# Design Document

*May13-06*

## Logging DC Wattmeter


Team Member:

**Ailing Mei**

**Andrew Kom**

**Chongli Cai**

**David Hoffman**

**Qiaoya Cui**

Advisor :

**Colin Christy**

Client:

**Garmin International**

# Table of Contents

# EXECUTIVE SUMMARY

Power consumption has become an important issue in designing electronic devices. When designing analog circuits and precise devices, people always need to measure the DC power consumed by those devices accurately. Most of the traditional analog wattmeters are made to run on the same power that it is measuring. This kind of wattmeter needs no external supply to run, and thus requires less maintenance. However, the traditional analog wattmeter has significant error in measuring the power consumed by a low power system. In addition, since the traditional analog wattmeter works based on electrodynamics, it is easily disturbed by an external magnetic field. Our project is aimed at developing a prototype digital DC wattmeter based on using microcontroller to accurately measure the power consumed by DC powered devices. Though the wattmeter would be useful an any DC power measuring situation, the application that has been specified is to measure the power produced by a solar array into a device.

# SYSTEM DESIGN

## • System Requirements

### Functional Requirements

**Compliance Requirements:**

- The system shall accept as close to a 1-24 VDC range
- Source the same 1-24V out the other side with a minimal voltage drop through the device
- Need the resolution to measure currents from 1mA to 3A
- Reads accurate measurements of voltage, current, time ,temperature
- Have a programmable logging time ideally from 1s to 10min
- Log the saved data to a removable SD card in a format easy to import with MS Excel
- The device needs the ability to run on battery power so that it doesn't distort the measurement being made.
- Utilize power management techniques to minimize the draw from the battery

**Interface Requirements:**

- Simple to set up a log interval
- Interface to display the data

### Non-functional Requirements

- Simple to connect in series between DC supply and load
- Provide easy user interface
- Lower price, ~ $50
- Pass the relative quality and legality test or certification
- Durable shell to protect from outside environment

## • System description

The system block diagram is shown in Figure 1. Since the device is connected in series between a DC supply and load, our design needs to not disturb the normal power consumption of the load. As shown below, input and output share the same negative line, while the input positive is routed through the current sensor.

