisor: Bruce R. Land

Degree Date: May, 2004

Abstract

Master of Electrical and Computer Engineering Program

Cornell University

Design Project Report

Project Title: A Secure Solar Power Source

Author: Vladimir Kozitsky

Abstract: A self-sustaining power system was designed to power a remote base-station of motion activated video equipment. The system provides 50 watts of power for four hours daily and energy is replenished using a photovoltaic panel. Relevant system operating parameters along with security information are transmitted wirelessly to a battery powered handheld receiver. An Atmel Mega32 MCU monitors relevant parameters and controls the transfer of power from the PV panel to the main battery using a maximum power point tracking algorithm. It also generates control signals to protect the main battery from over-current, over-voltage, and thermal runaway.

Report Approved by

Project Advisor: ____ Date:____________

Executive Summary

To monitor animal behavior unobtrusively in the natural environment, it is important that the environment is not excessively disturbed by the observer. Video recording provides this capability but the problem is that once the batteries are depleted, they must be removed from the field, recharged, and returned. The batteries are heavy and the act of removing them disturbs the environment.

The design detailed herein provides a solution through the use of a photovoltaic panel coupled with a smart charge controller to ensure that the batteries are recharged in the field by solar energy. An Atmel Mega32 controller measures ambient temperature, battery and panel voltages, charging/discharging currents and uses the information to control the output of the main switching power regulator as well as relays that control load and PV panel power. The controller optimizes battery longevity by protecting against over charge, excessive discharge, and thermal runaway. A MPPT algorithm is used to optimize the efficiency of power transfer from the panel to the battery.

To provide security of the main system, movement is sensed using an accelerometer and if the main system is disturbed, an alarm signal is sent wirelessly to a handheld receiver. The range of the wireless link is maximized at 500 meters and is more then sufficient for the application. Along with security information, current system state is updated periodically and transmitted to the operator.

This design meets the requirements that were initially in place. It is within the $350 budget and provides an efficient and stable power source. Within the requirements, overall system operating efficiency was a major objective and guided the decision making process.

Table of Contents:

Introduction……………………………………………………………………………..1
Design Problem and Requirements……………………………………………………1
Range of Solutions………………………………………………………………………1

Implementation:
Handheld Receiver....…………………………………………………………………3
Photovoltaic Panel……………………………………………………………………7
Voltage Sensing…..…………………………………………………………………8
Current Sensing…..……..……………………………………………………………9
Load and Panel Control………………………………………………………………11
Temperature Sensing…………………………………………………………………11
Motion Sensor/Alarm…………………………………………………………………12
RF Transmitter……………………………………………………………………………13
Battery……………………………………………………………………………………15
Main Power Converter…………………………………………………………………16
Main System Display……………………………………………………………………19
Embedded Application…………………………………………………………………20

Cost Analysis……………………………………………………………………………22
Results……………………………………………………………………………………23
Conclusion…………………………………………………………………………………25
Instructions for Use…..…………………………………………………………………26
References…………………………………………………………………………………26

Appendices:
Photos……………………………………………………………………………………26
Receiver Code…..…………………………………………………………………………30
Main Controller Code……………………………………………………………………34
Schematics…………………………………………………………………………………43
PV Datasheet…………………………………………………………………………..…..49
Laipac RX/TX Datasheet…………………………………………………………………51

Introduction

The Atmel AVR family of controllers provides the engineer with cost effective solutions in designing low power control systems. For under $50, a smart charge controller and battery monitor was designed using the Mega32 MCU. The solutions on the market are generic and cost upwards of $100 just for the controller. Controllers that provide MPPT double the price to $200.

Design Problem and Requirements

The design problem was to develop a self sustaining powers source utilizing solar energy. This source would be used to power a remote base station consisting of video recording equipment. The recording is activated via a motion sensor and in LP mode the VHS tape can hold 4 hours of video. It was determined that the maximum power draw during operation would not exceed 50 watts. Thus the power source must provide 50 watts for four hours and the panel through the charge controller must replenish that energy throughout the day. Since the system is relatively expensive, it is important that any attempts to disturb it are made known to the operator as soon as possible. Further, there was a budget constraint of $350.

Range of Solutions

Various solutions exist in controlling the transfer of energy from the PV panel to the battery. The simplest is to connect the PV panel directly to the battery however this result in the panel voltage dropping to the battery voltage and 20-30% of the potential power that the panel could supply is wasted because at the reduced voltage it will not be in its maximum power window. Another solution, just as bad is to use an adjustable linear voltage regulator to set the charging voltage. In this case the extra power that was to be gained will be lost due to the voltage drop in the regulator. A better solution is to use a switching regulator. Switching regulators provided efficiencies as high as 95%. For this design a variable output buck converter was utilized since in the process of charging a battery using a constant voltage charger, the voltage needs to be varied for different states of battery charge. Finally, a digitally controlled current converter can be used but this was decided against in this design because the peak power of the panel is best determined by its closed circuit voltage. In the conclusion of this design, a combination of step down converter and a limiting current source are proposed to further improve conversion efficiencies.

The tradeoff between battery capacity and mass was difficult to make. A lighter battery makes it easier to carry the system deep into the field but comes at the expense of reserve capacity. It was determined that a 16Ah battery would be sufficient to power the system for one day in case the panel was disconnected or could not deliver sufficient charge due to severe overcastting, rain, etc. However, a low capacity battery is more likely to be drained to almost 0% SOC and this is very bad for battery life. By over designing and using a higher capacity battery, these potential problems are alleviated and allow for future reconfigurations where more power is needed.

The tradeoff in PV panel output power and price was easier to make. In examining the worst case scenario, it was determined that a 20 or even 30 watt panel would prove insufficient in enough cases to cause problems that would lead to system shutdown due to insufficient energy replenishment. Because the batteries were donated, there was more room in the budget for the necessary higher output panel.

Handheld Receiver:

The handheld receiver unit enables the monitoring of the main system remotely. Requirement for the design of this unit include long battery life, portability (250gm), ergonomic correctness, and effective warning in case the main system is disturbed.

The receiver acquires the relevant data wirelessly at a frequency of 434 MHz from the main system using a Laipac RLP434 receiver module. The module provides two data outputs; digital and linear (analog). In order to ensure signal integrity, the digital output was chosen and was fed directly into a Holtek HT12D decoder IC. The decoder IC processes the incoming data and sets a digital output high when valid data has been acquired. Each data packet consists of 12 bits. The first 8 bits are the address of the packet and the last four bits hold the actual data. The packets are generated using a Holtek HT12E encoder on the transmitter side. The decoder has 8 address pins that are set digitally and it compares the address field of each incoming packet to its own address. If a match takes place, the data ready line is set and the Atmel 8515 MCU reads the four bits of valid data at the digital port. The functionality of the decoder can be implemented in software on the MCU however this would unduly tie up the resources of the MCU requiring a faster crystal to operate robustly which in turn would lead to increased energy consumption and inherently a reduced bandwidth of data since even at 16Mhz, the maximum estimated packet throughput would be at a rate of 20Khz. The decoder IC executes the comparison at 150 KHz.

The initial design of the communication protocol involved used a fixed address for the encoder/decoder and differentiating the nine packets of data using a header and a checksum to ensure data integrity. This was implemented as follows; the receiver would wait until it received an initialization byte which would trigger the recording of the subsequent nine incoming packets as data and the tenth packet would be used as the checksum. The checksum consisted of the sum of all nine 4-bit data packets truncated to 4-bits. The data ready line would go low with the termination of a packet and go back high when a new one was ready. The transmitter would transmit each packet for 500ms and go quiet for 500ms. In testing, it was quickly discovered that this design was not satisfactory. The key problem was the sensing of the data ready line. It would go low when the transmitter was done transmitting, but more importantly, it didn't always stay high during a valid transmission. Every time it went low, the receiver assumed that a valid data byte was received and preceded through the data processing algorithm. This concurrency problem could not be solved and initially panic set in. At a bare minimum, the receiver would be able to interpret one packet of data and thus an alarm warning could be implemented.

This problem was solved by allowing the MCU to control the address lines of the encoder/decoder. Each data packet was given a unique address and a state machine running on the receiver MCU would change states with every valid data packet. The address bits of the decoder were changed with every state transition. In the unlikely event that the receiver misses a data packet and hung in past state, the data is transmitted multiple times. Unlike the decoder, it was discovered that the encoder drew current on the address lines and inverting buffers were used to stabilize the lines by providing more current then the MCU could supply through its output pins. As a consequence, the inputs to the inverting amplifiers were inverted so that the output into the encoder is consistent with what the decoder is expecting at the receiver. No inverting amplifiers were needed for the decoder.

Table 1 is below,

TX ADDR	RX ADDR	Data (4-bit)
15	0	Init
14	1	Reserved
13	2	Voltage H
12	3	Voltage L
11	4	SOC H
10	5	SOC L
9	6	Out Temp H
8	7	Out Temp L
7	8	Bat Temp H
6	9	Bat Temp L
5	10	Charging/Alarm

Information such as current battery voltage, state of charge, relevant temperatures, etc are displayed on a 16 x 2 line LCD. As each data packet comes in, the displayed data is updated. Due to the limitations of display area, two screens are flashed alternatively every 3 seconds to display the data in its entirety.

Further, two super bright LED's are used to warn the user of problems. A yellow LED comes on when the receiver has not received a valid data packet in the last 3 minutes indicating a communication problem with the base station. Once a valid packet comes through, the LED is turned off. A red LED and a magnetic buzzer are enabled when the alarm is triggered at the base station. To conserve power and to grab attention, the buzzer is cycled on/off so long as the alarm is triggered. The buzzer draws 35mA when active and an NPN amplifier is used as the driver since the MCU can only provide 20mA per pin. To disable the buzzer, the user must turn off the receiver. If the perturbation at the base station was temporary, the alarm will be reset after one minute and the receiver can be turned back on.

Power for the handheld receiver is provided by 4 AAA batteries in a series configuration. Using manufacturer specifications, the batteries will provide ~120 hours of operation. The current draw of major components is listed in Table 2 below.

Component	Power Draw(mA)
MCU @ 4Mhz	12
RLP434 Receiver	5
LED's + Buzzer (5%)	2.45
LCD	1.5
HT12D Decoder	0.2
	21.15

Table 2

When the batteries are new, they will provide an output voltage of 6.4 volts. All of the components except the receiver are rated to safely operate at this voltage. The maximum allowable voltage for the receiver is 6 volts. A 190 ohm resistor was put in series with the receiver to reduce its input voltage. The receiver was modeled as a 1,111 ohm resistor and 190 ohms in series will reduce the input voltage by 14%. At 6.5, the receiver will see 5.5 volts and at a power of 5 volts, the input to the receiver will be at 4.2 volts which is above its minimum operating voltage of 3.3 volts. Output from the receiver will be at a voltage lower then system Vcc but the Holtek decoder interprets signal voltages up to 30% lower then Vcc as "high".

A potentiometer is provided to adjust the LCD contrast. One issue that might become a problem is that the LCD contrast is very sensitive to supply voltage and contrast falls as the batteries are depleted. At 5 volts, the LCD is still readable but only in a well lit environment. The advantage of this is that the fall in contrast serves as a signal to the user to replace the batteries.

Photovoltaic Panel:

The selection of the PV panel took into consideration the input/output power requirements of the system as well as the available budget. In order to provide 4 hours of run time at a 50 watt draw, the panel would have to replenish the lost energy using 8 hours of daylight. Taking into consideration cloud shadowing, panel positioning, and inherent conversion losses, a 20 watt panel will optimally provide ~14 watts for 8 hours leading to a deficiency of 88 watts. The reserve capacity of the main battery is 325 watts however after a few days it would have to be taken from the field to a charging station. This is unsatisfactory and violates the requirement that the system be self-sustaining.

A 40 watt KC40 PV multicrystal panel from Kyocera was chosen. At a conservative output of 28 watts, the panel, over a period of 8 hours will provide 224 watts to the battery. This is sufficient to fully recharge the battery daily and make up for days of low solar irradiance. A conversion rating of 70% (40 -> 28) is conservative for design purposes. The main charge controller using the MPPT algorithm will keep the conversion ratio higher approaching 90-95%.

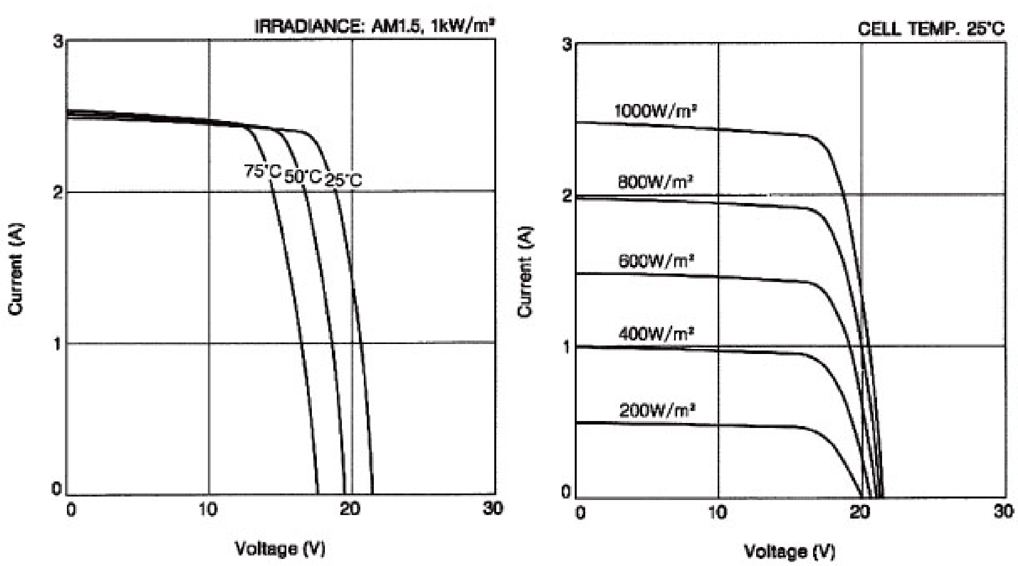


Figure 1: KC40 Electrical Characteristics

The PV panel is a simple electrical device and given any two of three variables, the third variable can be derived using the graphs in Figure 1. The three variables are panel temperature, panel voltage, and output current. The peak power current varies with solar isolation and thus using the panel output current in a maximal power point tracking (MPPT) algorithm would require sensing solar irradiance. This can be done, but would require mounting multiple sensors on the panel to accurately measure the irradiance onto the panel and would unnecessarily complicate the feedback controller.

