Interrupts and Real Time

Electrical Engineering 474 Lab 4
University of Washington

Name	Student ID
Denis Jivaikin	1432129
Novin Changizi	1434956
Ting-Yu (Jacky) Wang	1333301

INTRODUCTION

This lab focuses on the controlling of proximity sensors and motor drivers (H-Bridges) using interrupts. In addition to the core project, the group has decided to go above and beyond by adding remote control and autonomy capabilities to the tank. The tank can be remotely controlled by a smartphone or by using Leap Motion Camera.

HARDWARE OVERVIEW

The motor driver is connected directly to the BeagleBone's GPIO and PWM pins while the proximity sensors are hooked up directly to the BeagleBone's ADC pins. Moreover, a Bluetooth module was incorporated to allow remote control of the tank using either a phone application or the Leap Motion Camera. The hardware interconnection is as follows:

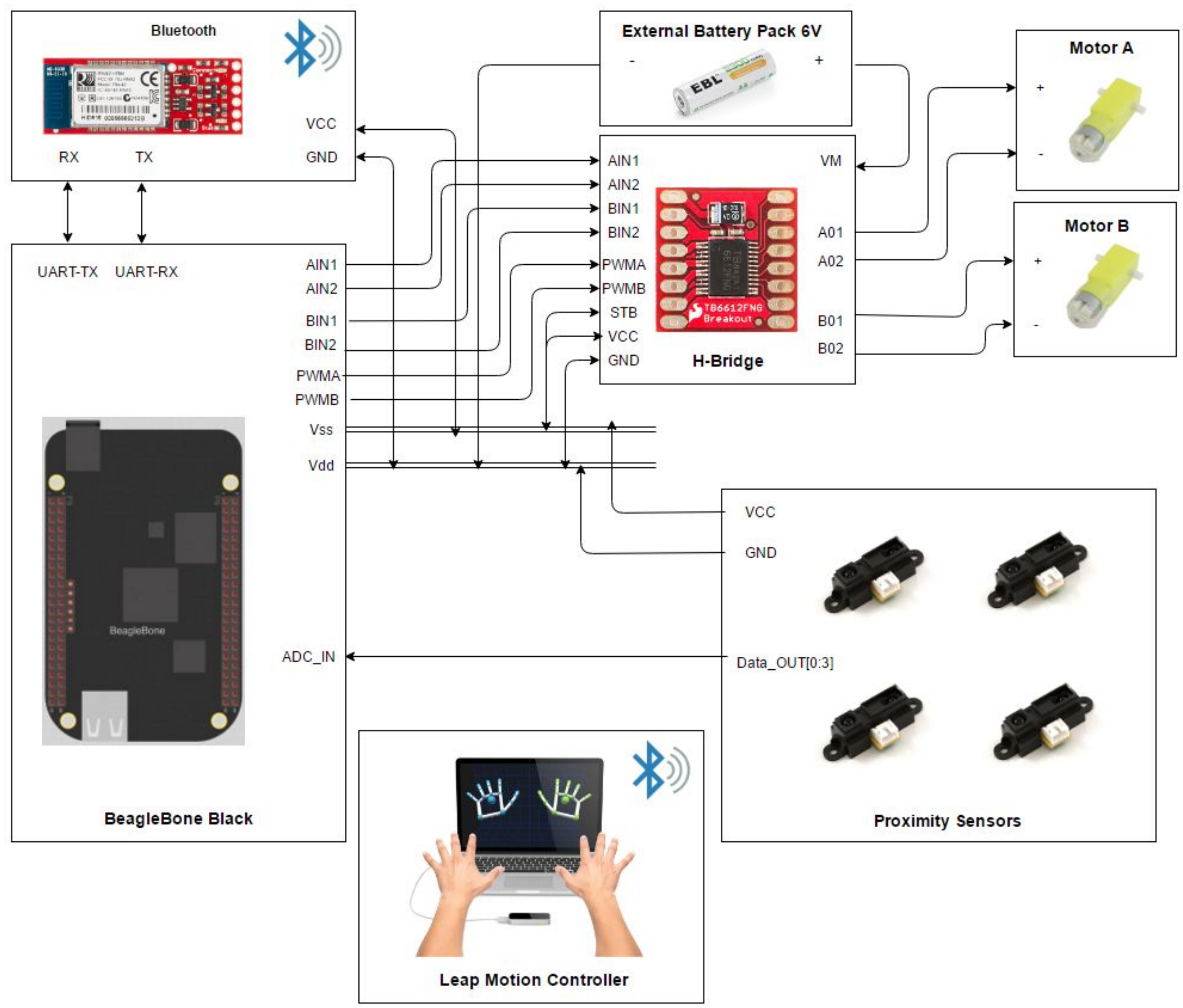

Figure 1: Hardware interconnection block diagram

SOFTWARE OVERVIEW

To properly control the tank, two executables were developed *motor* and *uart_drive*.

motor:

- Exports gpio and pwm pins and changes the directions with previously used functions: export_gpio_pins, export_pwm_pins, and change_dir. Duty cycles and frequencies of pwm pins are also configured
- Exports and samples ADC pins (proximity sensors)
- Creates a fifo that allows communication between the tank and the remote controller
- Remote Control Mode
    - In a while(1) loop constantly reads the fifo for user input. If the user specifies an action, prints it to the screen and calls a corresponding function: forward(...), backward(...), left(...), right(...), stop(...). In the action functions input pins of the H-bridges are toggled in different combinations to get the tank to move in desired directions
- Autonomous Mode
    - When the user selects autonomous mode, the tank starts to pilot itself. The timer interrupt is constantly triggered every 0.1 seconds to sample the proximity sensors. The outputs from the proximity sensors are fed back to the system in order for the master controller to decide the next action for the tank. For example, when the a sensor ADC value exceeds a certain preset threshold, the tank will stop and move to a different direction direction depending on if the sides are obstacle free.

uart_drive:

- Calls a python script UART_SETUP.py to setup UART4 for use on BeagleBone
- Opens the device (UART4) and modifies the settings of serial communication such as: baud rate, number of bits transferred, delays, control flow, and parity bit choice
- Writes the initialization string to Bluetooth (UART4) to enter the configuration mode and change the name of the device
- Uses while(1) loop to constantly read data from UART and write it to the pipe created by *motor*

To test the Bluetooth (UART4) communication an Arduino terminal app on an android phone was used to send commands through. In addition, a startup bash script was written to invoke both the aforementioned executables whenever the BeagleBone is powered up. This effectively allows the tank to be operating without being plugged into a computer.

ABOVE AND BEYOND

Our group used a Leap Motion Camera to capture hand gestures as controls to the motors. We defined 5 different hand gestures: forward, left, right, stop, and reverse. The Windows PC is first connected to the bluetooth module on the BeagleBone which opened up the COM3 port. The the python code is run, which does the following:

- Opens the COM3 serial port that is connected to the fifo buffer set up by the bluetooth server
- Initializes the Leap Motion Camera to grab images
- Wait until connection to Leap Motion Camera
- Main loop:
  - Grab a frame from the Leap Motion Camera
  - Find the closest hand in the frame
  - Use fist strength and palm positioning of the hand to determine the direction
  - Output that to the buffer
  - Wait 0.2 seconds before looping again