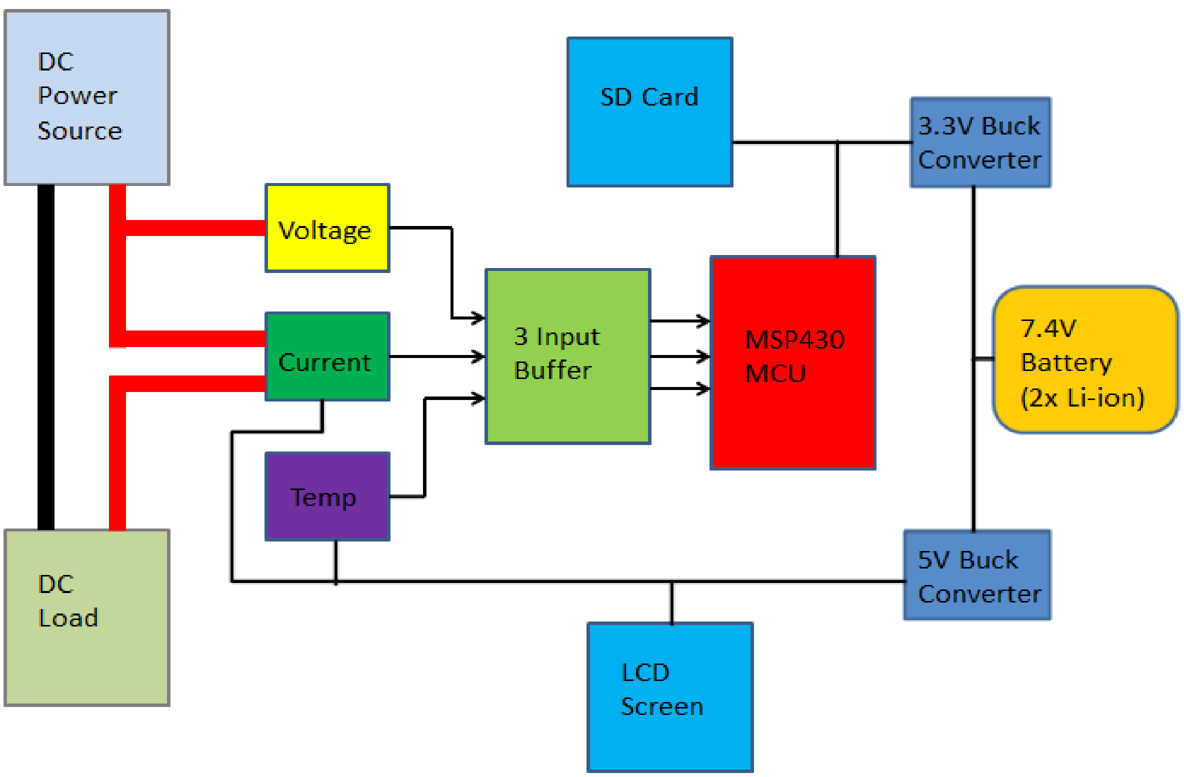


**Figure 1: System sketch**

## • Operating System

This device is designed for accurately measuring the power consumed by other devices running on DC. So the device will be used as a measurement instrument widely. In order to make this device more versatile, a decent range of voltage and current values need to be accepted. The following is the designed operating environment of the DC wattmeter.

- input/output DC voltage : Minimum value 1V , Maximum 24V (±0.125V)
- current measurement range: Minimum value 1 mA, Maximum value 3A (±0.01A)
- logging time interval: Minimum value 1s, Maximum value 10min

In addition, all the measurement data will be stored to SD card automatically.

## Functional Decomposition

- Overall Function

  Measure the power consumed by devices running on DC
- Sub-Function
  - Measurement Circuits

    Converting the current into proportional voltage (less than 3.3V)

    Converting the temperature into proportional voltage (less than 3.3V)

    Limit the input voltage up to 24V

    Attenuate the input voltage less than 3.3V
  - Microcontroller

    Use ADCs to process all the measurement data

    Count Logging time

    Control the 5V power source to turn off when unneeded
- Sub-Function
  - LCD module

    Display power, voltage, current, temperature and time
  - SD card

    Save the data

# DETAILED DESIGN

## Voltage Measurement

Since the input voltage is in the range of 1-24V, it needs a voltage attenuator to reduce the voltage to less than 3.3V (VCC) for it to be read by the ADC. For the initial design, this part is a simple voltage divider. The zener diode prevents more than 24V at the input of the device. The output of the voltage divider is fed into a buffer to get a clean measurement to the ADC.

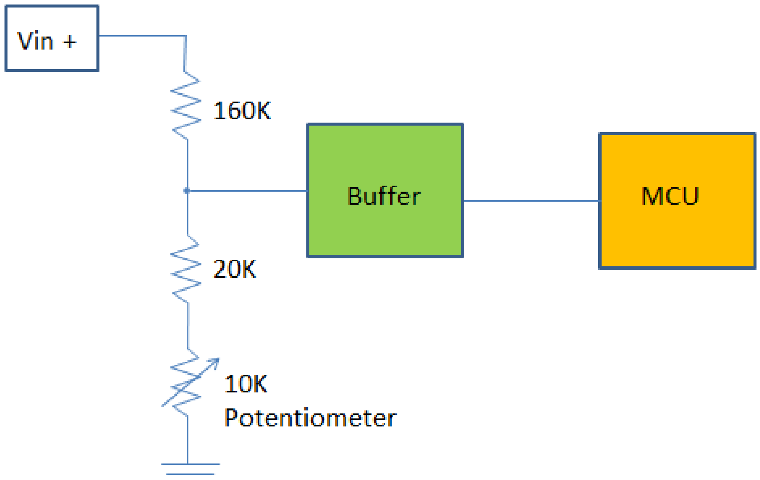


**Figure 2: Voltage Attenuator**

- ## Current Measurement

Current measurement is based on Hall-effect Linear Current Sensor. The output of current sensor is followed by an amplifier to arrange the voltage to be less than 3.3V. The Allegro System Inc. ACS714 Hall Effect-Base Linear Current Sensor is good choice for this, as it has an instrumentation amplifier built in, thus removing the need for an external one. The output of this chip is also fed into the buffer to produce a clean ADC signal.