The maximal power voltage, unlike the current, is constant regardless of irradiance. As shown on the right of Figure 1, the "knee" of each curve occurs at 16.9 volts. At that point, a rectangle drawn by extending a line from the point to either axes has the maximum area. The area is I * V which is power. The peak power voltage changes with panel temperature. The ambient temperature does not vary nearly as fast as solar ambience (clouds moving) and serves as a stable parameter for the controller.

Voltage Sensing:

Accurate sensing of battery voltage is necessary because it is the key indicator of the current battery state of charge. In order to protect the battery from overcharge or excessive discharge, the controller constantly monitors the battery voltage and toggles the load and panel relays accordingly. The battery voltage is sensed using a resistor divider network which consists of 15 and 7.5 Kohm 1% precision resistors. The voltage at the center tap of the divider is 1/3 the battery voltage and this is read by an A/D converter. The value of 1/3 was chosen because this provided the best resolution while ensuring that the input voltage to the A/D channel (5) would not exceed the MCU Vcc of 5 volts. From 12 to 15 volts, the sense voltage swing is one volt, for a divisor of four the range drops to 0.75 volts.

The sensing of the PV panel voltage is accomplished using a resistor divider network of 17.4 and 5 Kohm 1% precision resistors; reducing the input to 22%. The open circuit voltage of the PV panel is 21.5 volts this translates to a maximum A/D channel (4) input voltage of 4.8 volts.

Current Sensing:

Initially, current gauging was envisioned to track the battery status of charge (SOC). By accounting for losses in the battery, input current would be reduced to 80% and output current (load) would be subtracted from the running total. In testing however, this proved to be unreliable.

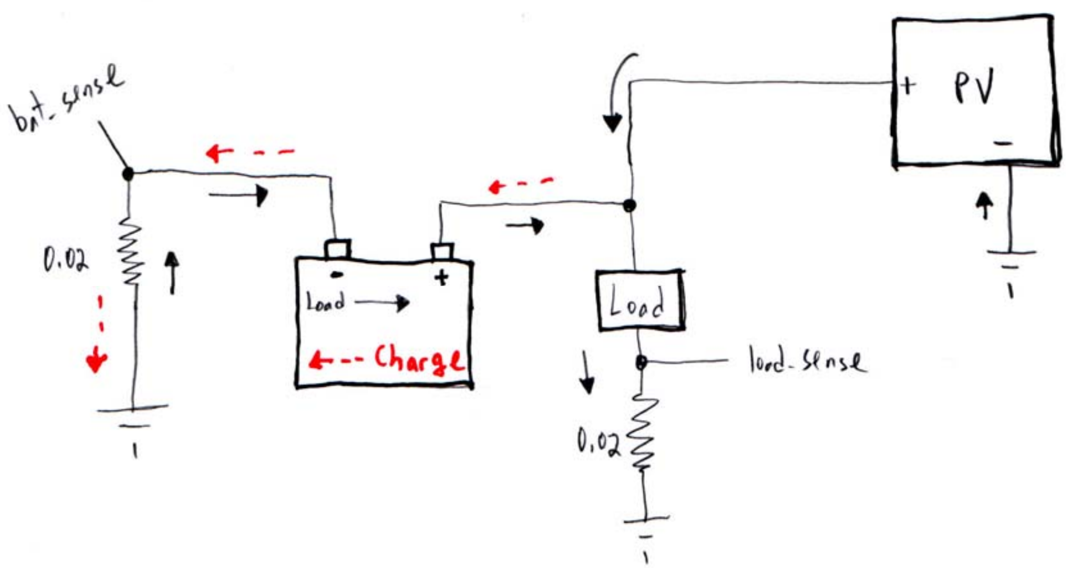


Figure 2: Current Flow Sensing

The original design utilized instrumentation amplifiers to sense current. The amplifiers did not work in the prototype for reasons still under investigation, and hence the design in Figure 2 was implemented for the final product. The current sense taps are sent to non-inverting Op-Amps with gains of 40-50 and the amplified voltages are sent to A/D channels 2 (Battery Sense) and 3 (Load Sense).

The load current is a combination of battery and PV currents and it is monitored precisely. The computation of a charging current going into the battery is a bit more complicated. The goal was to have a sense resistor placed at the PV positive output and to sense the voltage drop on the high side using an instrument amp. Since that didn't work, sensing had to be moved to the low side. Given an ambient temperature and a PV panel voltage, the current of the PV panel can be derived. Subtracting panel current from the load current we get the current being provided by the battery. During charging and for extended periods of time that the load is on standby, the load will be either disconnected or drawing very little current and charging current is measured at the battery sense tap. So long as charging current is greater then the load current, the battery sense tap will be at a positive voltage and the difference between the load current will yield the charging current.

Sense resistors of 0.02 ohms were chosen to minimize power loss. At a maximum load current of 5 amps, the voltage across the load sense resistor will be 100mv and 0.25 watts of power will be dissipated in the sense resistor. Because the sense voltages are indeed very small, care had to be taken in minimizing EMF noise. Most notably, errors emerged when the sense resistor was grounded too far away from the ground of its drop resistor in the non-inverting amplifier. The errors grew with increasing current because with high currents, the resistance of the copper traces began to play a role in creating different grounding potentials. With signals on the order of mV, small fluctuations in the ground potential are translated into large errors.

Load and Panel Control:

The connect/disconnect of input from the PV panel and the output to the load is accomplished using Omron G6C-1114P 5 volt relays. The Omron relays were chosen because they provided the best solution in terms of power consumption for the price ($5). The coil requires 30-40mA at 5 volts. This performance approaches much more expensive solid state relays ($20).

To drive the relays, a 2N4401 general purpose NPN amplifier was used. One end of the coil is connected to power and the transistor switches the other end to ground. A 1K resistor on the base limits the MCU pin output current. On system startup, the load relay is open and is closed once the controller ensures that the battery voltage is sufficiently high to support the load. The PV relay is closed so long as the battery voltage indicates charging is necessary. Both relays control power lines and a 5A fuse is in place before the load relay.

Temperature Sensing:

Two LM34 Fahrenheit temperature sensors were used to sense the battery and the ambient (PV Panel) temperatures. The importance of the PV panel temperature is documented above. Battery temperature affects the SOC calculation and to a smaller extent is used as an indicator of overcharging.

The output of the LM34 sensors is an analog voltage of 10mV/°F. This signal is sent to a non-inverting Op-Amp where it is boosted by a factor of four and sent to the A/D channels 6 (Battery) and 7 (Ambient). The ambient temperature sensor is currently mounted on the main solder board with the assumption that the ambient temperature doesn't vary greatly. Optimally, it would be glued to the back of the PV panel. The battery temperature sensor must be mounted on the battery to ensure fast and optimal detection of overcharge.

As the battery ages, its internal impedance grows and an increasingly larger percentage of charging power will be expended as heat.

Motion Sensor/Alarm:

The implementation of the security system is centered on the Analog Devices ADXL202 Dual Axis accelerometer. A mercury or electrolytic sensor could have been used in place of the accelerometer to detect movement of the device, but these would introduce a mounting issue, are position sensitive, and would not interface as well to the MCU. The ADXL provides both static (gravity) and dynamic (vibration) information which can be used to indicate the exact orientation of the main system (i.e. is the box tipped over) as well as whether the box is being moved and how aggressively it is being handled (a thief making a run for it). Currently absolute movement relative to initial positioning is used to trigger the alarm. The ADXL provides two forms of data output; duty-cycle and analog voltage. For this design, the analog voltage output is used and is connected to two A/D channels (0 and 1). On power-up, the MCU records the static acceleration values and thereafter continuously tests new acceleration data. If the difference between two measurements meets a certain user defined threshold, the alarm is triggered and sent wirelessly to the receiver. The alarm stays activated for one minute after which another test is made to determine if the main unit is still being moved.

Because a differential scheme is used to determine movement, a potential thief can theoretically move the device without triggering the alarm. This arises because of a tradeoff between alarm sensitivity and integrity. It would not be convenient to have gusts of wind set off the alarm and on the flip side; you don't want to set the threshold too high so that actual movement is not being detected. In the current design, it is highly unlikely that the device can be moved without triggering the alarm. More importantly, the thief in order to make a successful getaway would have to know that this "Achilles heel" exists and that's highly unlikely. Presently the threshold is set at 1/4 the acceleration due to gravity or 2.4 m/s^2 which in practice is equivalent to a slight bump. During movement, the orientation of the box with respect to the ground will change and so will the acceleration due to gravity varying from 1g when the box is upright to 0g when it's on its side.

RF Transmitter:

Relevant status information is transmitted wirelessly to a handheld receiver using a Laipac TLP434-0.5W transmitter. At 0.5 watts of output power, the transmitter is good for a range of 500 meters. Continuous output at 0.5W in the UHF band is not allowed by the FCC. However, provisions exist that do allow output at a reduced duty cycle ~5%. During testing, a 50 ohm resistor was used in series with the transmitter to limit supply voltage to 7 volts during transmission. At 7 volts, the output power drops to 200mW. This system was designed for use in a remote part of Africa where communication regulations are not as strict. To achieve maximum power and range, the limiting resistor can be reduced. The specs of the transmitter specify a maximum input voltage of 12 volts and thus a resistor of at least 20ohms is needed to limit the input voltage.

Because the transmitter operates at a voltage higher then Vcc of the MCU, interfacing circuitry was created to interface the two voltage levels. The output of the encoder is passed through two NPN inverters before being passed from the collector of the second stage to the data input pin of the transmitter. The encoder sends data at 3kHz and the inverters switch much faster so no problem exists. The data input pin of the transmitter does not draw any current and thus this solution is optimal.

The Holtek HT12E encoder is powered by the 5 volt rail and its data lines are set directly by the MCU. For more information on the wireless communication interface, please see the "Handheld Receiver" section above. Late in the design, it became necessary to allow the MCU to control the address lines of the encoder. With nine packets of data, four address pins were sufficient to provide each packet with a unique address. Unlike with the decoder on the receiver end, address pins could not be wired directly to the MCU. Initially, they were connected directly and the receiver stopped acknowledging wireless communication. After much time was spent debugging this problem, it was discovered that the address lines on the encoder, more specifically lines A7-A4 were periodically for a very brief time (2-3us) pulled low when they should have stayed high. This was not noticed when the address pins were connected directly to the power rail but became evident when interfaced to the MCU. This glitch caused the address portion of each packet to be corrupted and thus the receiver never (very sporadically) recognized the data. Once it was discovered that the encoder was drawing, for very brief time periods, more current then the MCU could provide, inverting amplifiers were used to provide the interface between the MCU output and encoder address pins. This solution involved adding more components to the final version of the solder board but was necessary after attempts to use capacitors failed. This is why the MCU output addresses to the encoder are inverted in Table 1.

Battery:

The choice of the battery was initially guided by safety and portability. The battery had to be a sealed cell to avoid acid spillage during transport. Sealed lead acid (SLA) absorption glass matt (AGM) batteries were considered. The requirement for supplying 200watts per day ruled out batteries below 16 amp-hours. Since a higher output PV panel was chosen, the battery had to be obtained cheaply to meet the budget constraint. The Cornell Hybrid Electric Vehicle (CUHEV) team had 26Ah SLA Hawker Genesis batteries in storage from their 2001 competition and they were looking to dispose of them. After some testing to gauge battery vitality, it was determined that these would be sufficient for use in this design. The batteries were restored utilizing the charge on top of a charge technique several times.[(1)]

The Hawker cell battery has many advantages over similar SLA batteries. It has a very low internal resistance (5 milliohms) leading to constant terminal voltages even under high rates of discharge. It has a very high acceptance current capacity allowing it to reach 90% SOC in less then one hour. The use of high purity lead-tin internal grid translates into lower corrosion rates and longer battery life; the battery can be stored for three years and brought into service with minimal conditioning. For these reasons and numerous others, the US Army has made the Hawker Genesis line of batteries their de facto standard.[(2)] This and other information regarding the Hawker batteries can be found in the application manual supplied by Hawker.[(3)]

Because of the problems inherent in current gauging, and because the Hawker cell battery exhibits very low voltage drop under load, the SOC can be best approximated by the terminal voltage as shown in the figure below.

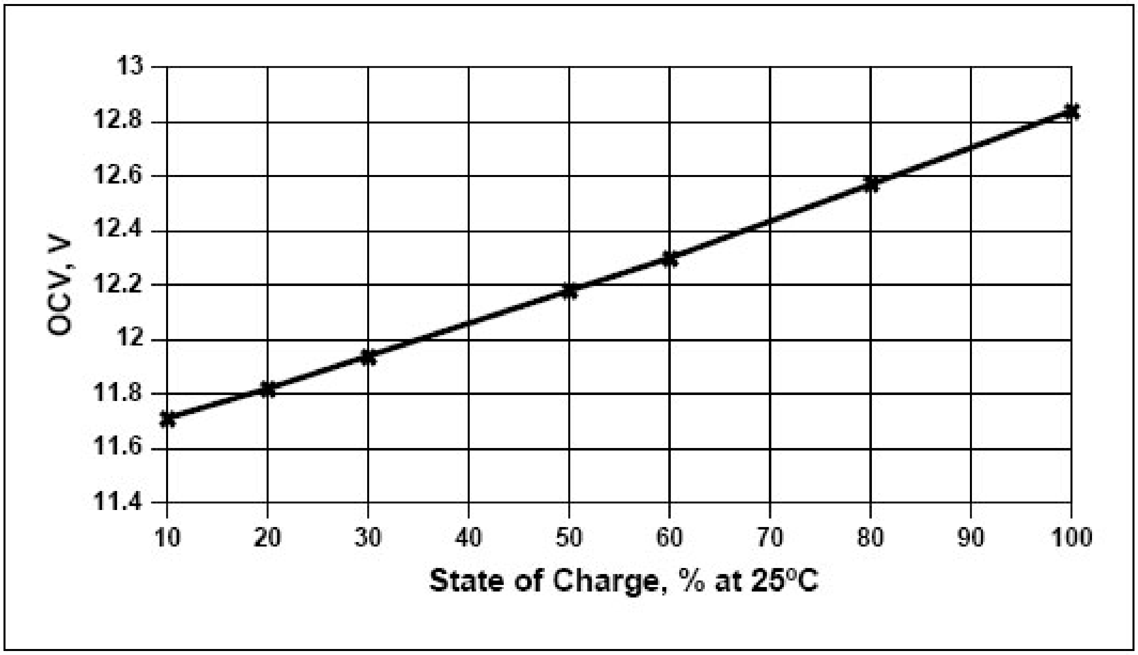


Figure 3: Open Circuit Voltage vs SOC

On startup, the controller, before connecting the load, would record the open circuit battery voltage and deduce an SOC based on the voltage. By constantly tracking the load current, once the load current falls below 0.5A, the controller periodically will disconnect the PV panel and measure the open circuit voltage to update the SOC.

