Advanced Computer Systems
Fall 2017

CC1350 BLE Multimeter

Gil Akiva & Guy Khazma

Supervisor: Prof. Sivan Toledo

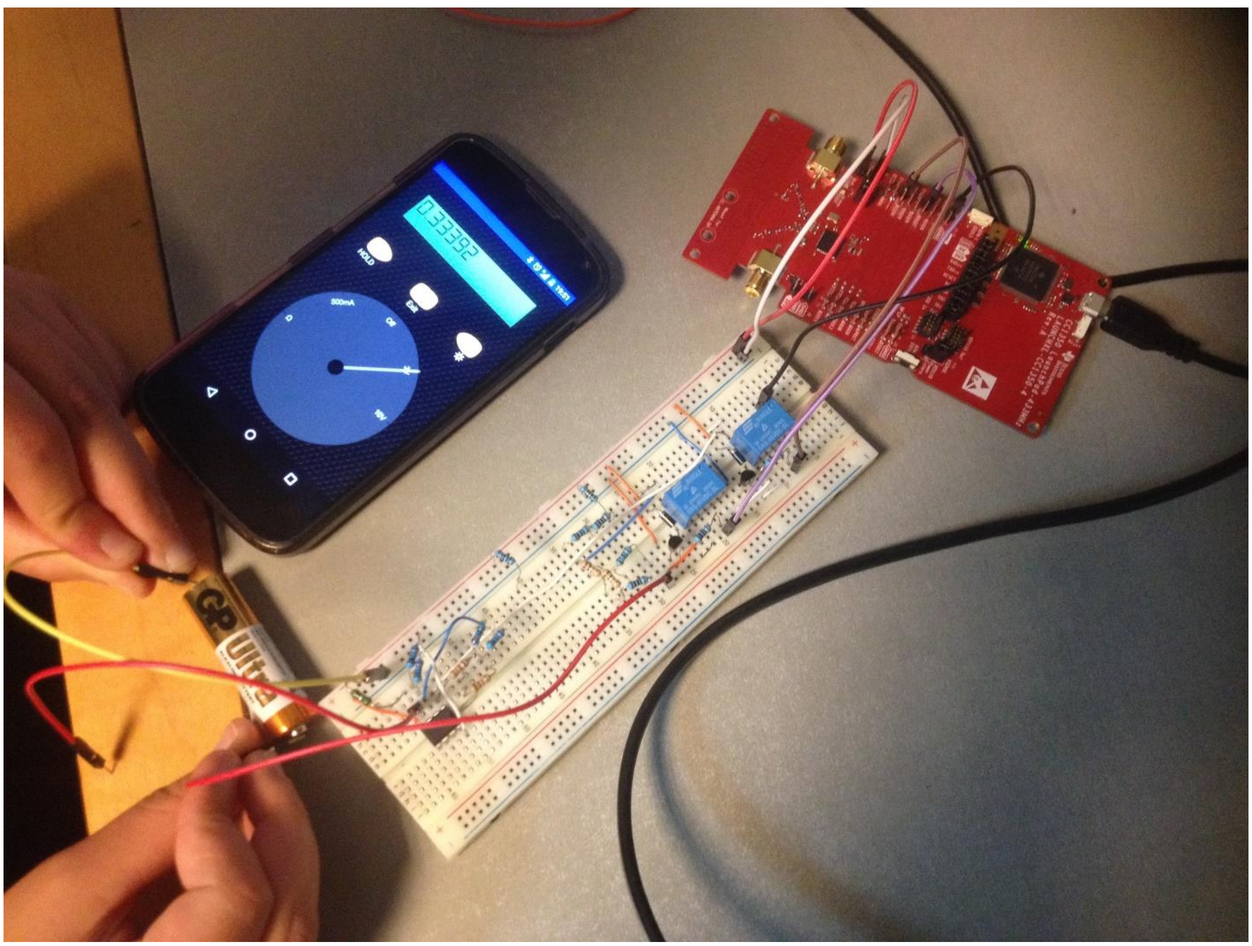
Abstract

As a final project in Advanced Computer System course we decided to implement a BLE Multimeter using TI CC1350 launchpad.

CC1350 is a wireless MCU targeting low power, long range wireless applications with Bluetooth low energy implementations.