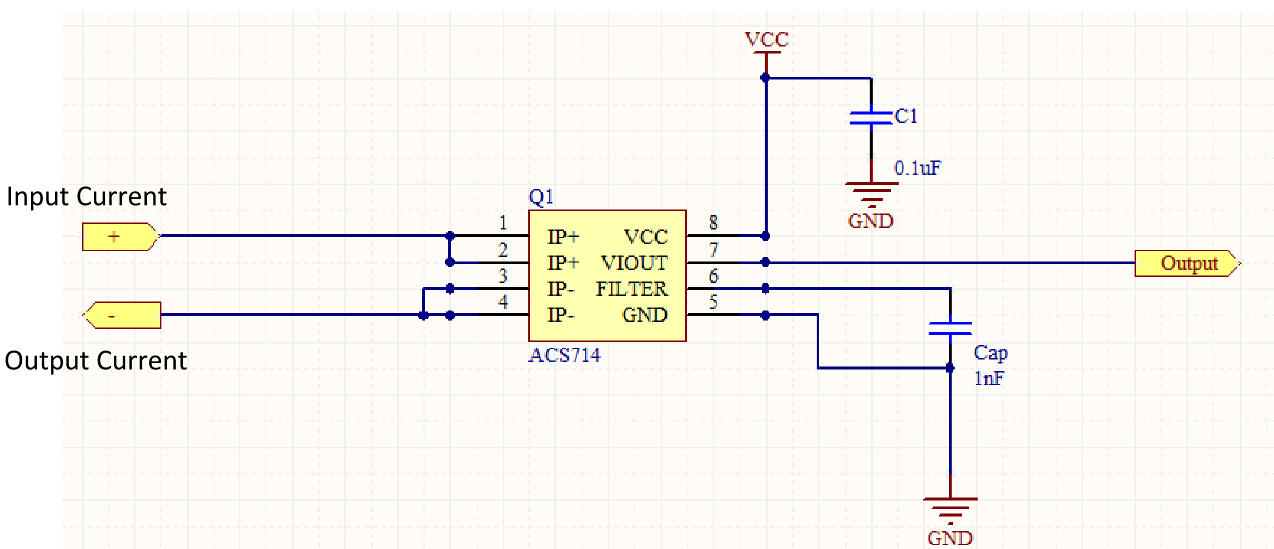


**Figure 3: Current Measurement circuit**

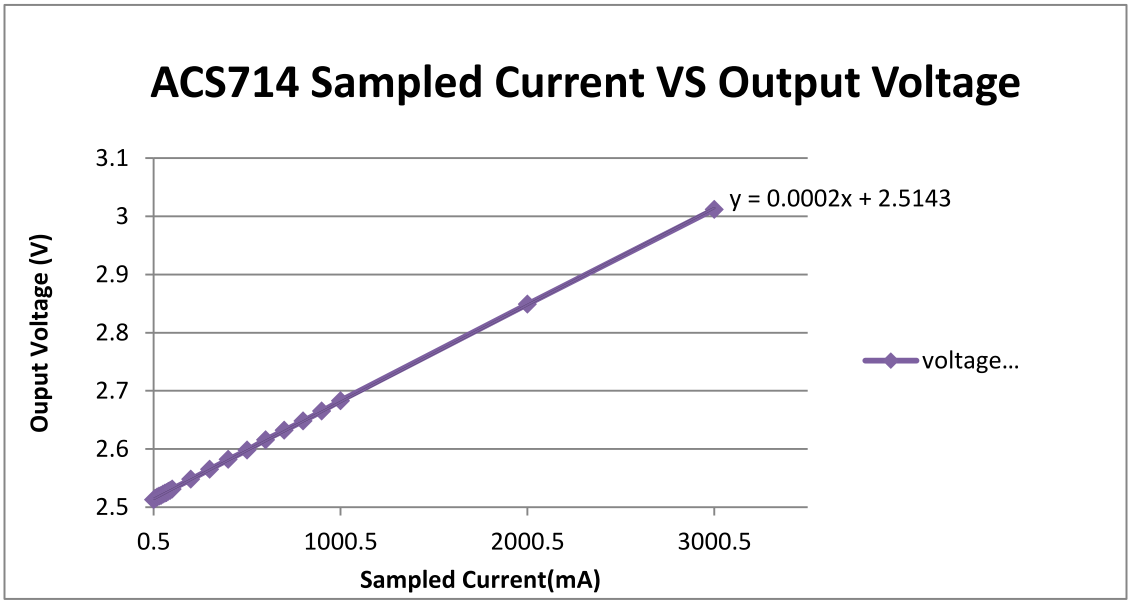


**Figure 4: Test Result for ACS714 Hall-Effect Current Sensor**

- **Temperature measurement**

  The temperature is measured by using LM35 temperature sensor from Texas Instruments Inc. The output voltage is linearly proportionally to the Celsius Temperature.

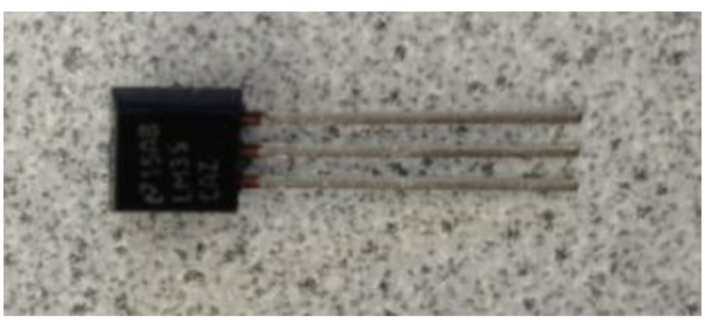

**Figure 5: TI LM35 Temperature Sensor**

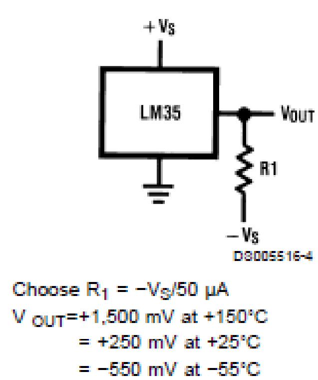


**Figure 6: Lab Measurement schematic**

We tested the Temperature Sensor in the Lab and found a linear relationship between output voltage and temperature. The Matlab result is shown in the Figure 7.

