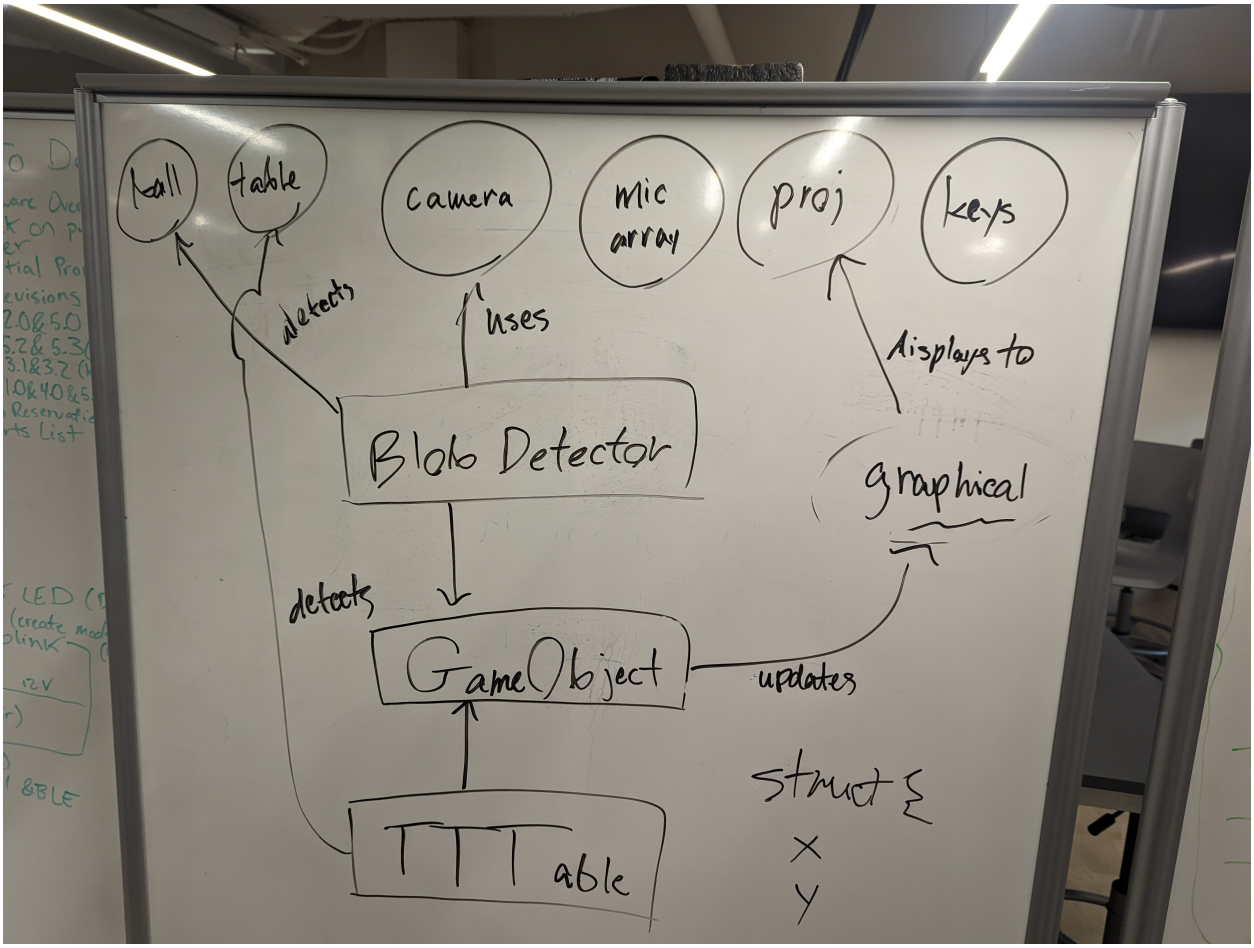
Week 6

This Week:

Game Logic:

This week James and I outlined the code structure for the laptop. Below is are plan. Each function is designated to it’s own object and interacts with the hardware accordingly.



Additionally, this week I upgraded the Uart code to be a class that can be instantiated and interacted with through wrapper functions.