The BLE Multimeter is a basic multimeter that enables measuring voltage and current, and transmits the measurements over BLE to a custom android app.

Possible applications can range from a cheap permanently attached monitor (for example for car battery) to a fully featured multimeter, using the BLE for logging or live debugging.



Project Architecture

Board Application

The board application is based on the BLE stack implementation provided by TI as part of their examples. We also used the built-in ADC buffer of the board to acquire the measurements from the multimeter circuit.

We implemented a custom multimeter GATT profile containing 2 characteristics:

- Mode characteristic – the value of the characteristic controls the mode of the multimeter according to the following specification:
    - 0x00 – Off mode. The multimeter is switched off by stopping the ADC conversion and the periodic clock that updates the profile.

  When multimeter's state is switched from off to one of the following states, it is first "turned on" by starting the ADC conversion and the periodic clock that updates the profile.
    - 0x01 – 3V mode. The MCU is configured to output low voltage in DIO21 and DIO22 to select the 3V mode in the multimeter circuit.
    - 0x02 – 10V mode. The MCU is configured to output low voltage in DIO21 and high voltage in DIO22 to select the 10V mode in the multimeter circuit.
    - 0x03 – 500mA mode. The MCU is configured to output high voltage in DIO21 to select the 500mA mode in the multimeter circuit.
    - 0x04 – Ohm mode. A place holder to demonstrate the ability to extend the features that the multimeter supports. Currently this mode is not supported by the circuit, so choosing it is being handled as off mode.

    - Other values are being ignored.

  The characteristic permits read and write operations.

- Measurement characteristic – the value of the characteristic represents the value of the measurements according to the multimeter mode:
    - Voltage mode – the value is the measured voltage in micro volts.
      In case of 10V mode, the voltage is being scaled according to the voltage divider formula by the MCU before the transmission.
    - Current mode – the value is the measured current in micro amperes. The value is being scaled according to the formula determined by the circuit by the MCU before the transmission.

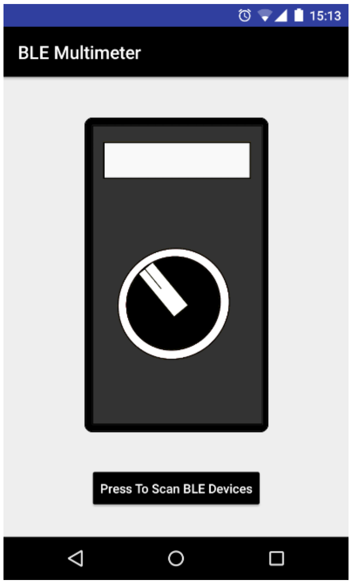
  The characteristic permits notify operation, and it is updated every second by acquiring the median of 100 consecutive measurements using the ADC buffer.

We used the PIN driver for controlling pins DIO21 and DIO22. These pins are used to control relays that switch between the different states of the circuit.
The measurements are done using the ADCBuf driver. The sampling is done via pin DIO23.

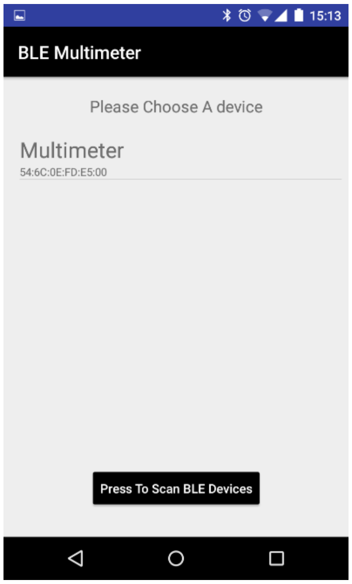
<u>Android App</u>

We developed an android app using Android studio and implemented a multimeter GUI that provides a familiar interface to interact with the BLE Multimeter.
We developed our application over Android's BLE example app.

