**Smart meter**

Till now I have prepared the following modules of the meter:-

1. LCD driver, LED driver and User Interface library
2. ADC driver and metering library
3. Relay driver
4. Push button driver and logic
5. Timer libraries
6. Prepaid logic and accounting library
7. **LCD driver, LED driver and User Interface library:**

For the LCD of the meter I am using four seven segment displays because these are the most cheap I found till now. We can use other LCD’s but we will require some driver IC for them, which increase the cost. So for now I am using the seven segment displays. We can change that in future if we find some cheap/better option.

The current LCD can display positive/negative numbers and some characters also.

I am also using two LED’s for indication to toggle the information shown on the LCD.

The user-interface library uses LCD, LED and Push button to show the user different information available.

There are three modes available in the UI:-

1. Normal mode
2. Info mode
3. Error mode
4. **Normal mode:**

In this mode the UI continuously shows the balance available to the user and the power of the load connected to the meter. It toggles b/w these two information each second.

1. **Info mode:**

This mode is activated when the user press the info button. In this mode the UI toggles b/w the following information each second:

1. Validity(Number of days ramaining)
2. Voltage
3. Current
4. Power
5. **Error mode:**

This mode gets activated whenever there is some error e.g. Zero balance, Zero validity, Meter tampered.

In this mode the LCD continuously displays the Error code.

1. **ADC driver and metering library**

There is 10-bit ADC available in the Atmaga32 microcontroller. And there are 7 channels available. We can measure up to 7 parameters. I am using the ADC in single ended mode now. For future I want to use it in differential mode so we can measure DC also using the same meter. Right now I don’t know much about the filters to filter the measurable AC quantity. So for simplicity I am using Single ended mode with 5v reference. The Resolution of the ADC is 10-bits.

The sampling rate is 125 Khz.

**Voltage measurement:**

The maximum voltage the meter can measure is 500V AC, and the minimum measurable voltage is 0.48 V.

**Current measurement:**

For current measurement Current Transformer of 3000 turns is being used, with a burden resistance of 2Kohm. The maximum current we can measure using this is 5Amps. We can Increase this limit by changing the burden resistor. But it will decrease the resolution.

The resolution of current measurement is 5 milliamps.

**Metering Library:**

This library calculates the RMS values of the AC quantities, and then uses it to calculate the original values using the scale and offset values stored in EEPROM. The scale and offset will be calculated at the time of meter calibration.

To form an RMS 25 raw values are first averaged and form one average raw value.

25 of these averages are then used to calculate the RMS value.

1. **Relay driver:**

A 5 volt 100ohm relay is being used currently. To drive the relay we are using some transistor. The relay driver shuts the relay off when there is some error in the meter e.g. Zero balance, Zero validity etc.

1. **Push button driver and logic:**

The push button is used by the user interface library to active the info mode. The push button is used in active high configuration. It is interrupt based, i.e. some interrupt occurs when we press the button. More code needs to be written to take care of denouncing.

1. **Timer logic:**

Right now one of the timer counter register of the Atmega controller is used to maintain time. Whenever the timer overflows it triggers an interrupt. Timers tick is increased by one on each interrupt. We need to add some backup battery to keep it running when no power to the meter is available, or we can use some RTC like DS1307.

1. **Prepaid logic and accounting library:**

The meter works on pay as you go model. There are following variables in the accounting library:

1. **Balance**: The meter shuts down when there is no balance left into it.
2. **Validity**: The meter shuts down when the validity expires. On a new recharge certain part of the balance left from previous recharge is forwarded next.
3. **Carry forward energy units**: This number of energy units is forwarded to next month.
4. **Recharge**: A recharge contains the following parameters:
   1. **Balance**: Number of credits to be added to the meter.
   2. **Validity**: Validity of the recharge.
   3. **Carry forward units:** This many units can be forwarded to next month if not consumed with in the validity period.