Main Power Converter:

The PV panel can be connected directly to the battery and a charging current will flow. The magnitude of that current will be based solely on solar intensity. Worst case, and as a default, this is indeed what takes place in this design. However when the panel is connected directly to the battery, the panel voltage drops down to the battery voltage and power transfer from the panel is reduced and not optimized. Looking at Figure 1, the maximum power provided by the panel is at 16.9 volts where a current of 2.35 amps yields 40 watts. If the voltage is reduced to 12.5 volts, a current of 2.4 amps will still flow but the power drops to 30 watts. In an effort to maximize the power transferred to the battery, a variable buck converter was designed.

The initial goal was to build a buck converter from discrete components and control the duty cycle - which directly controls the voltage drop, using the Fast-PWM output of the MCU. A high frequency PWM allows for physically small inductors and capacitors and leads to reduced system costs. After much investigation, due to my lack of experience with analog design, and primarily due to the availability of dedicated adjustable switching converters, the choice was made to use a dedicated regulator as the center of the converter.

The National LM2679S-ADJ is an adjustable buck converter with a maximum current capacity of 5A. The output voltage is set using a resistor network at the feedback pin. A digital potentiometer from Maxim was selected as one of the resistors in the feedback network. The output voltage is defined as:

$$Vout = 1.21(1 + R2/R1) \qquad \text{Eq. 1}$$

The recommended value for R1 is 1K. The resolution of the MAX5436 digital pot is 50Kohm/128 = 400ohms. Using 1K for R1, a Vout of 12 requires 8,917 ohms for R2 and a Vout of 15 requires an R2 of 11,397 ohms. Thus the variable range for R2 is 2,479 ohms which translates to 6 valid settings of the MAX5436. By choosing an R1 of 2,500 ohms, the usable range is increased to15 valid settings. Giving an output resolution of 0.2 volts. The MAX5436 is interfaced using a 3-wire SPI link. Updates of resistance values occur seamlessly. If the current resistance is set to 20K, and a command comes in to change that to 10K, the resistance will fall from 20K instantly down to 10K without hitting 0 or 50K.

Most of the debugging time expended in this project to date was in getting the prototype of the main converter to work on a solder board. In the initial designs, the output voltage could be varied dynamically but under load it dropped to zero. This problem was later attributed to analog noise problems and varying potentials in the ground lines. Because a solder board was used, the traces were thin and wires were used to connect various grounding points. Through various iterations of the design, the distance between grounding points was reduced and the operation of the converter improved however it never reached acceptable performance. The maximum output current that could be provided without excessive output voltage drop (< 0.5V) was 2 amps. Recommendations from National included using wide and short ground connections between key devices as illustrated in the schematic below.

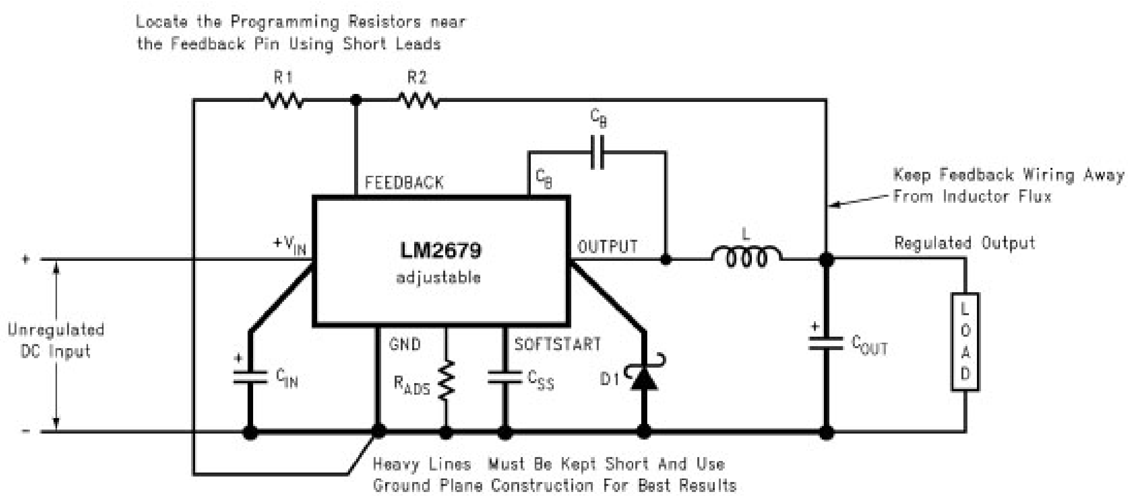


Figure 4: Recommended Wiring Diagram

Wide and short ground connections could only be created using a PCB and because I couldn't get the prototype to work, I was hesitant to start over using a PCB. The decision was made to order pre-built 5V evaluation modules from National and to modify them for use in the adjustable converter. The LM2679-5 IC was removed and replaced with the adjustable version. Leads were extended to connect the feedback pin to the digital potentiometer. In initial testing, the converter worked perfectly. A second evaluation module was ordered to provide a 5 volt supply for the main controller board. Since the evaluation board puts out a regulated 5 volts at up to 5 amps, the 5 volt line can be used to drive external equipment.

Currently however, a problem exists when the 5 volt converter is connected to the output of the main converter. It is believed that the input capacitance of the 5 volt converter adds to the output capacitance of the main converter and at startup, the delay in charging the output capacitance causes the main switcher to not turn on because it appears as if a short circuit exists at the output.

When the 5 volt converter is connected to just the battery, the output voltage is noisy causing the MCU to reset multiple times. Additional bulk capacitors at the 5 volt output solved the problem but because the main converter does not turn on with the 5 volt board connected, an LM340-T5 linear regulator is presently in place to supply 5 volt power to the controller board. To protect the sensitive analog signals, an aluminum sheet covered with electrical tape separates the power converters from the controller board above.

Main System Display:

The initial design did not call for an LCD at the main controller but it became necessary during testing and debugging. Since all the digital ports are being used for various peripherals, the only communication interface that was still available was the RS232 serial interface. A 16x2 serial LCD was available from previous years and was used to display system voltages and currents. The display runs off of 5 volts and allows for varying of the contrast and backlight using serial commands. For upgrades or reprogramming out of the field, a programming header exists to interface to the STK500 development platform and a serial jumper allows for RS232 communication with a PC.

Embedded Application:

The embedded application takes in various digital and analog inputs and processes them accordingly. The primary output is the SPI data to the digital potentiometer controlling the charging of the main battery. Secondary output is the data and address line toggling to the transmitter to send the data wirelessly.

The load compensated battery voltage is compared to 11.8 and once it falls below that, the load relay is turned off and the load is disabled to preserve the battery from excessive discharge. The load is reconnected at 12.3 volts. Likewise, the PV panel is disconnected once the battery voltage reaches 14 volts. It is reconnected at 12.7 volts. Intermittently, the PV panel is disconnected and the open circuit voltage of the panel is measured. So long as the PV panel voltage is below the battery voltage, the panel is disconnected from the main converter.

The main system loop monitors the PV voltage and adjusts the output voltage of the main converter to ensure that the PV voltage is held with +/- 1 volt of 16.9. If the PV voltage is greater then 18, the output voltage of the converter is raised increasing the voltage difference between the converter output and the battery and more current flows. The increased current flow causes the voltage on the panel to drop and if it drops below 15, the output of the main converter is reduced. Currently this is adjustment is being done once every 2 seconds. Further testing will reveal if stability can be maintained with faster update rates.

An SOC measurement is taken periodically so long as the load current is below 0.5 amps and the SOC data is updated by using a lookup table containing information from Figure 3. The ambient temperature measurement is taken periodically and the relevant float charging voltage will be compensated for in the final revision.

The data transmission is handled by a state machine that is called every 7 seconds. With nine packets of data, the state machine sends out the first packet every 63 seconds. On the transmitter end, the states are cycled through continuously. Before entering the state machine, the transmitter is turned on for 300ms. It is assumed that the relevant address and data lines were set by the previous state. Once the data is transmitted, the state machine is entered and the address - data lines for the next packet output are set accordingly. On the receiver end, the state transitions take place as valid data comes in. First the data is recorded into the appropriate variable given the current decoder address and the state machine is entered based on that current decoder address. The address is changed to prepare the reception of the next data packet and the state machine is exited. Entry back into the state machine is triggered by the valid data pin being set by the decoder. If a packet is missed, the decoder address is not changed and will only change once that packet has been received. With new transmissions taking place every minute, if a valid transmission doesn't take place within three minutes, a led is lit to indicate communication failure.

Cost Analysis:

The goal was to stay around $300 and using sampled/donated parts this requirement has been met. The bulk of the expense was for the PV panel and the initial budget was based on using a smaller 20 watt panel. Subsequent investigation revealed that a higher output panel would be required to meet the design constraints. The batteries, had they not been donated would have cost close to $100 each with shipping.

Part	Quant	Price	Total
40 Watt Kyocera PV Module	1	$220.00	$220.00
Main Power Converter (LM2679-Adj)	1	$20.00	$20.00
Omron G6C-1114P 5V Relay	2	$10.00	$20.00
Laipac TLP434-0.5W Transmitter	1	$25.00	$25.00
Laipac RLP434A Receiver	1	$10.00	$10.00
Main System Enclosure	1	$5.00	$5.00
Receiver Enclosure	1	$5.00	$5.00
Magnetic Buzzer	1	$2.50	$2.50
20 milliohm Sense Resistors	2	$0.80	$1.60
Fuse holder and 5A Fuse	1	$2.50	$2.50
Holtek HT12D Decoder	1	$1.50	$1.50
Holtek HT12E Encoder	1	$1.50	$1.50
Solder Board	3	$1.50	$4.50
Hawker Genesis G12V26Ah10EP	2	$0.00	$0.00
Analog Devices ADXL202	1	$0.00	$0.00
Maxim MAX5436	1	$0.00	$0.00
Atmel ATMEGA32	1	$0.00	$0.00
Atmel AT90S8515	1	$0.00	$0.00
LM34	1	$0.00	$0.00
2N4401	10	$0.00	$0.00
Generic 16x2 LCD	1	$0.00	$0.00
Serial 16x2 LCD	1	$0.00	$0.00
		Total	**$319.10**

Results:

The system has been tested sporadically since sunny days in Ithaca are hard to come by. Further it is difficult to determine the solar intensity incident on the panel and since this has not been quantitatively determined it is not possible to accurately determine the conversion efficiency. However, in testing, the controller was able to keep the PV panel voltage at 15-20. This range specifies an improvement over having the panel connected directly to the battery.

Initially the large range (5 volts) of the PV panel voltage fluctuation caused concern because the anticipation was that the controller would be able to keep the voltage about 16.9 with little fluctuation. The cause of the large voltage swing was discovered to be insufficient resolution in the digital potentiometer and the initial blessing of the low internal impedance of the Hawker battery only exacerbated this problem.

Charging Current = (Charging Voltage – Battery Voltage)/Battery Internal Impedance

In order to finely control the charging current and in turn keep the PV panel voltage at its maximum power peak, the charging voltage must be controlled at a resolution of 8.8mV. For example, to provide the battery with 2.5A given a battery voltage of 13 and an initial internal impedance of 0.004 ohms, the charging voltage needs to be set to 13.1 volts. 10mv is below the 0.2 volts being provided by the digital potentiometer. So since it can't reach the maximal power ideal, it switches back and forth about that ideal and thus the large swing in panel voltage. The small internal impedance means that ever smaller differences in charging and battery voltages are responsible for increasing charging currents.

The relevant information is being accurately transmitted to the handheld unit. Ranges of 50 meters have been tested and work fine. Perturbations of the main controller are being sensed and the alarm is triggered at the handheld unit. Further the load connects and disconnects take place at the specified battery voltages. To improve robustness by reducing the false switching of the load, modeling of the voltage drop while under load are being carried out. Typically under a 4 A load the terminal voltage drops by 0.25 volts. In the loaded case, the load should be cutoff at a terminal voltage of 11.6 volts. With the load removed the OCV will return to 11.85 indicating an SOC of 20%. To improve battery longevity, discharges below 11.85 OCV are not allowed.

Current consumption in an unloaded case without the LCD is 40mA at 12 volts. During transmission, the current jumps to 80-100mA. With the LCD and its backlight sans the transmitter, terminal current is at 200mA. The LCD backlight is a huge power draw. These measurements were taken using the 5 volt linear regulator to power the main controller. Essentially a voltage drop of 7 at the above mentioned currents takes place at the regulator leading to a power loss of nearly 60%. It is imperative that the problem with the 5 volt switching regulator is fixed. The switching regulator is 90% efficient and would cut the total system power down to 100-125mA with everything enabled. Currently, with the linear regulator, run time without the sun given the 26Ah battery is over 600 hours or 27 days. Even with the battery down to 20% capacity, which is where the controller disconnects the load, system run time is 5 days. It is highly unlikely that the system will not see the sun for 5 days.

All that remains for the main system is packaging. The packaging material has been obtained but the final packaging has been delayed due to testing.

Conclusion:

In order to improve the operation of the MPPT component of this design, the resolution of the digital potentiometer must be improved. As an alternative, a digitally controlled current source can be installed between the main converter output and the positive terminal of the battery. A digitally controlled current source would limit the acceptance current and ensure that the main converter output voltage does not drop excessively. This in turn would lead to a stabilization of the PV voltage.

In conclusion, a smart charge controller has been designed, built and tested. Testing shows that indeed the battery is being protected from over-charging as the controller cuts the PV panel power when the SOC reaches 95-100% and is reconnected intermittently to supply a float charge keeping the battery topped off. Over discharge protection is being provided and the controller disconnects the load when the SOC falls below 20%. The MPPT algorithm provides a net benefit with increased efficiency of power conversion in spite of the insufficient resolution of the digital potentiometer. The sensitivity of the security system has been tuned to ensure that the alarm is dependable and robust to false triggers. A handheld receiver allows for remote security via a wireless link. Additional functionality has been added allowing the operator to monitor critical operating parameters of the system remotely using the handheld receiver. The original power requirement of 200 watt-hours per day has been met and selected components allow for a flexible supply of power not exceeding 300 watt-hours for a single day. Further, the system supports PV panels of up to 100 watt maximal output without replacing any major components.

Further improvements of the design would include photo-cells to sense the position of the sun with respect to the panel and integrated motors to position the panel for optimal solar energy absorption daily and year round.

