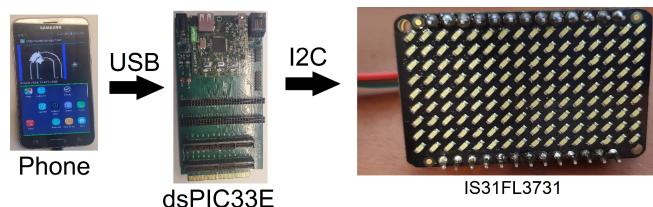


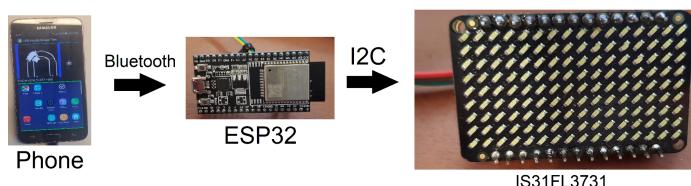
Progress So Far (Early April 2023)

Device

The device we were originally provided utilized a Microchip Technology dsPIC33E as its controller. This would control the IS31FL3731 used for the LED array over an I2C connection, and the controller would connect back to the phone using USB.



Due to the extremely limited documentation provided for the board, compiler options, and of what was provided for the previously written code, we were unable to compile and load software onto the original board. In addition, the I2C bus seemed to have issues on the board and would work intermittently. Due to these issues and for much lower cost, smaller size, ease of programming, and bluetooth functionality, we



switched to using an ESP32 Devkit C board. Generally, smartphones are more set up to use Bluetooth peripherals than USB peripherals, which is part of why we made this change. This change also allows us to write the program, compile it, and load it onto the device using [Arduino IDE](#), and use the large amount of existing libraries for Arduino.

```
#include "BluetoothSerial.h"
#include <dire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_IS31FL3731.h>

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif

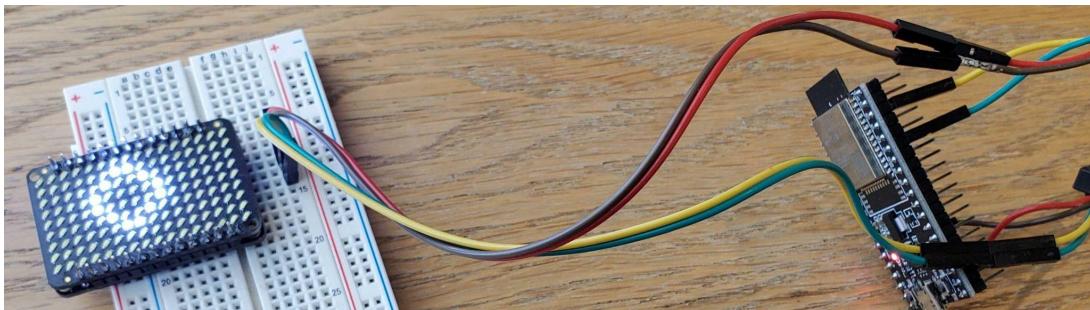
BluetoothSerial SerialBlue;
Adafruit_IS31FL3731 matrix = Adafruit_IS31FL3731();

uint8_t hapticsArray[6][6];
uint8_t serializedMatrix[36];
uint8_t currSerialIndex;

void initArrays(){
    for (int x = 0; x < 6; ++x) // init haptics array
    {
        for (int y = 0; y < 6; ++y)
        {

```

The device now uses a Bluetooth Classic connection to connect to devices to receive the data for the display. Serially, the device should receive 36 chars which the board will break into 6 rows of 6 for the display. The chars should be a ‘1’ for on or a ‘0’ for off. The display still connects over [I₂C](#) to the controller, but is much more reliable now.