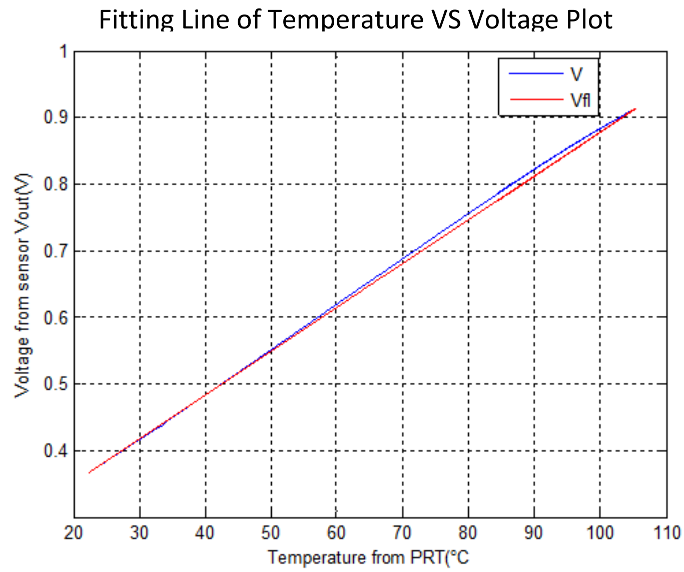


**Figure 7: Fitting Line of Temperature VS Voltage Plot**

The RED line is the measurement result and the BLUE line is the theoretical result.

- ## Power Management

After careful consideration, we determined that it would be very beneficial to be able to turn on and off the components whenever they are not needed. One big power saving decision was to turn off the LCD after displaying for a short time whenever the button is pushed. The largest consumer of power is the backlight that runs with the LCD, so having this off during normal operation, when not needed, will save a considerable amount of battery life. Additionally, there are 2 voltages we need to run everything in our circuit. The first is 3.3V for the MCU and SD card, and the second is 5V for the LCD, buffer, current sensor, and temperature sensor. Originally, we were given some single cell Lithium-ion cells by Garmin that have a capacity of 830mAh. After running some very rough runtime estimates, it was decided that a larger battery would be needed for any long-term life from the device. Thus, an 18650 single cell lithium cell with a capacity of 2Ah(2000mAh) seemed like a good choice.

Next, we needed a way of getting 5V and 3.3V out of the 3.7V (nominal) cell. Ultimately, to the lithium cell can run from a high of 4.2V all the way to a 3.0V during a safe discharge cycle. That means that a Boost converter would be needed to get the 5V, but a Buck/Boost converter would be needed for the 3.3 as it falls in the battery voltage range. However, due to the Buck converter having a better general efficiency, and the pre-made 3.3V and 5V buck ICs, it was determined to be easiest to series together 2 of the 18650 batteries, thus producing 7.4V nominal. Below is an image of the the 3.3V buck converter circuit. The 5V circuit is identical except the SD pin is instead tied to a pin on the MCU, to control on and off.

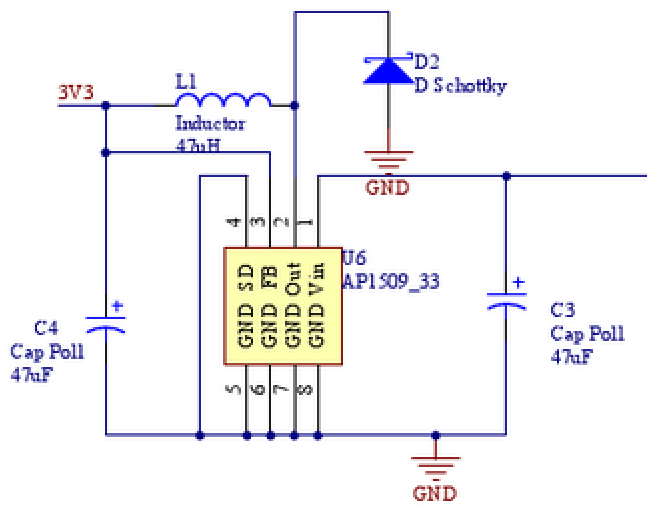


**Figure 8: 3.3V Buck Converter Circuit using an AP1509_33 IC**

- ## Microcontroller Unit

Our team worked on two Microcontrollers. One is the MSP430G2553 from Texas Instruments and another one is the PIC33FJ256GP710 from Microchip. We ended up moving our attention to the MSP430 due to its power saving features, like the “Ultra-low Power mode” that runs on interrupts only, and also it’s user friendly design with the TI Launchpad kit allowing for easy programming and testing.