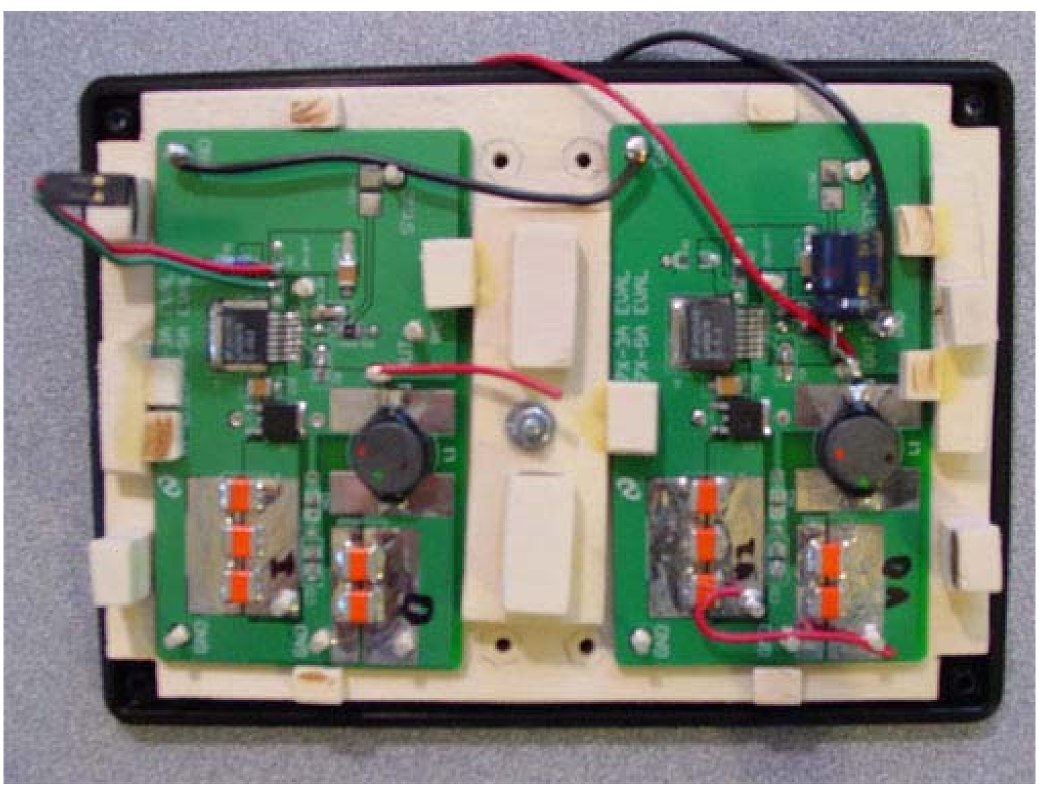
Instructions for Use:

The system is "plug and play" with terminals to connect the various components. A fuse is in place limiting load current to 5 amps. For prolonged storage, it is recommended that the battery is disconnected. To prolong battery life, never store the battery with a low SOC or OCV less then 12.2. If storing for periods in excess of six months, remove the AAA from the receiver.

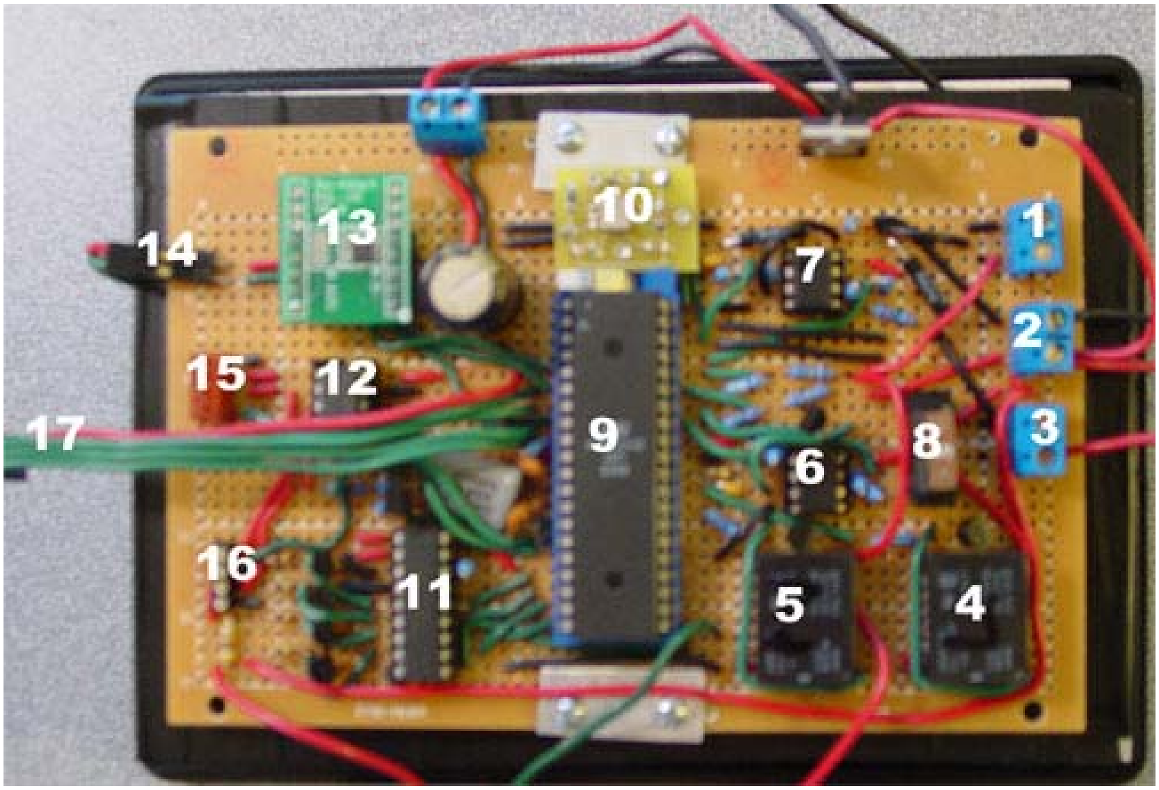
References:

1) http://www.batteryuniversity.com/parttwo-35.htm
2) http://www.enersysinc.com/defense/vehicles.asp
3) http://www.enersysreservepower.com/documents/US_GPL_SG_001_0303.pdf

Photos:



Switching Power Converters. The board on the left with the jumper is the main converter. The jumper connects to digital potentiometer on the main board. The board on the right is the 5 volt converter. Note the severed supply line.



Main Controller Board: The antenna on the transmitter does not go into the socket. When the transmitter is in place, the antenna is the pin nearest to the bottom of the picture above in 16. The MCU is in a ZIF socket and is released using the lever near the accelerometer.

1	PV Panel Terminal (Top - Ground)
2	Battery Terminal (Top - Ground)
3	Load Terminal (Top Ground)
4	Load Relay
5	PV Relay
6	Temp Sense Op-Amp
7	Current Sense Op-Amp
8	5A Load Fuse
9	ATMega32 MCU

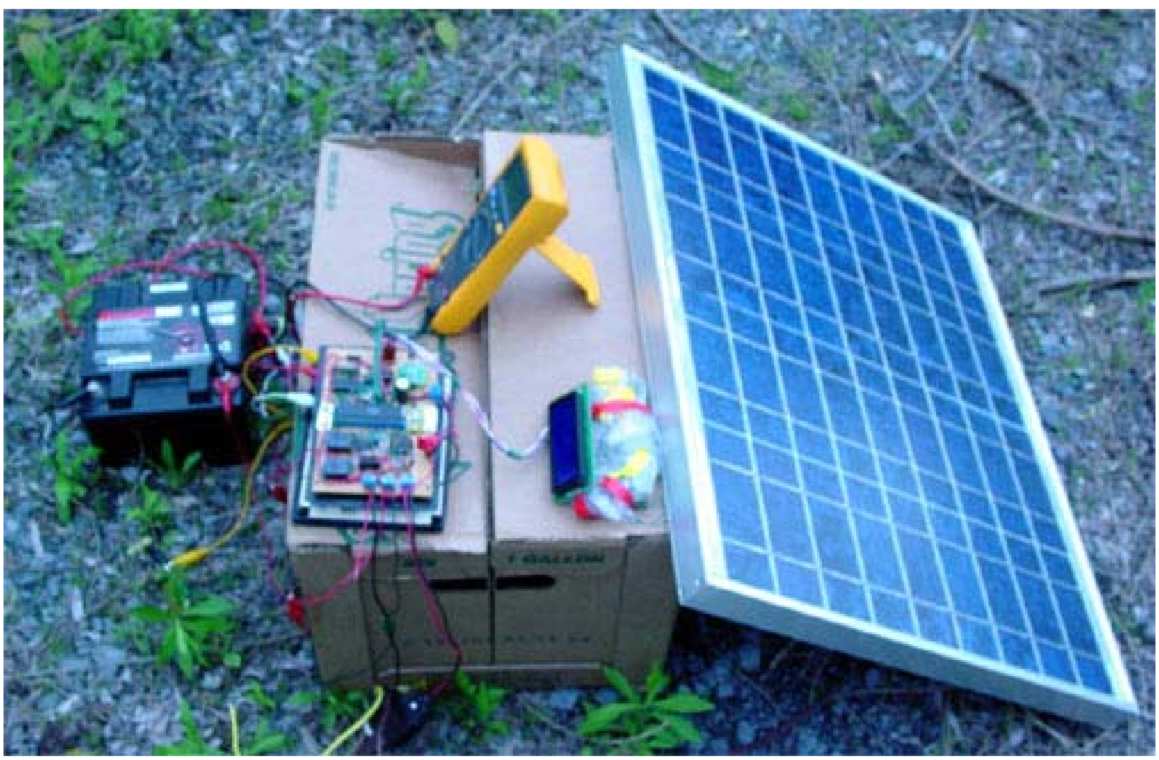
10	Accelerometer
11	Encoder
12	Driver for TX E/D
13	Maxim Digital Pot
14	Connector to Main Converter
15	LCD Connector
16	TX Socket, !Antenna at bottom!
17	Programming Header

The Receiver.

In operation as can be seen to the left, the battery voltage, SOC, ambient temperature, and battery temperatures are updated every minute. The plus sign indicates that the battery is being charged, a minus corresponds to a discharge greater then 0.5 A since the battery is always discharging due to the fact that it power the control circuitry. The red led and the buzzer come on to indicate that the alarm was triggered.

Load current is being monitored and is within 0.03 amps of the actual current draw. Since it's not being used as a sensitive control input, errors in the current measurement are tolerable.

The picture below is the system in operation. The battery and the main controller will be packaged in a toolbox. The panel has dimensions of 26"x21".



Receiver Code:

```
#include <90s8515.h>
#include <stdio.h>
#include <lcd.h>

//timeout values for each task
#define t1 50
#define t2 100
#define timeout_time 36000    // 3 mins at 5ms tick

#define LCDwidth 16 //characters

#asm
    .equ __lcd_port = 0x12
#endasm


// the task subroutines
void task1(void);
void task2(void);
void initialize(void);                  // all the usual mcu stuff

void new_rx(void);

char lcd_buffer[17];                    // LCD display buffer

unsigned int lcd_time;
unsigned int timeout;                   // no valid rx in timeout_time

unsigned char time1;
unsigned char time2;
unsigned int timer1;                    // max rx time plus some (1.2 sec)
unsigned int timer2;

unsigned char rx_error;                 // success of last rx
unsigned char rx_index;                 // state information

unsigned char voltage, soc, out_temp, bat_temp;

unsigned char alarm;                    // 0 - Disarmed
                                        // 1 - Activated
                                        // 2 - Armed

unsigned char charging;                 // 0 - Nothing
                                        // 1 - Charging
                                        // 2 - Discharging
                                        // 3 - Charging and Discharging

//**********************************************************
//timer 1 compare ISR
interrupt [TIM1_COMPA] void cmpA_overflow(void)
{
       if (time1 > 0)
              time1--;
       if (time2 > 0)
              time2 = time2 - 1;
       if (timer1 > 0)
              timer1--;
       if (timer2 > 0)
              timer2--;
       //if ((rx_error == 1) && (timeout <= timeout_time))
       //     timeout++;
       //if (rx_error == 0)
       //     timeout = 0;
       if (timeout <= timeout_time)
              timeout++;

       lcd_time++;                     // Flash alternate screens
}
```

```
//********************************************************
//Entry point and task scheduler loop
void main(void)
{
    initialize();

    lcd_clear();

    //main task scheduler loop
    while(1)
    {
            lcd_gotoxy(0, 0);
            if (lcd_time < 600)
            {
                if (charging == 1)
                        sprintf(lcd_buffer, "Voltage %2d.%1d  + ", voltage >> 4, voltage &
0x0f);
                else if (charging == 2)
                        sprintf(lcd_buffer, "Voltage %2d.%1d   -", voltage >> 4, voltage &
0x0f);
                else if (charging == 3)
                        sprintf(lcd_buffer, "Voltage %2d.%1d  +-", voltage >> 4, voltage &
0x0f);
                else
                        sprintf(lcd_buffer, "Voltage %2d.%1d    ", voltage >> 4, voltage &
0x0f);
                lcd_puts(lcd_buffer);

                lcd_gotoxy(0, 1);
                if (alarm == 2)
                        sprintf(lcd_buffer, "Soc %2d%1d%%   Armed", soc >> 4, soc & 0x0f);
                else
                        sprintf(lcd_buffer, "Soc %2d%1d%%        ", soc >> 4, soc & 0x0f);
                lcd_puts(lcd_buffer);
            }
            else if (lcd_time < 1200)
            {
                sprintf(lcd_buffer, "Outside Tmp %2d%1dF", out_temp >> 4, out_temp &
0x0f);
                lcd_puts(lcd_buffer);

                lcd_gotoxy(0, 1);
                sprintf(lcd_buffer, "Battery Tmp %2d%1dF", bat_temp >> 4, bat_temp &
0x0f);
                lcd_puts(lcd_buffer);

            }
            else
            {
                lcd_time = 0;
            }

            if (PINC.7 == 1)
            {
                new_rx();
            }

            if ((alarm == 1) && (time1 == 0))
                task1();

            if ((timeout >= timeout_time) && (time2 == 0))
                task2();
    }
}

//********************************************************
void task1(void)
{
       time1 = t1;
```

```
        PORTB.0 = ~PORTB.0;           // Buzzer

        PORTB.1 = ~PORTB.1;           // RED LED
}

//*******************************************************
void task2(void)
{
        time2 = t2;

        PORTB.2 = ~PORTB.2;           // Yellow LED
}

//*******************************************************
void new_rx(void)
{
        unsigned char input;
        unsigned char www;

        timeout = 0;                          // timeout incremented in interrupt
                                              // checked for timeout condition in main


        input = PINC & 0x0f;                  // Read the input
        while(PINC.7 == 1)                    // Wait until receive pin is cleared
        {
                www++;
        }

        switch (rx_index)
        {
                case 0: // Init

         break;
        case 1: // Reserved

         break;
        case 2:         // Voltage High
                voltage = input << 4;
                rx_index = 3;
                PORTA = 0x03;                 // Set Receive Address of Voltage Low
         break;
        case 3: // Voltage Low
                voltage = voltage | input;
                rx_index = 4;
                PORTA = 0x04;
         break;
        case 4: // Soc High
                soc = input << 4;
                rx_index = 5;
                PORTA = 0x05;
         break;
        case 5: // Soc Low
                soc = soc | input;
                rx_index = 6;
                PORTA = 0x06;
         break;
        case 6: // Out Temp H
             out_temp = input << 4;
                rx_index = 7;
                PORTA = 0x07;
         break;
        case 7: // Out Temp L
             out_temp = out_temp | input;
                rx_index = 8;
                PORTA = 0x08;
         break;
        case 8: // Bat Temp H
             bat_temp = input << 4;
                rx_index = 9;
                PORTA = 0x09;
```

```
        break;
        case 9: // Bat Temp L
            bat_temp = bat_temp | input;
               rx_index = 0x0a;
               PORTA = 0x0a;
        break;
        case 0x0a:     // Charging/Alarm
            charging = input >> 2;
            alarm = input & 0x03;
            rx_index = 2;
            PORTA = 0x02;
        break;
        case 11:

        break;
        case 12:

        break;
        case 13:

        break;
        case 14:

        break;
        case 15:

        break;
        }
        timer1 = 100;                 // Delay for decoder transition 0.5 sec
        while(timer1 > 0)
        {
               www++;
        }
}

//***********************************************************
//Set it all up
void initialize(void)
{
        lcd_init(LCDwidth);     // initialize the display
    // Port D is LCD

    //set up the ports
    DDRB = 0xff;              // PORT B is an output
       // B.0 = Buzzer                      Active High
       // B.1 = Red LED         Active High
       // B.2 = Orange LED      Active High

    DDRC = 0x00;              // PORT C is an input
    PORTC = 0x00;             //     Hi-Z inputs
       // C.3 - C.0 = Din
       // C.7 = Incoming Data

    PORTB = 0x00;

    DDRA = 0xff;              // PORT A is an output
       // A.3 - A.0   Address Control

    PORTA = 0x02;             // initial value
    rx_index = 2;             // initial value

    // Set up timer 1 to tick at 5ms
       TIMSK = 0x40;          // turn on timer 1 compare match interrupt
       TCCR1A = 0x00;
       TCCR1B = 0x0a;         // sets prescale to 8, clear on compare match
       TCNT1 = 0;             // and zero the timer
       OCR1A = 2500;          // 0.25us * 8 * 2500 = 5ms sec

       time1 = t1;            // alarm time
       time2 = t2;            // rx timeout time
```

```
        rx_error = 1;
        timer1 = 0;
        timer2 = 0;

    //crank up the ISRs
    #asm
        sei
    #endasm
}
```

Main Controller Code:

```
/*      Title:          solar_main.c
        Date:           5/11/2004
        Description:    Secure Solar Power Source - main file
        Note:           Timers assume 16 Mhz crystal
*/

/************************* #INCLUDE ******************************************/
#include <Mega32.h>
#include <spi.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>

/************************* #DEFINE *******************************************/
// Task timeout values in multiples of 1ms
#define tx_const 7000
#define buck_const 2000
#define lcd_const 500

// Task timeout values in multiples of 1 sec
#define soc_const 60
#define pv_const 300


#define Aref 5.06
#define bv_const (5.06 / (255 * 0.3))
#define pv_const (5.06 / (255 * 0.22))

#define lc_const (5.06 / (255 * 71 * 0.02))
#define cc_const (5.06 / (255 * 40 * 0.02))

#define out_const ((5.06 * 100) / (255 * 4))

/************************* FUNCTION PROTOTYPES********************************/
void check_butt(void);

void initialize(void); // Set everything up here first

unsigned char read_analog(unsigned char channel);
void set_voltage(float voltage);      // adjust the buck converter

void send_tx(void);
void lcd_show(void);
void check_buck(void);
void check_soc(void);
void check_pv(void);

/************************* GLOBAL VARIABLES **********************************/
unsigned int lcd_time;
unsigned int tx_time;
unsigned int buck_time;

unsigned int temp_timer;
unsigned char tx_index;
unsigned char soc_time;
unsigned char pv_time;
```

```
unsigned char last_accelx, last_accely;

float bat_voltage;
float pv_voltage;
float v_adjust;
unsigned char soc;
unsigned char bat_temp;
unsigned char out_temp;

unsigned int alarm_time;
unsigned char alarm;                                    // 0 - Disarmed
                                                        // 1 - Activated
                                                        // 2 - Armed

unsigned char charging;                                 // 0 - Nothing
                                                        // 1 - Charging
                                                        // 2 - Discharging
                                                        // 3 - Charging and Discharging

float charge_current;
float load_current;

float buck_voltage;

float holder;


//**********************************************************
// timer 0 compare ISR              ticks at 1 ms
interrupt [TIM0_COMP] void timer0_compare(void)
{
       // Decrement the times if they are not already zero
       if (tx_time > 0)
              tx_time = tx_time - 1;
       if (buck_time > 0)
              buck_time = buck_time - 1;
       if (temp_timer > 0)
              temp_timer = temp_timer - 1;
       if (lcd_time > 0)
              lcd_time = lcd_time - 1;

}

//**********************************************************
// timer 1 compare A ISR              ticks at 1 s
interrupt [TIM1_COMPA] void timer1_comparea(void)
{
       if (alarm_time > 0)
              alarm_time = alarm_time - 1;
       if (soc_time > 0)
              soc_time = soc_time - 1;
        if (pv_time > 0)
              pv_time = pv_time - 1;
}

//**********************************************************
void main(void)
{
       unsigned char temp;
       char diff;
       float adj_bat_v;

       initialize();


       // Delay for 2 seconds to init LCD
       temp_timer = 2000;
       while(temp_timer > 0)
    {
       temp = temp + 2;
```

```
    }
        putchar(12);           // Clear LCD Screen
        putchar(4);                    // Hide Cursor
        putchar(15);           // Set Contrast
        putchar(50);           // to 50%
        putchar(14);           // Set Backlight
        putchar(30);           // to 30%

//      PORTC.7 = 1;                   // Load Connect
//      PORTC.6 = 1;                   // PV Panel

        last_accelx = read_analog(0x00);
        last_accely = read_analog(0x01);

        // Read Battery Voltage
        temp = read_analog(0x05);              // 4.91 / (255 * 0.3) = 0.0642
    bat_voltage = temp * bv_const;



        v_adjust = 0.22;                                            // Default to .22
charge
        buck_voltage = bat_voltage + v_adjust;
        //set_voltage(buck_voltage);                    // Set charging voltage
        holder = 71;
        set_voltage(holder);                    // Set charging voltage (13.5 initially)

//      PORTC.6 = 1;                                                // Connect PV power

//      PORTC.7 = 1;                                                // Connect Load

        //main task scheduler loop
        while(1)
        {
                temp = read_analog(0x02);             // 4.91 / (255 * 71 * 0.02) = 0.0136
        charge_current = temp * cc_const;

            temp = read_analog(0x03);         // 4.91 / (255 * 40 * 0.02) = 0.0241
        load_current = temp * lc_const;

        temp = read_analog(0x04);              // 4.91 / (255 * .22) = 0.0875
        pv_voltage = temp * pv_const;

        temp = read_analog(0x05);              // 4.91 / (255 * 0.3) = 0.0642
        bat_voltage = temp * bv_const;

        temp = read_analog(0x07);
        out_temp = (unsigned char)(temp * out_const);

            if (buck_time == 0)
                check_buck();

            temp = read_analog(0x00);
            if (alarm_time == 0)
        {
                alarm = 0;
                diff = cabs(temp - last_accelx);
                if (diff >= 5)
                {
                        alarm_time = 60;
                        alarm = 1;
                }
        }
        else
        {
                last_accelx = temp;
        }

        temp = read_analog(0x01);
        if (alarm_time == 0)
        {
```

```
                alarm = 0;
                diff = cabs(temp - last_accely);
                if (diff >= 5)
                {
                        alarm_time = 60;
                        alarm = 1;
                }
        }
        else
        {
                last_accely = temp;
        }


                if (lcd_time == 0)
                lcd_show();

        if (soc_time == 0)
                check_soc();

        if (pv_time == 0)
                check_pv();

        //if (cb_time == 0)
        //      check_batt();
                if (tx_time == 0)
        {
                //check_butt();
                PORTD.2 = 0;            // Enable TX
                send_tx();
                PORTD.2 = 1;            // Disable TX
        }

        // Load Current Adjusted OCV Battery Voltage
        // terminal V drop = 0.1297 * load_current - 0.0211
        adj_bat_v = 0.1297 * load_current - 0.0211;
        adj_bat_v = bat_voltage + adj_bat_v;

        if (adj_bat_v < 11.8)
                PORTC.7 = 0;
        else if (adj_bat_v > 12.3)
                PORTC.7 = 1;

        if (bat_voltage > 13.7)
                PORTC.6 = 0;
        else if (bat_voltage < 12.7)
                PORTC.6 = 1;

    }
}

//**********************************************************
void lcd_show(void)
{
        lcd_time = lcd_const;                   // reset the task timer

    putchar(17);        // Cursor to 0,0
        putchar(0);
        putchar(0);
        printf("BV:%2.1f", bat_voltage);

    putchar(17);        // Cursor to 8,0
    putchar(8);
    putchar(0);
    printf("LC:%1.2f", load_current);

        putchar(17);    // Cursor to 0,1
        putchar(0);
        putchar(1);
        printf("PV:%2.1f", pv_voltage);
```

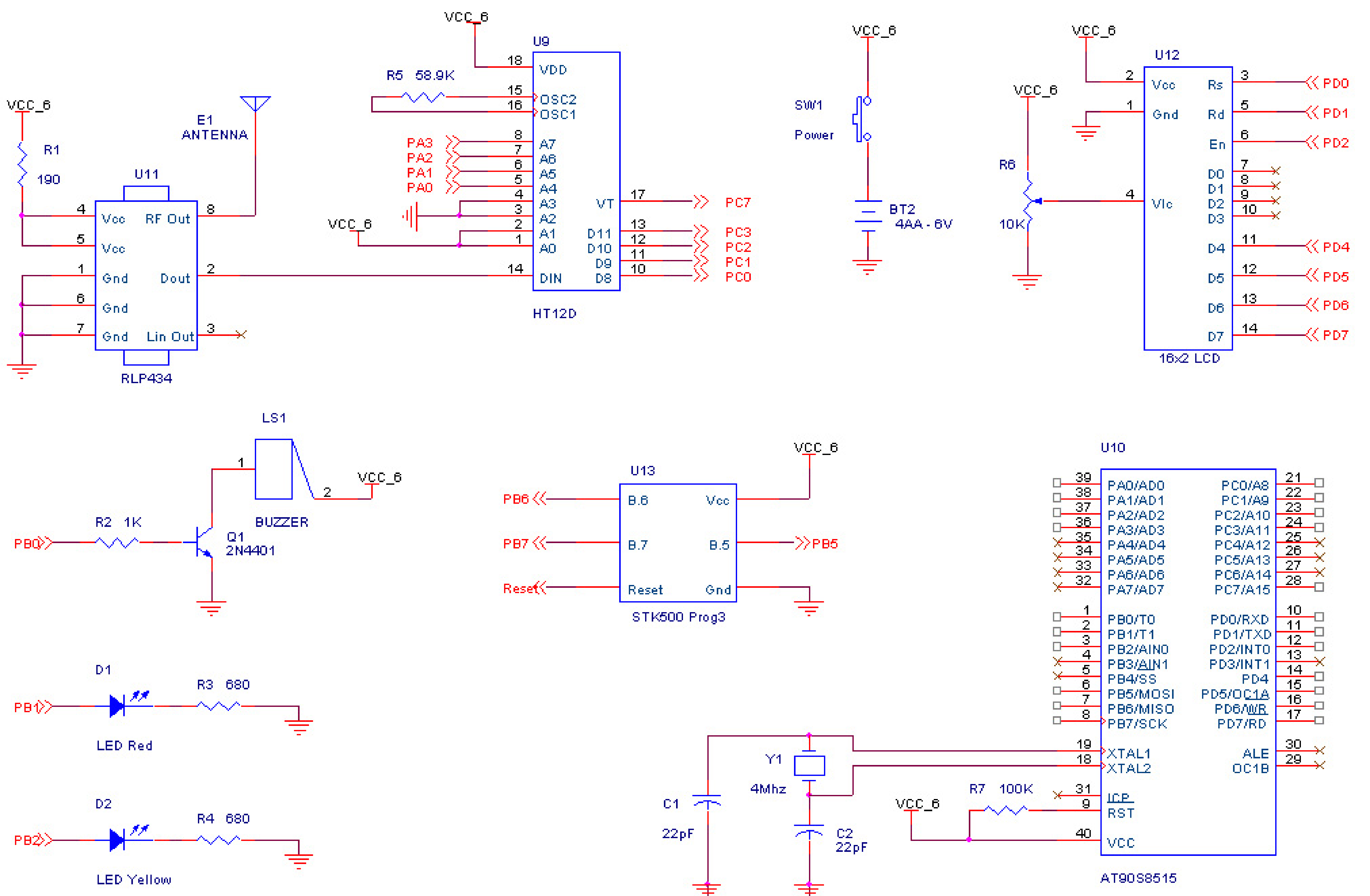
```
        putchar(17);   // Cursor to 8,1
        putchar(8);
        putchar(1);
        printf("CC:%1.2f", charge_current);
}

//*************************************************************
void check_butt(void)
{
   //   butt_time = butt_const;              // reset the task timer


}

//*************************************************************
void check_soc(void)
{
        float temp;

        soc_time = soc_const;                // reset the task timer

        // SOC = 78.3 * bat_voltage - 905.7
        if (load_current < 0.5)
        {
                temp = 78.3 * bat_voltage - 905.7;
                soc = (unsigned char)temp;
        }
}

//*************************************************************
void check_pv(void)
{
        unsigned char temp;
        float pv;

        pv_time = pv_const;           // reset the task timer

        PORTC.6 = 0;           // cut PV supply
        temp_timer = 500;      // delay for 500 ms
        while (temp_timer)
        {
                temp = temp + 3;
        }
        temp = read_analog(0x04);            // 4.91 / (255 * .22) = 0.0875
    pv = temp * pv_const;

    if (pv >= 13)
        PORTC.6 = 1;

}

//*************************************************************
void check_buck(void)
{
        buck_time = buck_const;              // reset the task timer

        if (pv_voltage < 16)
        {
                //v_adjust = v_adjust - 0.2;
                //PORTC.6 = 0;
                //set_voltage(bat_voltage + v_adjust);
                holder = holder + 1;
                set_voltage(holder);
                //PORTC.6 = 1;
        }
        else if (pv_voltage > 18)
        {
                //v_adjust = v_adjust + 0.2;
                holder = holder - 1;
                set_voltage(holder);
                //PORTC.6 = 0;
```

```
                //set_voltage(bat_voltage + v_adjust);
                //PORTC.6 = 1;
        }
}

//*********************************************************
unsigned char read_analog(unsigned char channel)
{
        // Select the channel
        ADMUX = (ADMUX & 0xf8) | channel;

        // Assume that the ADC has been turned on already
        // Start reading
        ADCSRA.6 = 1;

        while(ADCSRA.6 == 1)
        {}

        return ADCH;
}

//*********************************************************
void set_voltage(float voltage)
{
        // Vout = 1.21 * (1 + R2/R1); where R1 = 2500 ohms
        // Vout = 1.21 + 0.000484 * R2
        // R2 = 2066 * Vout - 2500
        // R2 = 53310 - 406 * SPI             SPI: 0 (53.31K) to 127 (1.795K)
        // SPI = 137.46 - 5.089 * Vout

        // SPI resolution ~0.2 volts


        float temp;

        temp = voltage;
   /*
        if ((voltage >= 12) && (voltage <= 15))
        {
        temp = 5.089 * voltage;
        temp = 137.46 - temp;
        PORTB.4 = 0;          // Pulls /CS low
        spi((unsigned char)temp);
        PORTB.4 = 1;          // Pulls /CS high
        }
        */
        PORTB.4 = 0;          // Pulls /CS low
    spi((unsigned char)temp);
    PORTB.4 = 1;              // Pulls /CS high
}

//*********************************************************
void send_tx(void)
{
        unsigned char www, t_voltage;
        float temp;

        tx_time = tx_const;           // reset the task timer

        if ((tx_index == 0x0d) || (tx_index == 0x0c))
        {
                t_voltage = ((unsigned char)bat_voltage) << 4;
                temp = bat_voltage - floor(bat_voltage);
                temp = temp * 10;
                t_voltage = t_voltage | ((unsigned char)temp & 0x0f);
        }

        PORTD.3 = 0;                          // initiate transmission
        temp_timer = 500;
        while(temp_timer)                     // transmitt for 300ms
        {
```

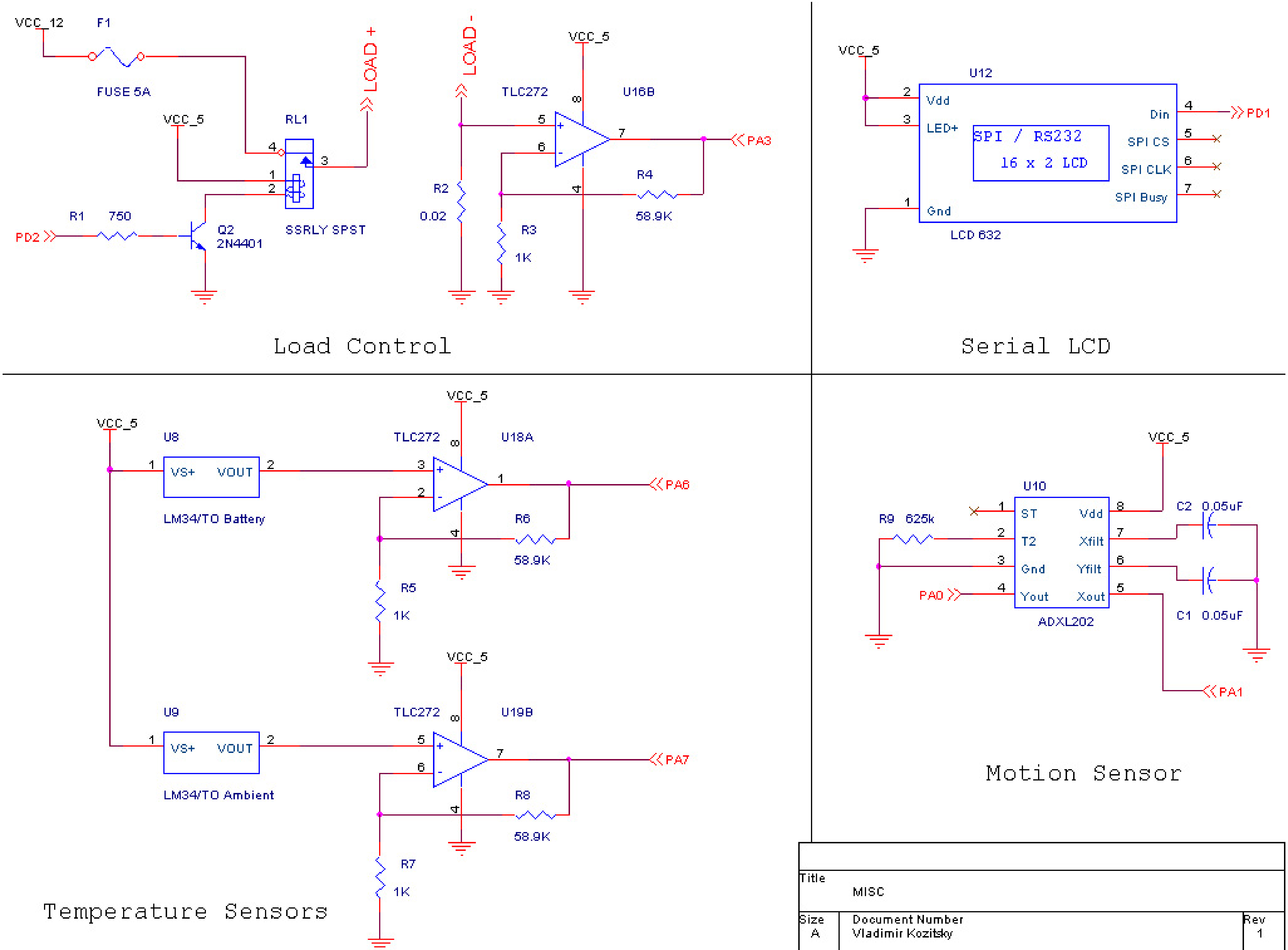
```
        www++;
}
PORTD.3 = 1;

switch (tx_index)
{
        case 0x00:

 break;
 case 0x01:

 break;
 case 0x02:

 break;
 case 0x03:

 break;
case 0x04:

 break;
case 0x05:     // Charging/Alarm
     PORTD = (charging << 6) | (alarm << 4) | 0x0c;
     printf("Char/Alarm: %3d\n\r", PORTD);
     PORTC = 0x05;
     tx_index = 0x0d;
 break;
case 0x06:     // Bat Temp L
     PORTD = ((bat_temp % 10) << 4) | 0x0c;
     //printf("Bat Low: %3d\n\r", PORTD);
     PORTC = 0x06;
     tx_index = 0x05;
 break;
case 0x07:     // Bat Temp H
     PORTD = ((bat_temp / 10) << 4) | 0x0c;
               //printf("Bat High: %3d\n\r", PORTD);
     PORTC = 0x07;
     tx_index = 0x06;
 break;
case 0x08:     // Out Temp L
     PORTD = ((out_temp % 10) << 4) | 0x0c;
     //printf("Out Low: %3d\n\r", PORTD);
     PORTC = 0x08;
     tx_index = 0x07;
 break;
 case 0x09:    // Out Temp H
     PORTD = ((out_temp / 10) << 4) | 0x0c;
               //printf("Out High: %3d\n\r", PORTD);
     PORTC = 0x09;
     tx_index = 0x08;
 break;
 case 0x0a:                                  // Soc Low
               PORTD = ((soc % 10) << 4) | 0x0c;
     //printf("SOC Low: %3d\n\r", PORTD);
     PORTC = 0x0a;
     tx_index = 0x09;
 break;
 case 0x0b:                                  // Soc High
               PORTD = ((soc / 10) << 4) | 0x0c;
               //printf("SOC High: %3d\n\r", PORTD);
     PORTC = 0x0b;
     tx_index = 0x0a;
 break;
 case 0x0c:                                  // Voltage Low
           PORTD = (t_voltage << 4) | 0x0c;
     //printf("Vol Low: %3d\n\r", PORTD);
     PORTC = 0x0c;                  // set addr of data being transmitted
     tx_index = 0x0b;         // next
     //tx_index = 0x0d;
 break;
case 0x0d:                                   // Voltage High
```

```
                PORTD = (t_voltage & 0xf0) | 0x0c;
             //printf("Vol High: %3d\n\r", PORTD);
             PORTC = 0x0d;                    // set addr of data being transmitted
             tx_index = 0x0c;         // next
         break;
        case 0x0e:                                 // Reserved

         break;
        case 0x0f:                                 // Init

         break;
        }
}

//************************************************************
void initialize(void)
{
  DDRD = 0xff;        // PORT D is an output
        // D.7 - D.4   TX Data Out
        // D.3                TX Enable
  PORTD = 0x0c;

  DDRC = 0xff;        // PORT C is an output
        // C.7                PV Voltage Enable
        // C.6                Load Enable
        // C.3 - C.0   Address Control
  PORTC = 0x0f;

  tx_index = 0x0d;     // initial value

  // Initialize PORTB
  // PB.0 output
  // PB.1 output
  // PB.2 output
  // PB.3 output
  // PB.4 output to MAX5436 /CS
  // PB.5 output (SPI MOSI)
  // PB.6 input  to MAX5436 DIN
  // PB.7 output to MAX5436 SCLK
  DDRB = 0b10111111;
  PORTB = 0b00010000;

  // Init the SPI
  // SPI master, no interrupts, MSB first, SCK low when idle,
  // clock phase = 0, SCK = fxtal / 16
  SPCR = 0x51;
  SPSR = 0x00;

  // Set up timer 0 to tick at 1 ms
  TIMSK = 2;                        // turn on timer 0 cmp match ISR
  OCR0 = 250;                               // set the compare re to 250 time ticks
  TCCR0 = 0b00001011;         // prescalar to 64 and turn on clear-on-match

  // Set up timer 1 to tick at 1 sec
  TIMSK = TIMSK | 0x10;             // turn on timer 1 compare match interrupt
  TCCR1A = 0;
  TCCR1B = 0x0d;                    // sets prescale to 1024, clear on compare match
  TCNT1 = 0;                        // and zero the timer
  OCR1A = 15625;                    // 0.0625us * 1024 * 15625 = 1 sec

  // Serial Comms For Mega32:
  //UCSRB = 0x10 + 0x08 ;
  UCSRB = 0x08 ;
  //UBRRL = 103 ;      // using a 16 MHz crystal (9600 baud)
  //UBRRL = 25 ;       // using a 16 MHz crystal (38400 baud)
  UBRRL = 51 ;         // using a 16 MHz crystal (19200 baud)



  // Init the A/D converter
  ADCSRA = 0b10000101; // enable ADC prescale to 32
```

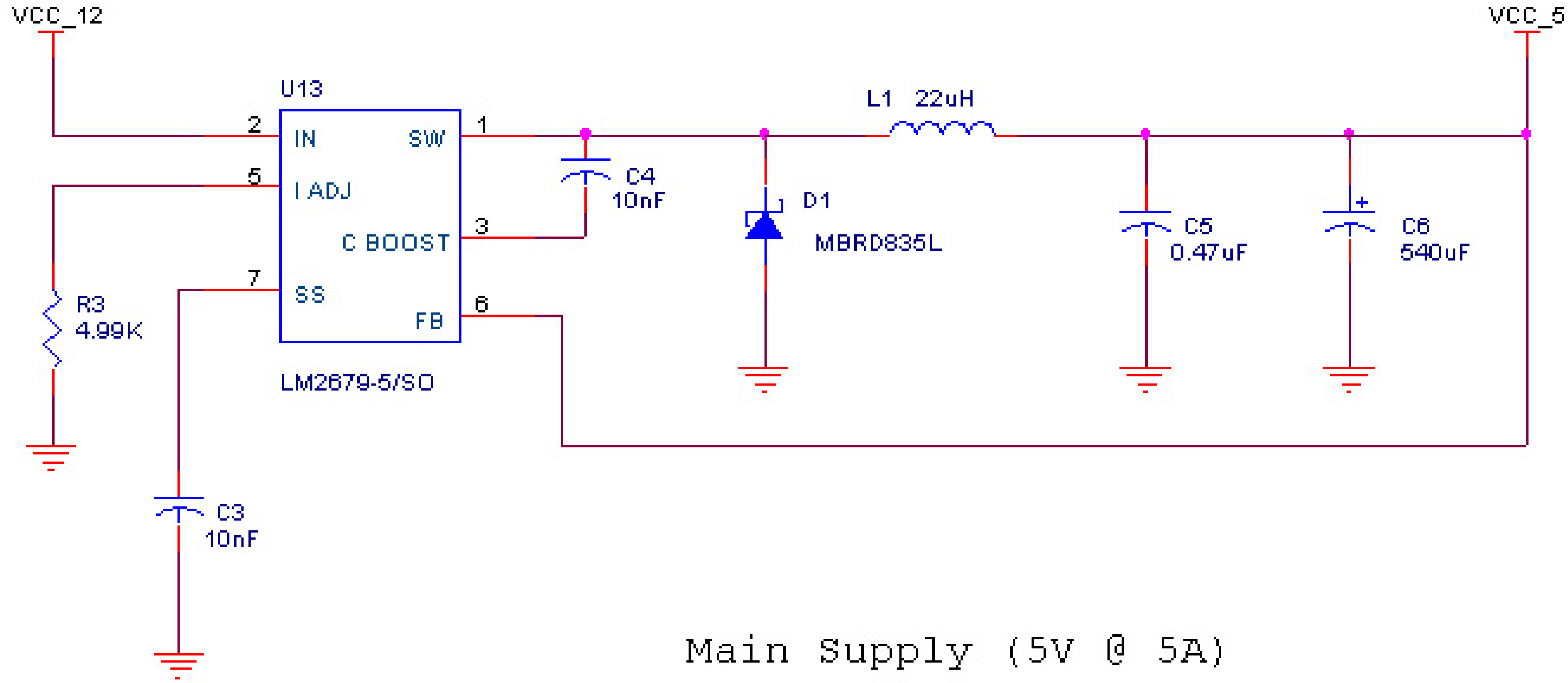
```
  ADMUX = 0b00100000;  // AREF, left adjust, no gain, chan 0 default

  // init the task timers
//  cb_time = cb_const;
  lcd_time = lcd_const;
  tx_time = tx_const;
  buck_time = buck_const;
  soc_time = soc_const;
  pv_time = pv_const;

  bat_voltage = 13.8;
  soc = 79;
  bat_temp = 70;
  out_temp = 65;

  charging = 1;

  //printf("System Online.\n\r");

  // Crank up the ISRs
  #asm
       sei
  #endasm
}
```



