Boston University ENG EC464: Second Deliverable Test Plan

Team 14: Bikeputer (Re.Cycle)

Brian Kane

Katherine Murphy

Ben Paolillo

Jiwon Song

Todd Sukolsky

**First Deliverable Description:**

For the first deliverable, we demonstrated the functionality of the LCD Touch

Screen and the Beaglebone. For the LCD screen, we showed that we can navigate

between nine different screens and maintain data corresponding to the current screen

position; additionally, we were able to calibrate the screen whenever the user needs to do

so. This component is significant because the screen is the main user interface and will

eventually display the data supplied by the user, GPS, and various sensors. In addition, we demonstrated that we can stream NMEA strings and battery voltage through a UART serial connection. This component is significant because it is the battery management system.

**1.0 Battery Charging and Switching System**

**Second Deliverable Description:**

The charging and switching system takes advantage of the cyclist’s pedal motion to charge one battery while the other is being used. When the first battery, in use by the main computer module, is drained to a safe level, analog circuitry is used to connect the second battery to the main module and the first battery to the charging chip and generator. When the first battery recharges to a safe level, the switches are returned to their original position. The purpose of this circuit is to power the main module at all times while safely recharging part of the power supply by environmental means. The switching circuit will be demonstrated to function consistently as predicted, according to the various battery levels possible.

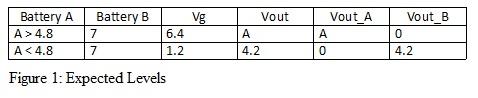
**Significance of Deliverable:**

This system is significant because it powers the entire main module, with the exception of the backup battery, which separately powers only the real time clock. The system is essential in satisfying the following major requirements from our customer: wall chargeable in under twelve hours; main functions powered for at least sixty minutes; supplemented by at least 50% through the environment. Each Li-Ion battery may be recharged using an appropriate wall charger within 12 hours. From a fully charged state, one battery should easily be able to handle a 60 minute ride without needing to be switched. However, if the user does not charge the batteries between 60 minute rides, the switching circuit will take care of this. The ride is supplemented at least 50% by the environment in that one battery charges while the other is in use.

**Measurable Criteria for Success:**

Qualitatively, the test will be considered a success if we can show that, when Battery A reaches such a level that it needs to be charged, it will be switched to charging position while Battery B is simultaneously switched to supplying position. Additionally, it must be shown that when Battery A is sufficiently recharged, the two batteries switch back to their original positions. Finally, it must be shown that the output of the switching circuit is a continuous voltage supply.

Quantitatively, it must be shown that the output voltage (the supply voltage to the main module) is acceptable and behaves as expected, and that the switches operate at times that keep the batteries from over and undercharging. Table 1 shows what the outputs must be for each battery level in order for the demonstration to be considered a success. Vout indicates the switching circuit output to the main module; Vout\_A indicates the contribution of Battery A; Vout\_B indicates the contribution of Battery B; Vg indicates the voltage at the gates of the switches. All units are measured in volts.



**Equipment and Setup:**

The switch circuit will be set up on a breadboard, powered by an Agilent E3631A Triple Output DC Power Supply, to simulate battery levels. The batteries themselves will not be used due to the fact that they cannot be drained/charged safely in the time given for demonstration, and that the purpose of this demonstration is to show switch functionality and control. An Agilent 34401A 6½ Digit Multimeter will be used to monitor the output voltages and LEDs will indicate the status of the switches, so it is clear which “battery” is supplying and which is being charged at each given stage. The circuit schematic (“Switch Circuit”) is included in the Schematics section. Battery A is designated as the master battery, and a comparator is used to monitor this battery and control the gates.

**Measurement Plan/Data Collection:**

The output voltage Vout will be measured at each stage of the process and compared with expected/required values. An oscilloscope will also be used at the output to show smooth transitions between stages.

**Assessment Process:**

The demonstration will be assessed according to how closely the results match the predictions.

**2.0 Communication Protocols and BeagleBone functionality**

**Description and Significance:**

The second deliverable testing will also illustrate the main communication protocols between chips, real-time clock features, GPS streaming and location parsing.