Opening screen:



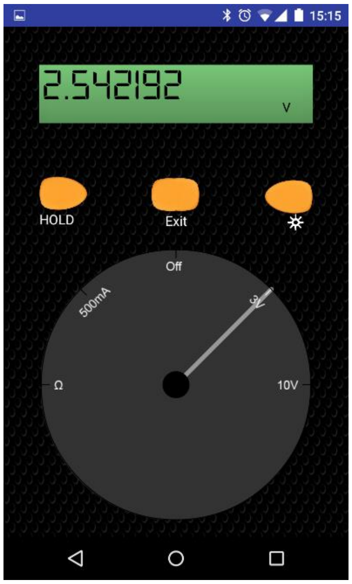
After pressing the scanning button, a list with all available BLE devices that support our custom Multimeter GATT service will appear:



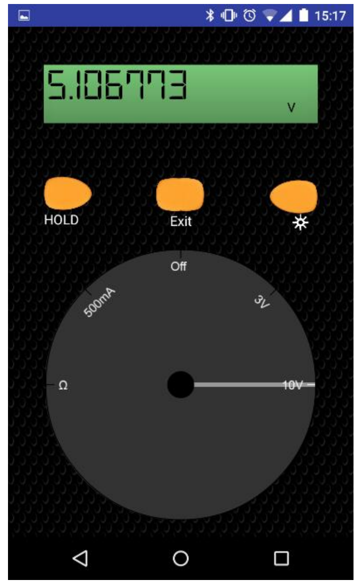
Selecting one of the devices will lead you to the main multimeter display:

Using the knob, you can switch between the different working modes:

Voltage measurement,
3V full scale

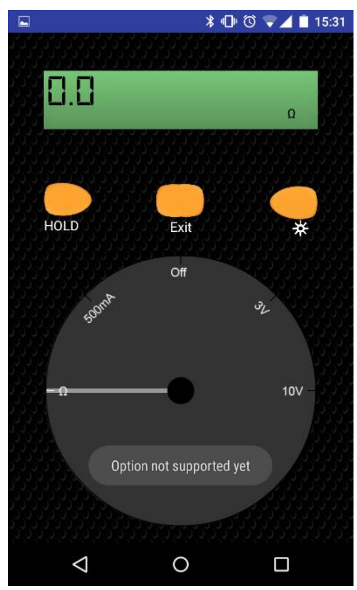


Voltage measurement,
10V full scale

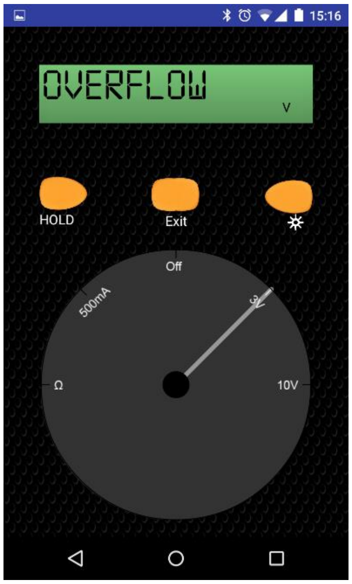


Current measurement,
500mA full scale

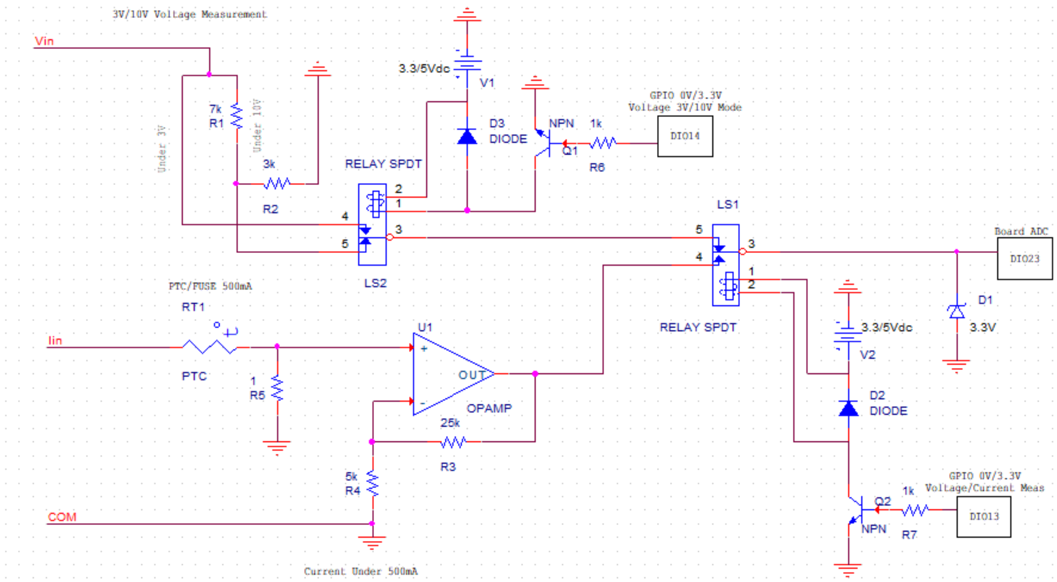
Resistance measurement,
currently not supported



When measured voltage or current exceeds the full scale:



Multimeter Circuit



Parts:

- 2x 5V relay
- LM358 opamp
- 2x 1N4001 Diode
- 2x NPN Transistor (8050)
- 3.3V Zener diode
- 500mA PTC/Fuse

Connections to the Board:

- DIO23 - Input to the adcbuf pin of the board, the voltage on this pin represents the value of the measurement as explained below.
- DIO21 - Drives a BJT transistor that is being used to switch the relay that controls the multimeter state between 3V and 10V
- DIO22 - Drives a BJT transistor that is being used to switch the relay that controls the multimeter state between voltage measurement to current measurement.

Voltage Measurement

- Voltages under 3V - the voltage is directly sampled by the microcontroller.

- Voltages under 10V – using voltage divider to scale the input voltage, yielding the following equation:

$$V_{meas} = V_{in}\frac{R_2}{R_1 + R_2} = 0.3V_{in} \rightarrow V_{meas,max} = 0.3 * 10V = 3V$$

Current Measurement

- The circuit can measure up to 500mA.

  Using the opamp properties we get the following equation:

$$V_{meas} = I_{in} * R_5\left(1 + \frac{R_3}{R_4}\right) = 6I_{in} \rightarrow V_{meas,max} = 3V$$

Issues during development

- TI built-in stack doesn't match the specific version of the board we acquired (we have cc1350-433 while the version of the stack use cc1350).

  The mismatch caused the ADC buffer capability to fail. In order to fix the issue, we had to replace the struct ADCBufCC26XX_adcChannelLut in the file simplelink_cc13x0_sdk_1_50_00_08\source\ti\blestack\boards_CC1350_LAUNCHXL\ _CC1350_LAUNCHXL.c so that it will match the struct of CC1350_433 board which can be found in the file simplelink_cc13x0_sdk_1_50_00_08\source\ti\boards\CC1350_LAUNCHXL_433\ CC1350_LAUNCHXL_433.c
- Circuit Related
  - We noticed that when using Zener diode to limit the voltage on the ADC pin, we get significantly lower voltage measurement. We suspect the reason is the leakage current the diode, so we decided to remove it.

    The results without Zener diode are much better, however for proper usage one should buy a Zener diode with parameters that suits the board specifications.
  - We wanted to use boards DIO pins for controlling the relays. However, these pins do not supply enough current. We had to use transistors (BJT NPN), so now the transistors control the relays, and the pins control transistors' conduction.
  - When measuring voltage without any voltage source connected to the ADC pin, we noticed that the measurement maintains a level of voltage of about 0.6-0.7V.

    If we connect the voltage source to GND, the measurement goes down to 0V, but once we disconnect the probe from ground the measurement goes back to about 0.6-0.7V. This leads us to the assumption that this is a property of the board.

Searching the technical reference manual and in TI's forums didn't result in any solution.

- Current measurements are very inaccurate oppose to voltage measurements which are quite good.
  We checked the circuit and concluded that the voltage that is being measured by the MCU is fairly accurate, however the opamp output voltage doesn't follow the equations formulated above.
  We detected several factors that contribute to the errors
  - Using the same GND pin for the transistors that control the relays and the COM probe causes current leakage which results in lower measurements. Therefore, we used separate GND pin for the relays and the COM probe.
  - The opamp is not ideal and for better measurement one should measure its actual gain and account for its V offset.
- In order to improve the accuracy of the measurements, we measured the actual size of the resistors we used and tuned the formulas accordingly.

External References

Demonstration video

Code

Multimeter CC1350 Board Code

Multimeter Android App Code