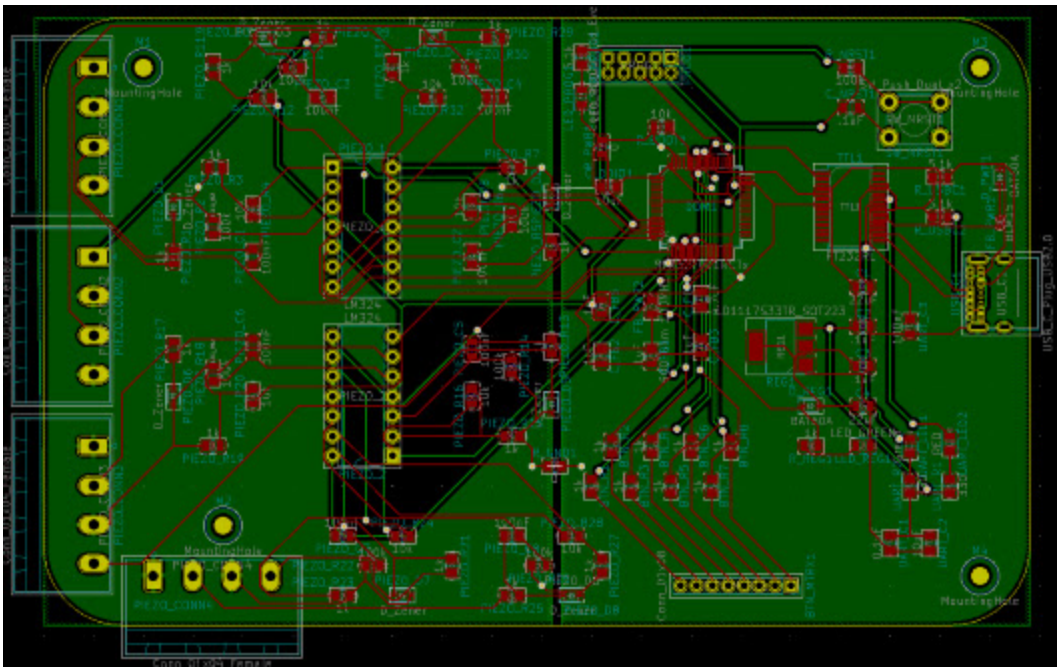
Below is the header file with a few defined interfaces:

```
#ifndef INCLUDE_UART_DECODER
#define INCLUDE_UART_DECODER
// MESSAGE MASKS
#define BUTTON_PRESS_MASK 0x10
#define BUTTON_SHIFT 0
#define BUTTON_MASK 0x0F
#define BOUNCE_MASK 0x60
#define BOUNCE_SHIFT 5
#define NONE -1

// LINUX TTY ASSOCIATED WITH SERIAL DEVICE
#define DEVICE "/dev/ttyUSB0"
#include <string.h>
class UartDecoder{
  public:
    int curr_bounce;
    int curr_press;
    int serial_port;
enum Button{
    ONE,
    TWO,
    THREE,
    FOUR,
    FIVE,
    SIX,
    SEVEN,
    EIGHT,
    NINE,
    A,
    B,
    C,
    D,
    E,
    F,
};

enum Bounce{
    LEFT=1,
    RIGHT,
};

UartDecoder(std::string&);
int decode(unsigned char message);
void closePort();
};
#endif
```

This week I also updated the decode function to incorporate the new message structure outlined in last week’s report.

PCB Design:

This week I also aided Bart with the layout of the PCB. Specifically the piezo filters. It was important to group each filter in it’s own cluster close to the amplifier IC:



PCB Design. Analog Filters on the left

I also ordered the final components to assemble our first PCB. The bill of materials is listed below.