### ➢ MSP430G2553

The MSP430G2553 is mainly used to do the ADC readings of the voltage, current, and temperature measurement results. Serial communication with the LCD and SD card through the SPI format is another important function. After consideration of power consumption, we added functionality to turn off the LCD when running, and turn it back on for a few seconds when a button is pushed. Additionally, it has the ability to turn off the 5V buck converter whenever a reading is not taking place. This ends up saving a decent amount of power given longer intervals. Hard data on the battery life being saved by this feature have yet to be measured. Below is the main flow outline of the program.

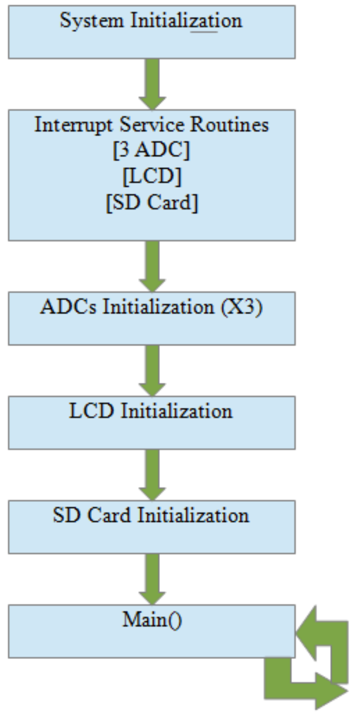


**Figure 9: Main Function Flowchart**

1. Main function

The Fig 9 shows the Main() function and its prototype functions. All sampling and UART communications is done via the Interrupt Service Routine. The ISR must be properly set up before most other devices. There will be 5 of them, one for each of the three ADCs and one for each of the two UARTs. In order for the ISR to work and the device to perform its function, it must be properly initialized, so each device will get its own initialization, except for the ADC, which will be initialized as a group.

The actual Main() function will be put in a continuous loop. It will have the responsibility of getting the measurement from the ADC once its ISR has triggered. This measurement will then be saved until all 3 ADCs have been scanned. Once this happens, Main() will then format this information for the SD Card and the LCD and send it to both of them using an ISR. The next thing will be to start the loop over. This will be accomplished using the interval the user selected to take these measurements.

ADCs have been scanned. Once this happens, Main() will then format this information for the SD Card and the LCD and send it to both of them using an ISR. The next thing will be to start the loop over. This will be accomplished using the interval the user selected to take these measurements.

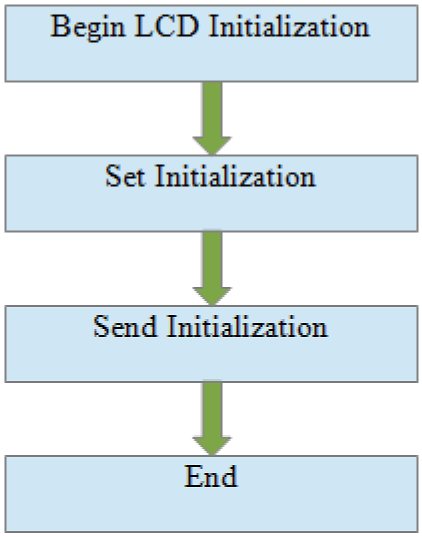


**Figure 10: LCD Initialization Flowchart**

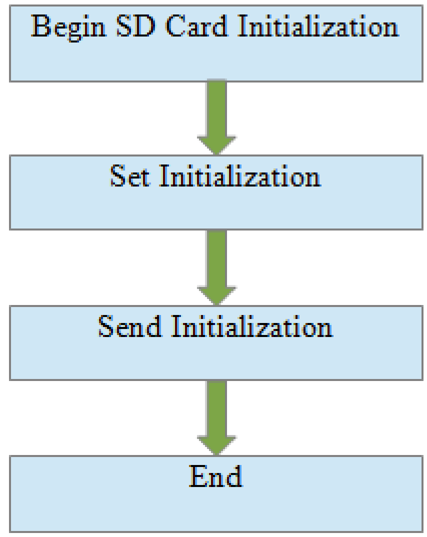


**Figure 11: SD Card initialization flowchart**

The SD Card initialization is very similar to the LCD initialization. This time we set and then send the initialization to the SD Card via the UART interrupt.

2. ADC initialization function

It is obvious from this flow chart that the initialization for the ADC is longer and more complicated than the initialization for the UARTs. All 3 of these initializations, one for each of the 3 ISRs are done at the same time. The important thing to remember here is that certain features need to be turned on in order for a good measurement to be taken. Those steps are all the steps before Set Interrupt Enable. The Set Interrupt Enable ensures that we can use the ISRs for getting the values from the ADCs while not using a lot of the processor time to get them. And, of course, the last step is to enable the ADC to start taking measurements on its inputs.

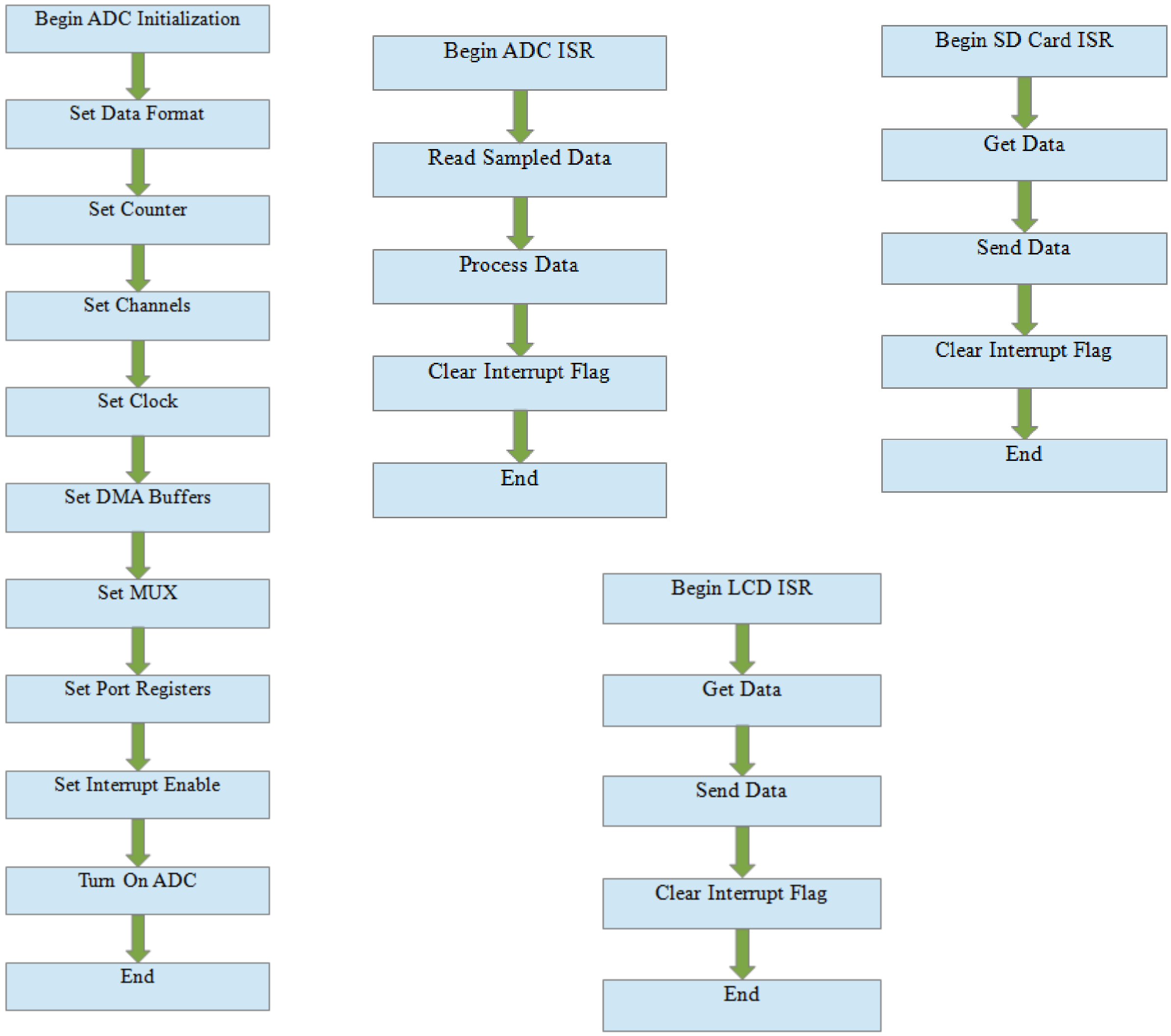


**Figure 12: ADC /LCD/SD Interrupt function Flowchart**

3. The Interrupt Function

There are 3 ADC Interrupt Service Routines, one for each initialization, which are for measuring the temperature, voltage, and amperage. The only thing the ISR does is read the data, process it so it is in a format that is useful, and then it resets the interrupt flag so another interrupt can occur.