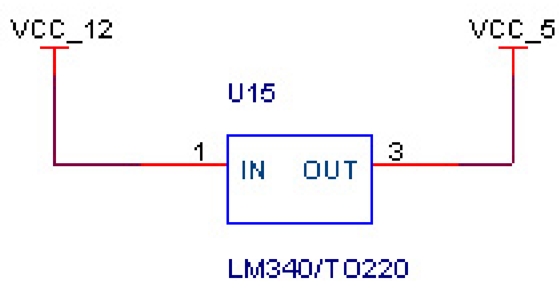
Title Receiver		
Size A	Document Number Vladimir Kozitsky	Rev 1
Date: Saturday, May 15, 2004	Sheet 1 of 1	



Title	MISC	
Size A	Document Number Vladimir Kozitsky	Rev 1
Date: Monday, May 17, 2004	Sheet 1 of 1	

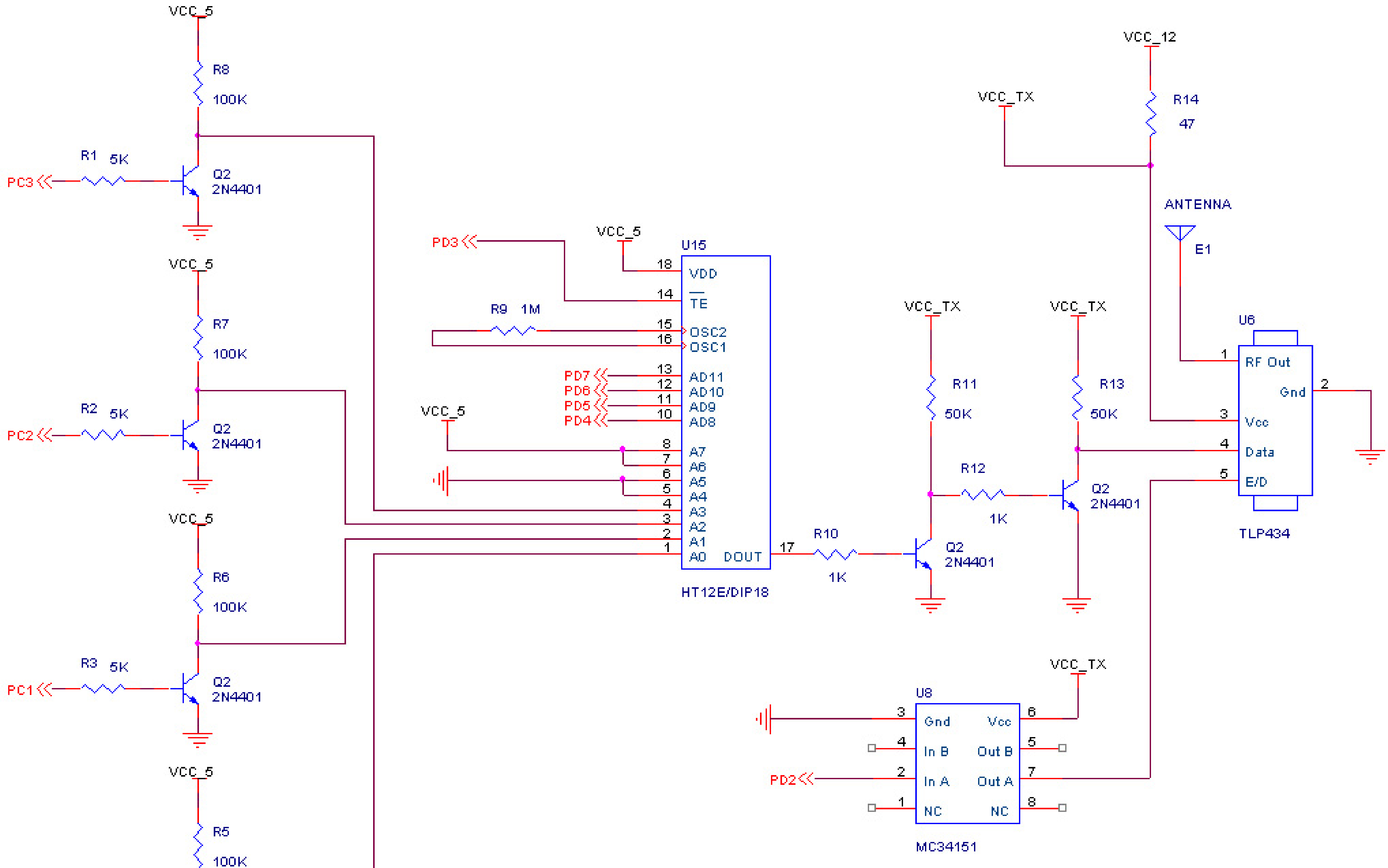


Main Supply (5V @ 5A)

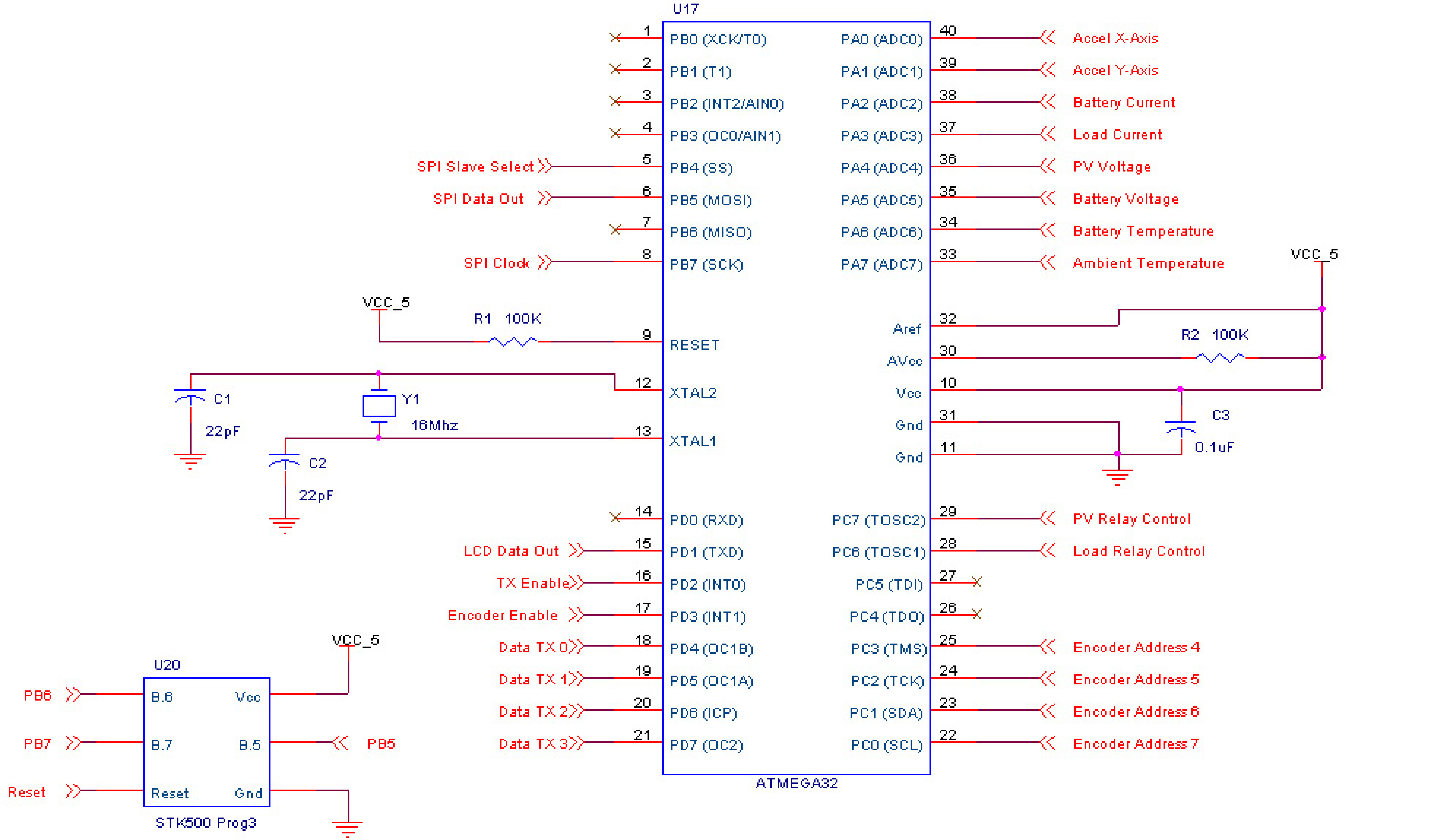


Alternate Supply (Cleaner)

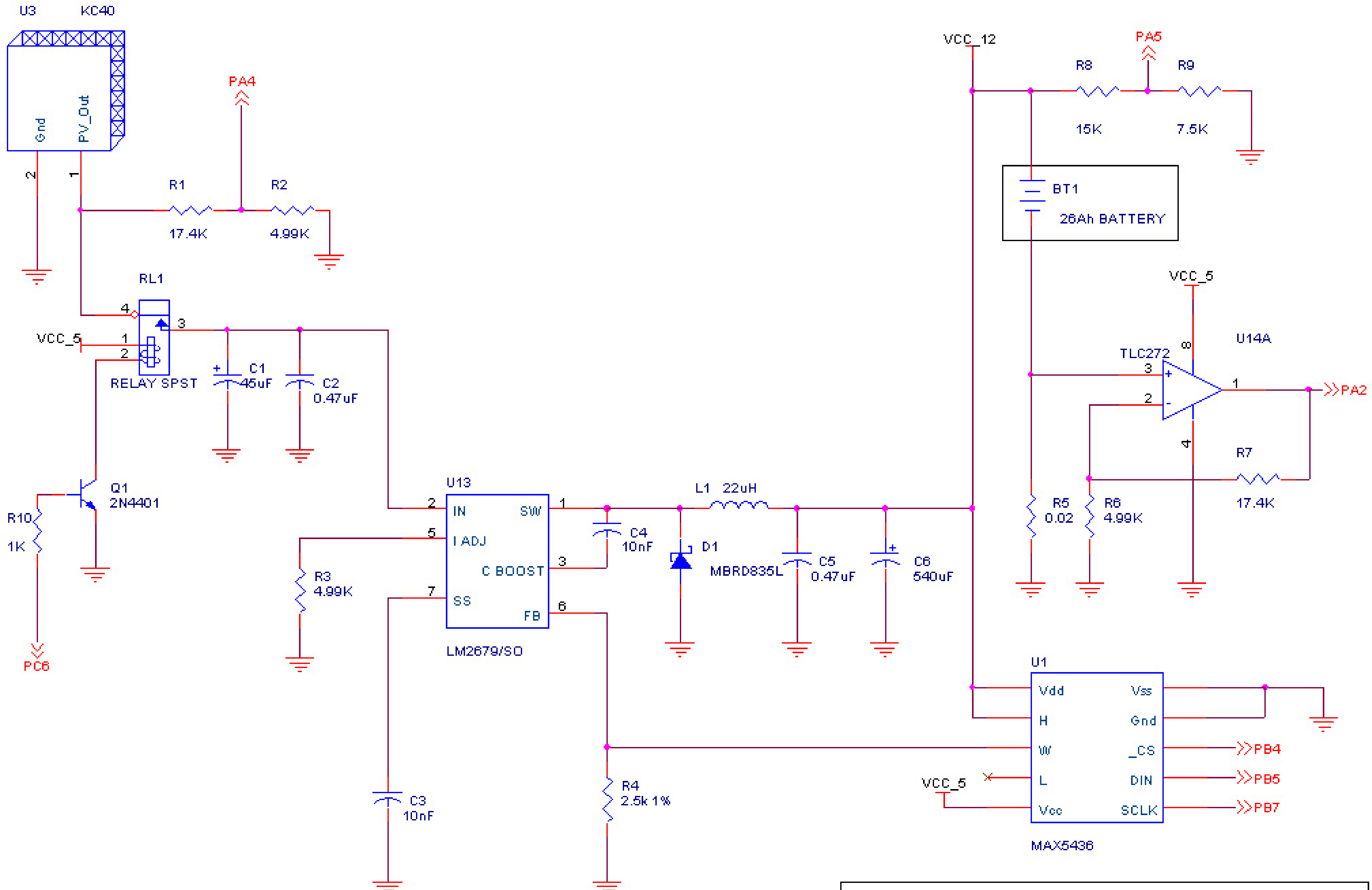
Title	Five Volt Supply		
Size A	Document Number <Doc>		Rev 1
Date:	Saturday, May 15, 2004	Sheet 1 of 1	



Title RF Transmitter			
Size A	Document Number Vladimir Kozitsky		Rev 1
Date: Wednesday, May 19, 2004		Sheet 1 of 1	



Title MCU Mega32		
Size A	Document Number Vladimir Kozitsky	Rev 1
Date: Monday, May 17, 2004	Sheet 1 of 1	



Title Power Converter		
Size A	Document Number Vladimir Kozitsky	Rev 1
Date: Monday, May 17, 2004	Sheet 1 of 1	

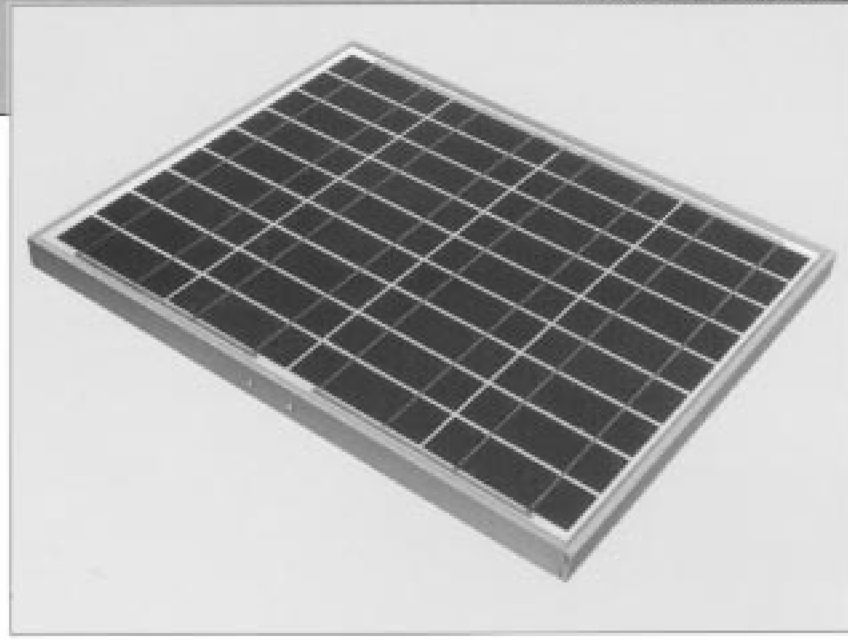
MODEL KC40

KYOCERA

KC40

HIGH EFFICIENCY MULTICRYSTAL PHOTOVOLTAIC MODULE

TYPICAL OUTPUT 40 Wp



HIGHLIGHTS OF KYOCERA PHOTOVOLTAIC MODULES

Kyocera's advanced cell processing technology and automated production facilities have produced a highly efficient multicrystal photovoltaic modules.
The conversion efficiency of the Kyocera solar cell is over 14%.
These cells are encapsulated between a tempered glass cover and an EVA pottant with PVF back sheet to provide maximum protection from the severest environmental conditions.
The entire laminate is installed in an anodized aluminum frame to provide structural strength and ease of installation.