Product	Quantity
2073-USB4085-GF-ACT-NDUSB4085-GF-ACONN RCPT USB2.0 TYPE C 16+8POS	4
BAT60BE6327HTSA1CT-NDBAT60BE6327HTSA1DIODE SCHOTTKY 10V 3A SOD323-2	10
553-2392-1-ND PE-0805FB601ST FERRITE BEAD 600 OHM 0805 1LN	4
399-9619-1-ND Z0805C330ASMST FERRITE BEAD 33 OHM 0805 1LN	6
MM3Z3V3ST1GOSCT-ND MM3Z3V3ST1G DIODE ZENER 3.3V 300MW SOD323	16
160-1423-1-ND LTST-C171GKT LED GREEN CLEAR CHIP SMD	3
1497-1450-1-ND XZM2CRK54WA-8 LED RED CLEAR CHIP SMD	3

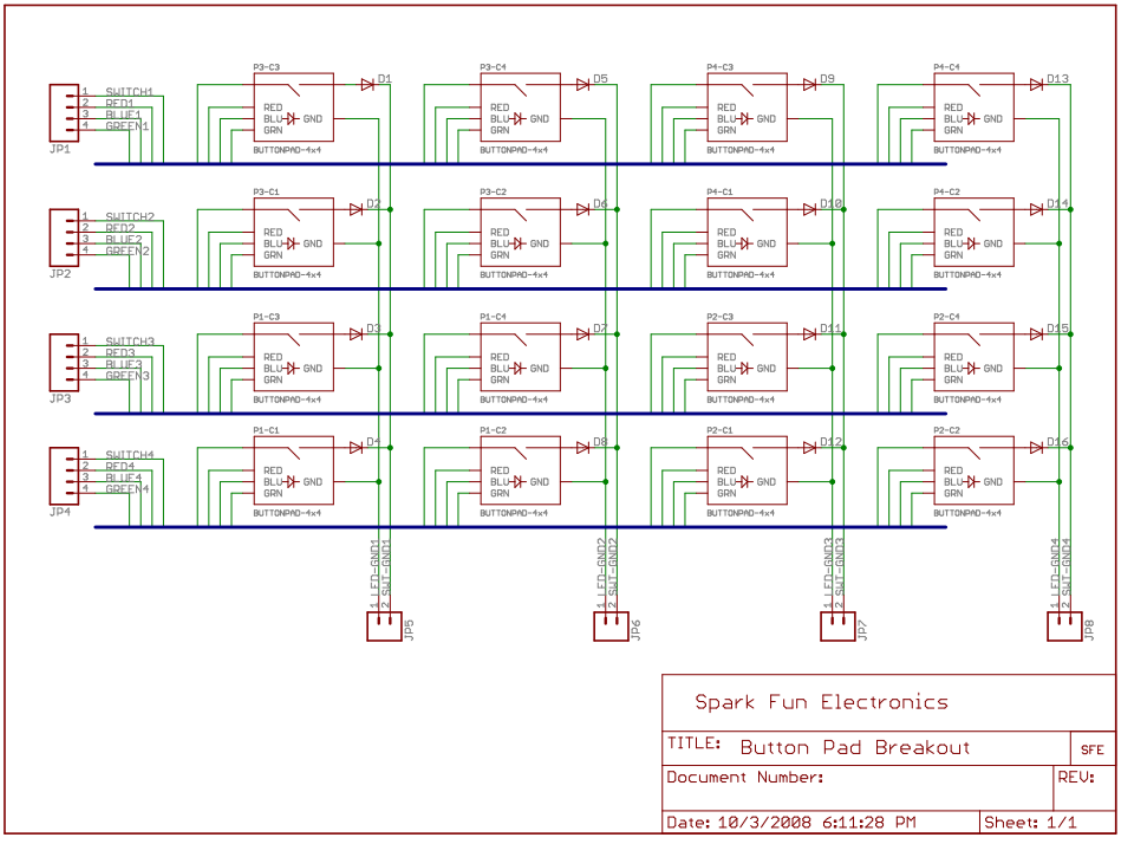
Plans for Next week:

Computer Vision:

As we are pivoting away from using the depth sensor, next week we will attempt to use the RGB sensor on our camera, and use the same smoothing and contour detection that we used with depth sensor.

Button Matrix:

Time permitting I will design a simple PCB for our own Button matrix, similar to the one below, minus the LEDs.



Projector:

This week we ordered a female to female converter for a vga cable so we can reach the projector mounted in the ceiling. Once we have this connection we will start to create the projector objects that will display to the device. We will also, in our game object,

develop the interface to use information from the microcontroller to inform the user interface.

PCB:

Next week I will help Bart finalize the PCB design before we send in our first order.

Uart:

Next week I will add additional interfaces to the UART object, to allow for getting the 8 bit message from the USB connection.