

```
/******
```

```
*Linh Do
```

```
*Final Project-Wearable Technology 01
```

```
*Fall 2012
```

```
*
```

This is an example sketch for the Adafruit 1.8" TFT shield with joystick  
----> <http://www.adafruit.com/products/802>

Check out the links above for our tutorials and wiring diagrams  
These displays use SPI to communicate, 4 pins are required to interface

One pin is also needed for the joystick, we use analog 3  
Adafruit invests time and resources providing this open source code,  
please support Adafruit and open-source hardware by purchasing  
products from Adafruit!

Written by Limor Fried/Ladyada for Adafruit Industries.

MIT license, all text above must be included in any redistribution

```
*****/
```

```
#include <Adafruit_GFX.h>
```

```
#include <Adafruit_ST7735.h>
```

```
#include <SD.h>
```

```
#include <SPI.h>
```

```
// TFT display and SD card will share the hardware SPI interface.
```

```
// Hardware SPI pins are specific to the Arduino board type and
```

```
// cannot be remapped to alternate pins. For Arduino Uno,
```

```
// Duemilanove, etc., pin 11 = MOSI, pin 12 = MISO, pin 13 = SCK.
```

```
#define SD_CS 4 // Chip select line for SD card
```

```
#define TFT_CS 10 // Chip select line for TFT display
```

```
#define TFT_DC 8 // Data/command line for TFT
```

```
#define TFT_RST -1 // Reset line for TFT (or connect to +5V)
```

```
Adafruit_ST7735 tft = Adafruit_ST7735(TFT_CS, TFT_DC, TFT_RST);
```

```
#define BUTTON_NONE 0
```

```
#define BUTTON_DOWN 1
```

```
#define BUTTON_RIGHT 2
```

```
#define BUTTON_SELECT 3
```

```
#define BUTTON_UP 4
```

```
#define BUTTON_LEFT 5
```

```
void setup(void) {
```

```
  Serial.begin(9600);
```

```

// Our supplier changed the 1.8" display slightly after Jan 10, 2012
// so that the alignment of the TFT had to be shifted by a few pixels
// this just means the init code is slightly different. Check the
// color of the tab to see which init code to try. If the display is
// cut off or has extra 'random' pixels on the top & left, try the
// other option!

// If your TFT's plastic wrap has a Red Tab, use the following:
tft.initR(INITR_REDTAB); // initialize a ST7735R chip, red tab
// If your TFT's plastic wrap has a Green Tab, use the following:
//tft.initR(INITR_GREENTAB); // initialize a ST7735R chip, green tab

// Serial.println("OK!");
tft.fillScreen(tft.Color565(174,223,228));
/*
  Serial.print("Initializing SD card...");
  if (!SD.begin(SD_CS)) {
    Serial.println("failed!");
    return;
  }
*/
}

uint8_t readButton(void) {
  float a = analogRead(3);

  a *= 5.0;
  a /= 1024.0;

  Serial.print("Button read analog = ");
  Serial.println(a);
  if (a < 0.2) return BUTTON_DOWN;
  if (a < 1.0) return BUTTON_RIGHT;
  if (a < 1.5) return BUTTON_SELECT;
  if (a < 2.0) return BUTTON_UP;
  if (a < 3.2) return BUTTON_LEFT;
  else return BUTTON_NONE;
}

uint8_t buttonhistory = 0;

void loop() {
  uint8_t b = readButton();
  tft.setTextSize(1);

```

```

tft.setTextColor(ST7735_WHITE);

int sensorValue0 = analogRead(A0);
int sensorValue1 = analogRead(A1);
int sensorValue2 = analogRead(A2);
int sensorValue3 = analogRead(A3);
int sensorValue4 = analogRead(A4);
int sensorValue5 = analogRead(A5);
int mappedSensorValue0 = map (sensorValue0, 0, 1023, 0, 10);
int mappedSensorValue1 = map (sensorValue1, 0, 1023, 0, 10);
int mappedSensorValue2 = map (sensorValue2, 0, 1023, 0, 10);
int mappedSensorValue3 = map (sensorValue3, 0, 1023, 0, 10);
int mappedSensorValue4 = map (sensorValue4, 0, 1023, 0, 10);
int mappedSensorValue5 = map (sensorValue5, 0, 1023, 0, 10);

// tft.print (sensorValue1);

// tft.print (mappedSensorValue1);

if (mappedSensorValue0<5){

    // right
    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

    tft.drawCircle(46, 65, 5, ST7735_BLACK);
    tft.fillCircle(46,65,5,ST7735_BLACK);

    tft.drawCircle(98, 65, 5, ST7735_BLACK);
    tft.fillCircle(98,65,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);

    delay(1000);

}