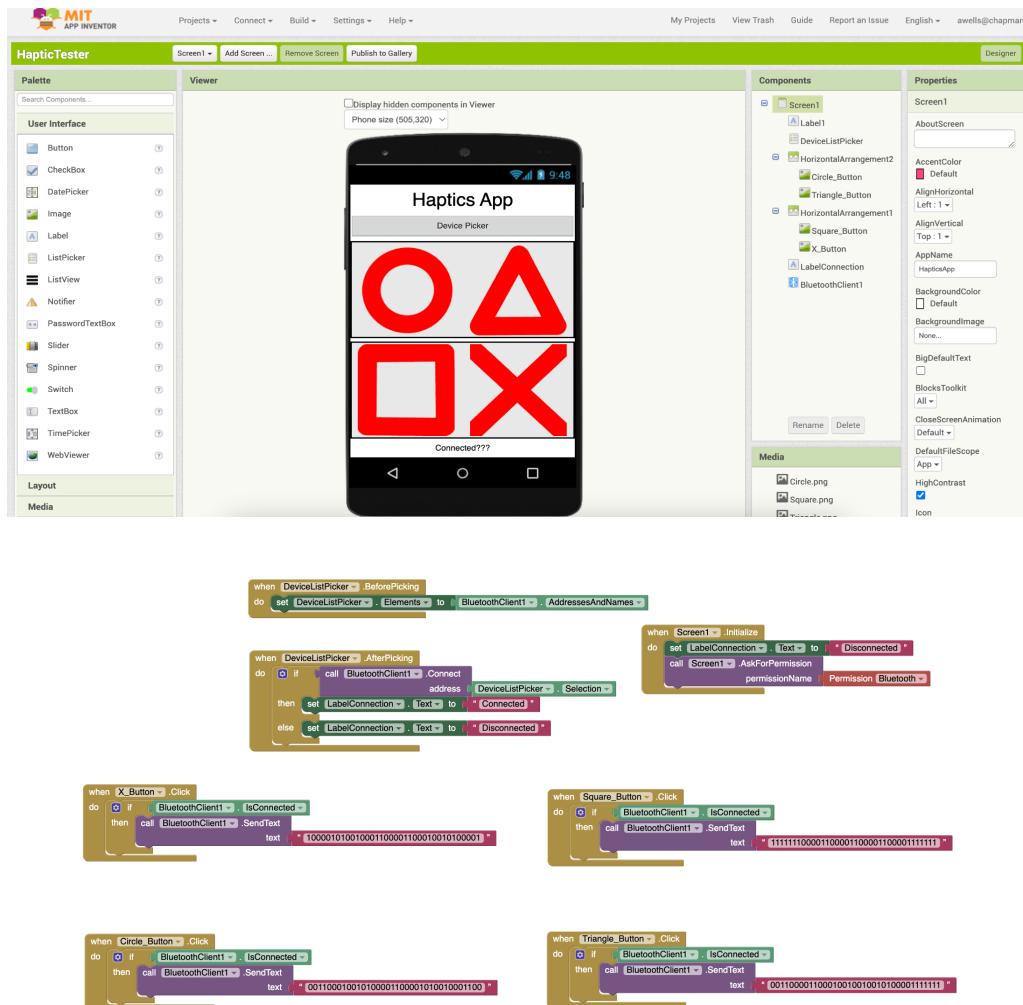
App

The original app was written in C# with Visual Studio using Xamarin. While this app worked with the original device on the original smartphone provided to us, it had several critical problems. Due to how long ago the code and its accompanying Visual Studio solution were created without updates, there were significant library and IDE compatibility issues, and a significant portion of the code utilized deprecated functionality. While we were ultimately able to compile the app and install it on another phone, it would not run due to the deprecated code. At this point, we deemed that creating a new application would likely be faster than fixing the issues with the existing

app, a significant portion of which was just the drivers for communication with the controller over USB.

To build the new app, we initially used Unity due to our familiarity with it. After spending some time working to develop the app this way, I decided to switch to another app development tool. While Unity can be a powerful app development tool, its support for using the phone's Bluetooth functionality is not great, even when using assets for the purpose.

Ultimately, we developed the current app using MIT's [App Inventor](#) tool. This tool is meant for quickly developing small apps for education and research purposes, and most importantly has built-in support the phone's different hardware functionalities including Bluetooth.



The current app displays for different shapes on the screen, and clicking a shape will send it to the device to be displayed. Currently, one must go into the phone's permissions settings and manually give the app Bluetooth permissions. Once this is done, pair the phone with the device (this will appear as Haptic32) in the Bluetooth settings of the phone. Next, open the app, click the device picker, and choose the device. The app should now display that it is connected at the bottom.

Conference

We have submitted a short paper on this research and were accepted for the [IARIA Smart Accessibility 2023 conference](#), where we will be submitting a prerecorded video on our research to be presented alongside our paper.