The LCD ISR is responsible for only sending data to the LCD when there is data in the buffer to send. It must get that data, process it so the LCD can display it, and then send it to the LCD. The interrupt flag must be reset before data can be sent to the LCD again.

The SD Card ISR is similar to the LCD ISR in the way it operates, except the data does not have to be formatted before it is sent, as it should already be in the correct format. The data must however append to the existing file and not overwrite it. That should be a function of the SD Card and not our software.

- ## PCB Schematic

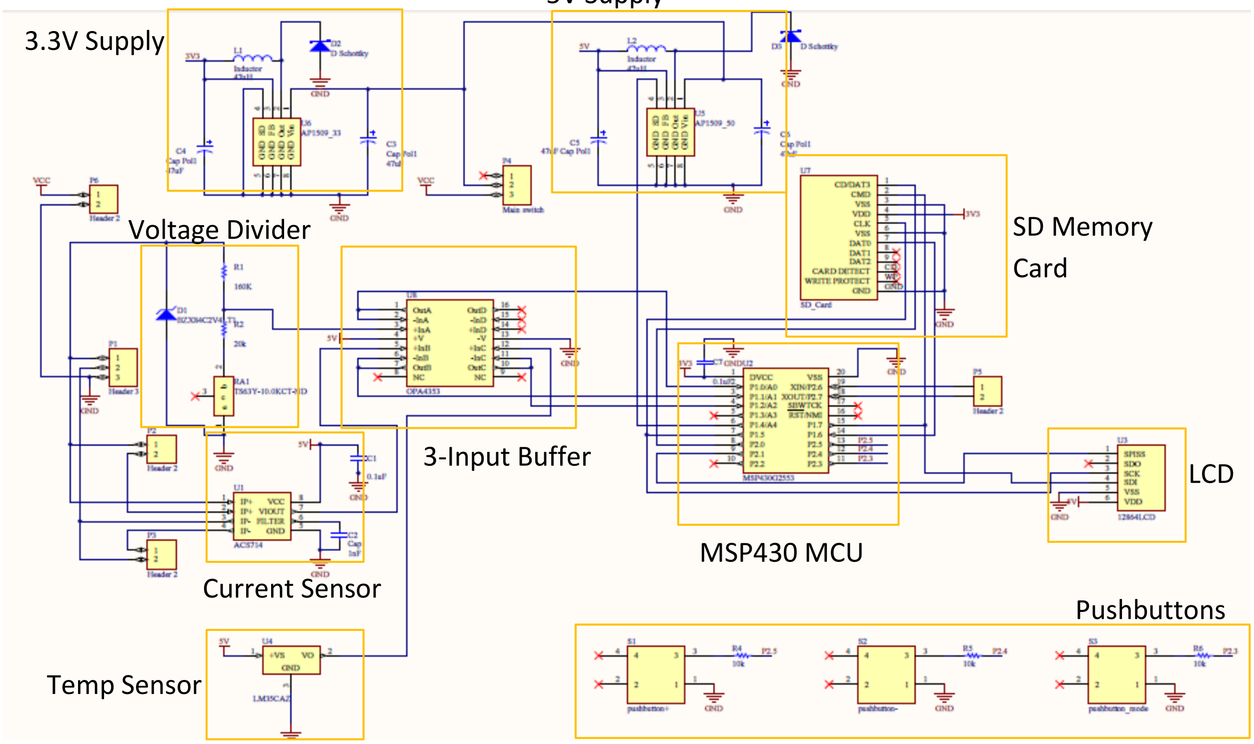


**Figure 13: PCB Schematic for Whole System (Based on MSP430G2553)**

- ## Test Plan

The test includes two stages: the component test part and the system test part. For the component test part, the current sensor, and the temperature sensor will be tested respectively. After that we will test the whole current measurement circuit and voltage measurement circuit. Once both of them satisfy the design requirements, we will connect them to the microcontroller to test the whole system. Before doing the system testing we need to make sure every component of the system satisfies the design requirement. Once finishing the first time whole system testing, we will draw the PCB for the system and deliver it to fabricate. After getting the PCB, we will test the system again to make sure the prototype satisfies the design requirement.