```

```

else if (sensorValue1>220) {
  //hot
  tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
  tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

  tft.drawCircle(38, 65, 5, ST7735_BLACK);
  tft.drawCircle(90, 65, 5, ST7735_BLACK);

  tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
  tft.fillCircle(34,80,5,tft.Color565(247,165,170));

  tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
  tft.fillCircle(94,80,5,tft.Color565(247,165,170));

  tft.drawLine(54,75, 74, 75, ST7735_BLACK);

  delay(1000);
  }

else if (sensorValue1<100) {
  //cold
  tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
  tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

  tft.drawCircle(38, 65, 5, ST7735_BLACK);
  tft.drawCircle(90, 65, 5, ST7735_BLACK);

  tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
  tft.fillCircle(34,80,5,tft.Color565(247,165,170));

  tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
  tft.fillCircle(94,80,5,tft.Color565(247,165,170));

  tft.drawLine(54,75, 74, 75, ST7735_BLACK);

  delay(1000);
  }

else if (mappedSensorValue2<5) {

  //left
  tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
  tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
  tft.drawCircle(30, 65, 5, ST7735_BLACK);
  tft.fillCircle(30,65,5,ST7735_BLACK);

```

```

tft.drawCircle(82, 65, 5, ST7735_BLACK);
tft.fillCircle(82,65,5,ST7735_BLACK);

tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(34,80,5,tft.Color565(247,165,170));

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);

delay(1000);

    }

else if (mappedSensorValue4<5) {

    //smiling face

tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

tft.drawCircle(38, 65, 5, ST7735_BLACK);

tft.drawCircle(90, 65, 5, ST7735_BLACK);

tft.drawRect(32, 65,65,6, tft.Color565(174,223,228));
tft.fillRect(32, 65,65,6, tft.Color565(174,223,228));

tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(34,80,5,tft.Color565(247,165,170));

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);

delay(1000);

    }

else if (mappedSensorValue5>0) {

    //scared face
tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));

```

```

tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

tft.drawLine(35,60, 40, 65, ST7735_BLACK);
tft.drawLine(35,70, 40, 65, ST7735_BLACK);

tft.drawLine(93,60, 88, 65, ST7735_BLACK);
tft.drawLine(93,70, 88, 65, ST7735_BLACK);

tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(34,80,5,tft.Color565(247,165,170));

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);

delay (1000);

    }
else {

    //normal face

tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

tft.drawCircle(38, 65, 5, ST7735_BLACK);
tft.fillCircle(38,65,5,ST7735_BLACK);

tft.drawCircle(90, 65, 5, ST7735_BLACK);
tft.fillCircle(90,65,5,ST7735_BLACK);

tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(34,80,5,tft.Color565(247,165,170));

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);

delay(1000);

}

```

```

/*

// example 1 - light on off
if (sensorValue>350) {
// Serial.println("Lights are on!");
// digitalWrite (13,LOW);
    // tft.fillScreen(tft.Color565(174,223,228));

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));

    tft.drawCircle(38, 65, 5, ST7735_BLACK);
    tft.fillCircle(38,65,5,ST7735_BLACK);
// tft.fillScreen(tft.Color565(174,223,228));

    tft.drawCircle(90, 65, 5, ST7735_BLACK);
    tft.fillCircle(90,65,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);

    delay(2000);

    if (b == BUTTON_DOWN) {
// buttonhistory |= 1;
// tft.setCursor(0, 65);
// tft.print("          SNOW");
// delay(1000);
// tft.setCursor(0, 85);
// tft.print("a story by Sethaniel from choose your story dot com");
// delay(2000);
// tft.setCursor(0, 115);
// tft.print("developed by Linh Do 2012");
// delay(1000);
// bmpDraw("parrot.bmp", 0, 0);
// tft.setCursor(0, 115);
// tft.print("pressed any button to continue");

// tft.drawPixel(tft.width()/2, tft.height()/2, ST7735_GREEN);
// delay(500);
// testfillcircles(10, ST7735_BLUE);

```

```

// testdrawcircles(10, ST7735_WHITE);
// delay(500);

// testroundrects();
// delay(500);

// testtriangles();
// delay(500);

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
//tft.fillScreen(tft.Color565(174,223,228));
//    tft.drawCircle(38, 50, 5, ST7735_BLACK);
//    tft.fillCircle(38,50,5,ST7735_BLACK);
    tft.drawLine(35,70, 41, 70, ST7735_BLACK);

//    tft.drawCircle(90, 50, 5, ST7735_BLACK);
//    tft.fillCircle(90,50,5,ST7735_BLACK);
    tft.drawLine(87,70, 93, 70, ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);
        delay(2000);
// }

}

if (b == BUTTON_LEFT) {
//    tft.fillScreen(tft.Color565(174,223,228));

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.drawCircle(30, 65, 5, ST7735_BLACK);
    tft.fillCircle(30,65,5,ST7735_BLACK);
// tft.fillScreen(tft.Color565(174,223,228));

    tft.drawCircle(82, 65, 5, ST7735_BLACK);
    tft.fillCircle(82,65,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));
}

```



```

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);

delay(2000);
}

if (b == BUTTON_UP) {

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
    // tft.fillScreen(tft.Color565(174,223,228));
    tft.drawCircle(38, 60, 5, ST7735_BLACK);
    tft.fillCircle(38,60,5,ST7735_BLACK);
// tft.fillScreen(tft.Color565(174,223,228));

    tft.drawCircle(90, 60, 5, ST7735_BLACK);
    tft.fillCircle(90,60,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);

        delay(2000);
}

if (b == BUTTON_RIGHT) {

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
    // tft.fillScreen(tft.Color565(174,223,228));
    tft.drawCircle(46, 65, 5, ST7735_BLACK);
    tft.fillCircle(46,65,5,ST7735_BLACK);
// tft.fillScreen(tft.Color565(174,223,228));

    tft.drawCircle(98, 65, 5, ST7735_BLACK);
    tft.fillCircle(98,65,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));

```

```

    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);
        delay(2000);
}

if (b == BUTTON_SELECT) {

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
    // tft.fillScreen(tft.Color565(174,223,228));
    tft.drawCircle(38, 65, 5, ST7735_BLACK);
    tft.fillCircle(38,65,5,ST7735_BLACK);
// tft.fillScreen(tft.Color565(174,223,228));

    tft.drawCircle(90, 65, 5, ST7735_BLACK);
    tft.fillCircle(90,65,5,ST7735_BLACK);

    tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(34,80,5,tft.Color565(247,165,170));

    tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
    tft.fillCircle(94,80,5,tft.Color565(247,165,170));

    tft.drawLine(54,75, 74, 75, ST7735_BLACK);
        delay(2000);
}

}

else if (sensorValue < 350) {
    Serial.println("Lights are off!");
//    digitalWrite (13,HIGH);
//tft.fillScreen(tft.Color565(174,223,228));

    tft.drawRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.fillRect(20, 43,100,30, tft.Color565(174,223,228));
    tft.drawCircle(38, 50, 5, ST7735_BLACK);
//    tft.fillCircle(38,50,5,ST7735_BLACK);
//    tft.drawLine(35,70, 41, 70, ST7735_BLACK);

    tft.drawCircle(90, 50, 5, ST7735_BLACK);

```

```
// tft.fillCircle(90,50,5,ST7735_BLACK);
// tft.drawLine(87,70, 93, 70, ST7735_BLACK);

tft.drawCircle(34, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(34,80,5,tft.Color565(247,165,170));

tft.drawCircle(94, 80, 5, tft.Color565(247,165,170));
tft.fillCircle(94,80,5,tft.Color565(247,165,170));

tft.drawLine(54,75, 74, 75, ST7735_BLACK);
    delay(2000);

}
*/
// delay(100);
}
```