APPLICATIONS

- Microwave/Radio repeater stations
- Electrification of villages in remote areas
- Medical facilities in rural areas
- Power source for summer vacation homes
- Emergency communication systems
- Water quality and environmental data monitoring systems
- Navigation lighthouses, and ocean buoys
- Pumping systems for irrigation, rural water supplies and livestock watering
- Aviation obstruction lights
- Cathodic protection systems
- Desalination systems
- Recreational vehicles
- Railroad signals
- Sailboat charging systems

SPECIFICATIONS

■ Electrical Specifications

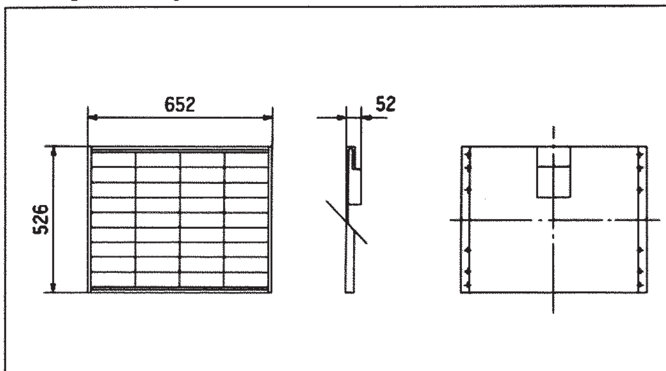
MODEL	KC40
Maximum Power	40 Watts
Maximum Power Voltage	16.9 Volts
Maximum Power Current	2.34 Amps
Open Circuit Voltage	21.5 Volts
Short-Circuit Current	2.48 Amps
Length	526mm (20.7in.)
Width	652mm (25.7in.)
Depth	52mm (2.0in.)
Weight	6.0kg (13.2lbs.)

Note: The electrical specifications are under test conditions of Irradiance of $1kW/m^2$, Spectrum of 1.5 air mass and cell temperature of 25°C

Kyocera reserves the right to modify these specifications without notice.

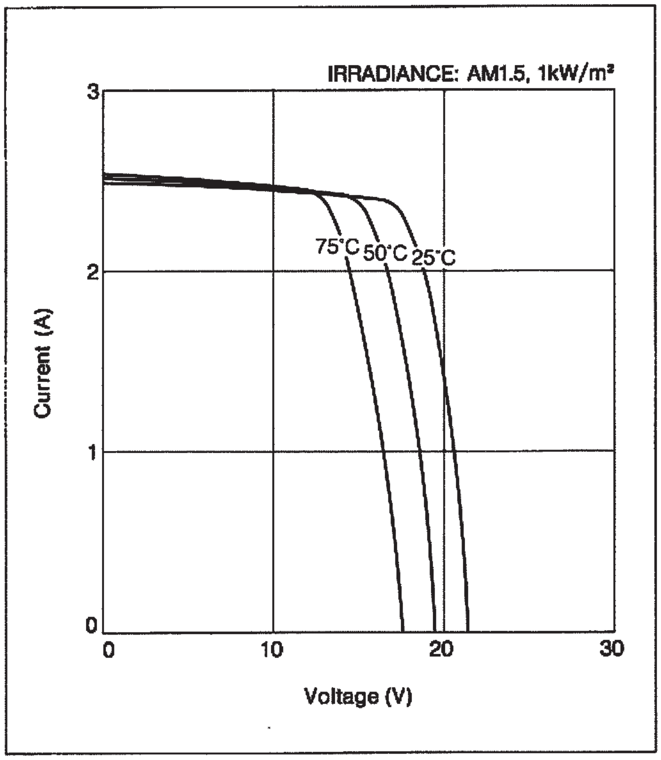
■ Physical Specifications

(Unit: mm)

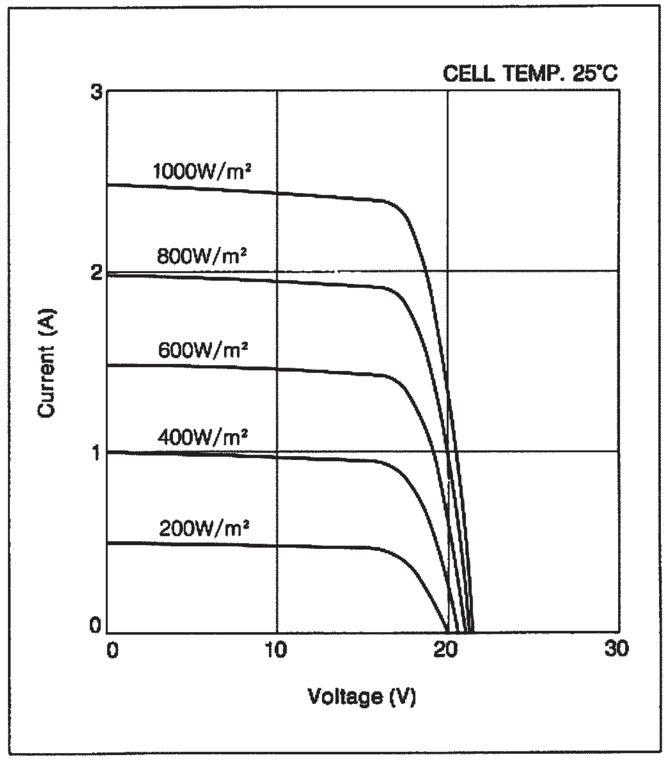


ELECTRICAL CHARACTERISTICS

Current-Voltage characteristics of Photovoltaic Module KC40 at various cell temperatures



Current-Voltage characteristics of Photovoltaic Module KC40 at various irradiance levels



QUALITY ASSURANCE

Kyocera multicrystal photovoltaic modules exceed government specifications for the following tests.

- Thermal cycling test
- Thermal shock test
- Thermal/Freezing and high humidity cycling test
- Electrical isolation test
- Hail impact test
- Mechanical, wind and twist loading test
- Salt mist test
- Light and water-exposure test
- Field exposure test

Please contact our office to obtain details without hesitation.

KYOCERA CORPORATION

■ KYOTO KARASUMA OFFICE

SOLAR ENERGY DIVISION
680 Karasuma-Bukkoji-Sagaru,
Shimogyo-ku, Kyoto 600, Japan
Phone: (075) 344-8241 Telex: KCCP J 65540
Telefax: (075) 344-8240

● KYOCERA AMERICA INC.

8611 Balboa Avenue, San Diego, California 92123, U.S.A.
Phone: (619)576-2647 Telex: ITT 4723069
Telefax: (619)569-9412

● KYOCERA FINECERAMICS GmbH

Fritz Müller Straße 107, D-73730 Esslingen, F.R.G.
Phone: (0711)9393417 Telefax: (0711)9393450

● KYOCERA (HONG KONG) LTD.

Room 803, Tower 1 South sea Centre, 75 Mody Road,
Tsimshatsui East, Kowloon Hong Kong
Phone: (02) 7237183 Telefax: (02) 7244501

● KYOCERA (HONG KONG) LTD. TAIPEI BRANCH

Suite 532,Asia Enterprise Center
602,Min Chuan E. Road Taipei, Taiwan
Phone: 7183595 7183596 Telex: 13724 KYOCETWN
Telefax: 7183587

● KYOCERA (HONG KONG) LTD. SINGAPORE BRANCH

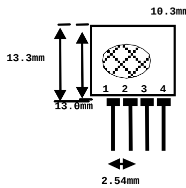
100 Beach Road, # 22.01/03
Shaw Towers, Singapore 0718
Phone: 2917900 Telex: 20133 Telefax: 2918468

CAT5T9604YS(J)

TLP434A & RLP434A **RF** ASK Hybrid Modules for Radio Control (New Version)

Easy-Link Wireless

TLP434A Ultra Small Transmitter



pin 1 : GND
pin 2 : Data In
pin 3 : Vcc
pin 4 : Antenna (RF output)

Frequency 315, 418 and 433.92 Mhz

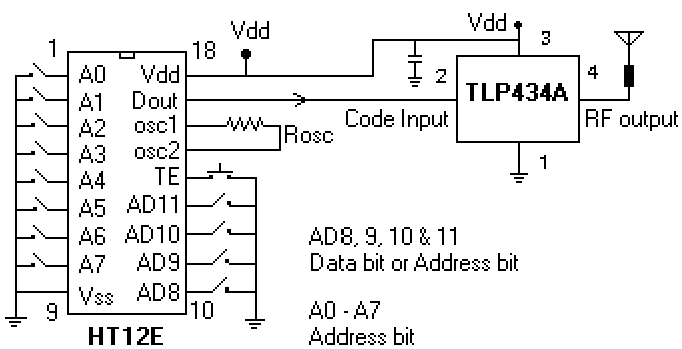
Modulation : ASK
Operation Voltage : 2 - 12 VDC

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Vcc	Operating supply voltage		2.0	-	12.0	V
Icc 1	Peak Current (2V)		-	-	1.64	mA
Icc 2	Peak Current (12V)		-	-	19.4	mA
Vh	Input High Voltage	Idata= 100uA (High)	Vcc-0.5	Vcc	Vcc+0.5	V
Vl	Input Low Voltage	Idata= 0 uA (Low)	-	-	0.3	V
FO	Absolute Frequency	315Mhz module	314.8	315	315.2	MHz
PO	RF Output Power- 50ohm	Vcc = 9V-12V	-	16	-	dBm
		Vcc = 5V-6V	-	14	-	dBm
DR	Data Rate	External Encoding	512	4.8K	200K	bps

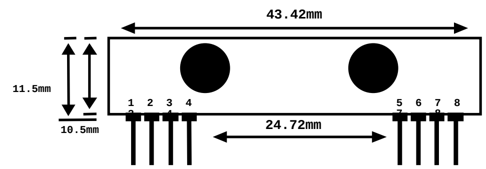
Notes : (Case Temperature = 25°C +- 2°C , Test Load Impedance = 50 ohm)

Application Circuit :

Typical Key-chain Transmitter using HT12E-18DIP, a Binary 12 bit Encoder from Holtek Semiconductor Inc.



RLP434A SAW Based Receiver



pin 1 : Gnd
pin 2 : Digital Data Output
pin 3 : Linear Output /Test
pin 4 : Vcc
pin 5 : Vcc
pin 6 : Gnd
pin 7 : Gnd
pin 8 : Antenna

Frequency 315, 418 and 433.92 Mhz

Modulation : ASK
Supply Voltage : 3.3 - 6.0 VDC
Output : Digital & Linear

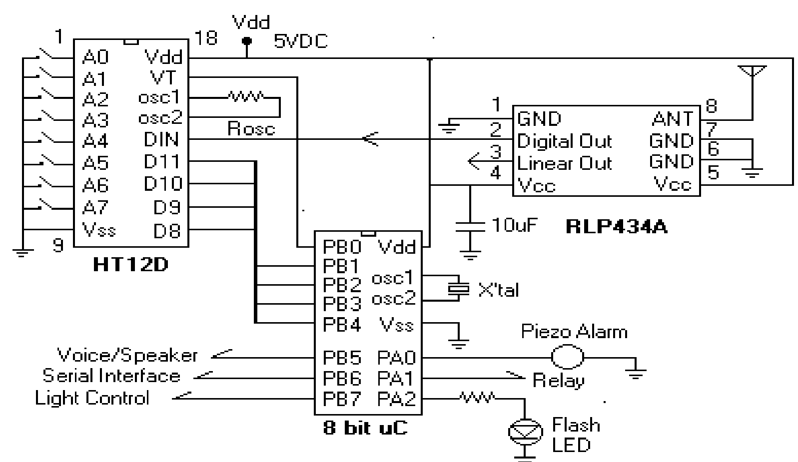
Symbol	Parameter	Conditions	Min	Typ	Max	
Vcc	Operating supply voltage		3.3	5.0V	6.0	V
Itot	Operating Current		-	4.5		mA
Vdata	Data Out	Idata = +200 uA (High)	Vcc-0.5	-	Vcc	V
		Idata = -10 uA (Low)	-	-	0.3	V

Electrical Characteristics

Characteristics	SYM	Min	Typ	Max	Unit
Operation Radio Frequency	FC	315, 418 and 433.92			MHz
Sensitivity	Pref		-110		dBm
Channel Width			+-500		Khz
Noise Equivalent BW			4		Khz
Receiver Turn On Time			5		ms
Operation Temperature	Top	-20	-	80	C
Baseboard Data Rate			4.8		KHz

Application Circuit :

Typical RF Receiver using HT12D-18DIP, a Binary 12 bit Decoder with 8 bit uC HT48RXX from Holtek Semiconductor Inc.



Laipac Technology, Inc.
105 West Beaver Creek Rd. Unit 207 Richmond Hill Ontario L4B 1C6 Canada
Tel: (905)762-1228 Fax: (905)763-1737 e-mail: info@laipac.com

LAIPAC TECH