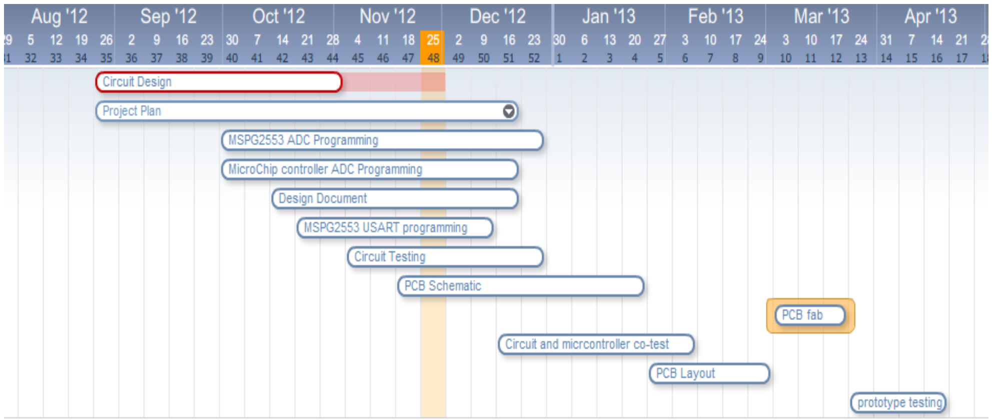


**Figure 14: Testing Timeline**

- Circuit Testing
- ➢ Have finished both current and voltage measurement test
- Microcontroller Function Testing
- ➢ Give data from voltage source to ADC and read the result from the LCD or Hyper Terminal
- ➢ Still working on ADC result testing for both microcontroller
- ➢ Finish testing before Mid of January 2013
- System Testing
- ➢ Finish before the end of January 2013
- ➢ Prototype Testing
- ➢ Final testing step
- ➢ Finish before mid of April 2013

## Operating Prototype

The operating prototype will be a Wattmeter that can operate reliably within a specific temperature range measuring a given power source. It will be some form of integrated circuits and digital logic that will have practical inputs and outputs, and a display for showing the measured characteristics of the power source.

## • Standards

Our project is a very much an initial design and we did not use many standards. For using UART of microcontroller, we use the SPI-type serial communication.

# PROJECT SCHEDULE

## • Work Plan

### Work breakdown structure

#### Research 20%

Research is an important part of design. According to the project goals, first is to understand how a DC wattmeter works and then come up with the structure of the whole project design. Based on the design constraints, it is essential to look for the best technical approaches and tools to accomplish the design such as the test equipment, simulation tools and device choices. At the same time, the cost of design is also considered in this portion of the work plan.

#### Design 30%

After identifying the project goals and researching the required components to make a wattmeter, a design could be formulated. This included the current sensor design, voltage divider design, ADC program design, MCU program design and PCB design and fabrication. The most important parts of the design are the measurement circuits and the main program being run by the MCU. To minimize the error in measurement, it is necessary to figure out the best method to design the circuit.

#### Implementation 30%

Once we obtained the desired circuit, implementation via simulation and breadboard will make sure the design is sound. The first step is to pick up a simulation tool such as Cadence to test the circuit. If the circuit is acceptable and practical, the next step is to build the circuit by hand. The devices involved in this project depend on the previous research results. The implementation is to combine all of these elements together and then see if we can get the desired results. After the design is confirmed on breadboard, design and fabrication of the PCB comes next. The last step is to test the final product.

#### Debugging/validation 20%

In the implementation section, the design should have been evaluated. If there are any errors in the design, they should be revised to meet the requirements of the design specifications.

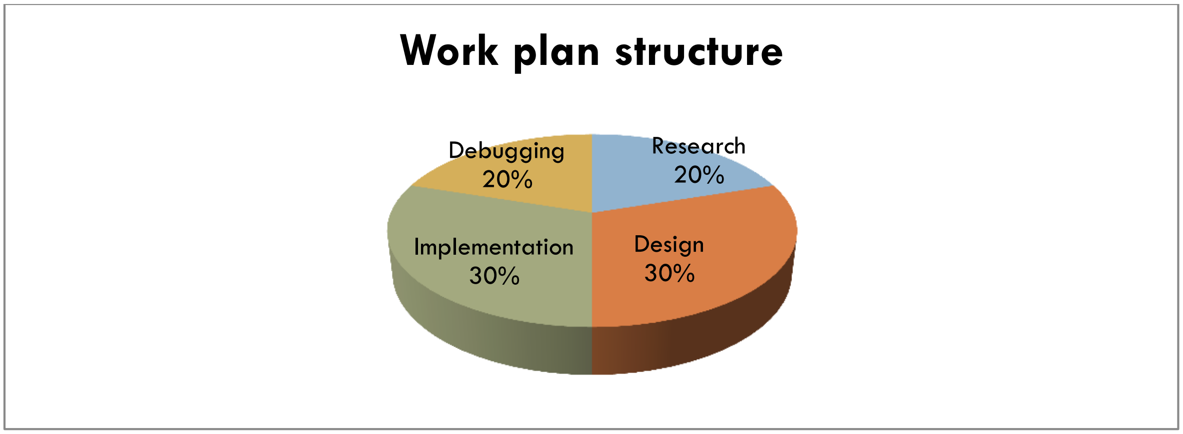


**Figure 15: Work Plan Structure**