The communication protocol between all chips is based loosely on TCP and HTTP/1.1 protocol, all done over UART. Communication is done between all three chips, the watchdog AVR, BeagleBone and Graphics AVR, for the following reasons: First, the BeagleBone will have GPS data streaming into it. This data contains the actual time in UTC (five hours ahead of east coast). The BeagleBone will share this with the watchdog AVR to establish a real time clock for the system. Second, the graphics processor needs to communicate with the BeagleBone so that trip data can be sent from the AVR to the Bone, which is where a USB drive will be plugged in and data will be offloaded. This trip data includes average speed, time, distance and average heart rate. Thirdly, the graphics AVR needs to be able to synchronize with the real-time clock on the watchdog AVR because the LCD screen will display time.

On the BeagleBone scripts have been deployed to do several things: First, GPS data coming in onto the UART2 ports will be streamed into a file for reading when a USB dump is about to occur. Second, this GPS data is parsed on startup for an accurate time, and if none, will vocalize it to the watchdog AVR. Third, the bone has a parser to take the raw NMEA strings and convert them into real Longitude and Latitude coordinates.

Communication can be measured by viewing UART strings sent from the different chips to one another. Because the BeagleBone is an ssh device, strings can be printed to the terminal that it receives. These strings include initial time, NMEA strings and trip data from the graphics AVR. Another aspect we can measure is the timeout interval for each chip. The timeout of most connections is set to ten seconds. If the timeout occurs, the connection is killed and a new connection needs to be established. Debug LED’s have been placed on the board to display of the AVR’s during communication.

**Equipment Setup**

The BeagleBone's UART4 ports, on P9 pins 11 and 13, is connected to the graphics AVR's UART0 ports. Also, there are GPIO's connected between the two \*to act as an interrupt. The Beaglebone sends an interrupt out from P8, pin 4, which is received as a Pin Change Interrupt on the graphics AVR. The graphics AVR's UART1 ports are connected to the watchdog avr's UART1 ports along with an interrupt from the watchdog to graphics avr. Next, the bone's UART1 on P9, pins 24 and 26, are connected to the watchdog AVR's UART0 ports along with an interrupt GPIO attached to the WAVR (watchdog avr) INT2 pin. Finally, a GPS streaming device is connected to the USB port of the BeagleBone to demonstrate the parsing and streaming of NMEA strings.

**Measurement Plan and Data Collection:**

First an overview of the connections will be given. Then we will execute a program that will pipe GPS NMEA strings through the USB port into a file for approximately 20 seconds. At the end of this another script will be called that will parse the NMEA strings and create another file that has only longitude, latitude and time coordinates. During the inital piping a second file should be created that has a “time” string representing the EST time of the location lock. This is what will be sent to the WAVR in future releases. Because the incoming NMEA strings are dead/null a test file is being used to simulate the time placement functionality.

Second, two ssh terminals will be launched that connect to the BeagleBone. From these we will run two very similar scripts. Once the command terminal is open they should connect to the two AVR’s and then an input string can be pinged to them individually. To test the real-time clock functionality of the watchdog we will see if it accepted the pre-programmed EEPROM time, ping the AVR for current date, time, “both” (should display time and date) and set the time to a new value that looks exactly similar to the ones in the initial time file from the NMEA parsing. After this the AVR should ping back the exact string we sent and the script should exit. Upon re-execution we should be able to ping for the time and see the value we just set.

**Assessment Process:**

A successful test can be agreed upon if the scripts are able to accurately parse locations out of NMEA strings, store time values in another file for writing to the AVR’s and communication lines are open and running, with error correction, between the bone and the other processing chips. Even if there is an error in a string or timeout, the scripts and avrs should interact with one another and resolve conflicts.

**3.0 Power and Watchdog AVR Functionality**

**Description and Significance:**

Another feature for testing is the entire watchdog AVR functionality. This includes the real-time clock that is maintainable on backup battery for more than 5 days, communication with the BeagleBone and graphics AVR, regulator cut-off and signaling to the other chips, ADC readings of the main battery and temperature readings of the circuit board. The last two functions, ADC and temperature monitoring, are used as safety features for the system to avoid overheating and possible battery damage.

**Measurable Criteria:**

On startup the watchdog will grab previous date and time from EEPROM and then wait for a UART string with initial time from the bone and then set it's own clock if its correct/valid. The second measurable form is the ADC and temperature reading of the main battery and the circuit board. If the main battery voltage goes to a minimum level power will be cut from all modules except the watchdog. This is also the case if the circuit board gets too hot. We can see the readings via UART strings between the bone and the watchdog, or by simulating low voltage with a change in resistors. If a concern is found, the watchdog should toggle GPIO lines alerting the bone and graphics AVR they are about to be powered down and then kill power after 4 seconds. We can simulate this with LED's. The last bit of measurable criteria is the ability to switch between power sources in the event of a low power or high temperature error. In the event of an error the bone and GAVR will be alerted of an issue to allow final processing, then power will be killed. At this time the power source of the WAVR needs to be switched from the main battery’s to a backup. We have chosen a 9V battery because of it’s longevity of life for such small current loads. There is a power mux chip located on the breadboard that will switch between the main power. By measuring the input voltages to the mux we can determine whether it switched appropriately.

**Equipment Setup:**

There are many LED's attached to the watchdog, refer to the attached schematic for those. To elaborate, the ADC of the battery is being referenced to a 1.1V internal source and the reading is being taken on ADC0. The temperature monitor communicates SPI and is located on the SPI0 ports on pins five through eight. There is a 32.768kHz oscillator on TOSC1/2 for the real-time clock and a 20MHz oscillator on the XTAL1/2 pins for the system clock. The communication lines are the same as state in section 2.0. See schematic for more details

**Measurement Plan and Data Collection:**

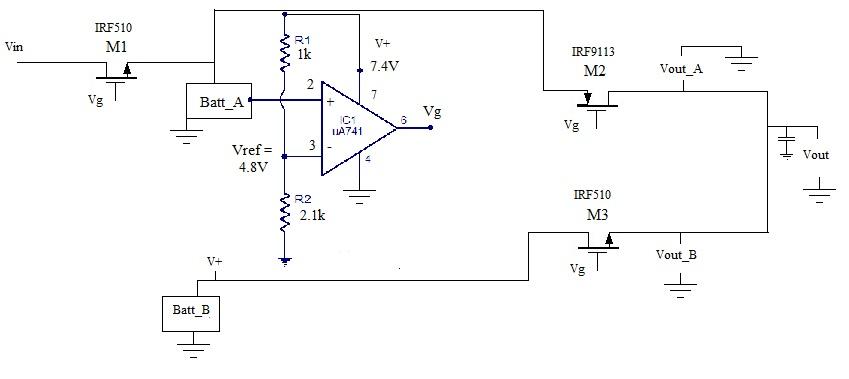
Continuing on from the test in section 2.0 we will ping the WAVR for temp readings, hand calculate what they mean, and also grab ADC readings of the battery voltage. This is similar to the first deliverable testings except things are being done with all of our specifications. By measuring the ADC readings the WAVR can make a judgement on whether to cut off main power. After pinging for these values we will show the outputs of the regulators (one linear 5v output regulator and one 5v-3.3v regulator) that turn a 9V source into a 3.3V source. We will then toggle the enable line on the 5V linear regulator and observe that the 3.3V output is 0 V when no enable. Then, we will show that the output of the power mux never drops below 3.1V during a switch of the enable line (when power is switched from the 9V to an accessory dev board).

**Assessment Process:**

A successful test entails the ability to accurately read ADC readings from the battery as well as temperature measurements, alert the other chips if a power down is imminent, toggle the 5V regulator using a 3.3V GPIO output pin on the AVR and have the power mux switch power to the microcontroller seamlessly.

**4.0 Schematics:**

Switch Circuit:



Battery Block Diagram:

