

Advanced Prototyping Final Study Sheet

Switch Cases

```
int mode = 0;
int button = 2;

void setup() {
  Serial.begin(9600);
  pinMode(button, INPUT);
}

void loop() {
  checkButton();

  switch (mode) {
    case 0: splash(); break;
    case 1: mainMenu(); break;
    case 2: about(); break;
  }
}

void checkButton(){
  int raw = digitalRead(button);
  if(raw == 1){
    mode = (mode + 1) % 3;
    delay(250);
  }
}

void splash() {
  Serial.println("Splash");
}

void mainMenu() {
  Serial.println("Main Menu");
}

void about() {
  Serial.println("About");
}
```

Screen Elements

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 4
Adafruit_SSD1306
display(OLED_RESET);

const unsigned char umbrella []
PROGRAMMEM = {
  0x00, 0x01, 0x80, 0x00, 0x00, 0x00,
  0x01, 0x80, 0x00, 0x00, 0x01,
  0x80, 0x00, 0x00, 0x01, 0x80,
  0x00,
  0x00, 0x01, 0xf0, 0x00, 0x00, 0x00,
  0xff, 0xfe, 0x00, 0x01, 0xff, 0xff,
  0x80, 0x07, 0xff, 0xff, 0xc0,
  0x0f, 0xff, 0x0f, 0xe0, 0x1f, 0xff,
  0xff, 0xff, 0x3f, 0xff, 0xff, 0xf8,
  0x3f, 0xff, 0xff, 0xff,
  0x7f, 0xff, 0xff, 0xff, 0x7f, 0xff,
  0xff, 0xff, 0x7f, 0xff, 0xff, 0xfe,
  0xff, 0xff, 0xff, 0xf0,
  0xe0, 0xf8, 0xfe, 0x06, 0x06,
  0x31, 0x88, 0x00, 0x00, 0x01,
  0x80, 0x00, 0x00, 0x01, 0x80,
  0x00,
  0x00, 0x01, 0x80, 0x00, 0x00, 0x00,
  0x01, 0x80, 0x00, 0x00, 0x01,
  0x80, 0x00, 0x00, 0x01, 0x80,
  0x00,
  0x00, 0x01, 0x80, 0x00, 0x00, 0x00,
  0x81, 0x80, 0x00, 0x01, 0xc1,
  0x80, 0x00, 0x00, 0x01, 0xc3, 0x80,
  0x00,
  0x00, 0xc3, 0x80, 0x00, 0x00,
  0xff, 0x00, 0x00, 0x00, 0x7e,
  0x00, 0x00, 0x00, 0x08, 0x00,
  0x00,
  };

void setup() {
  Serial.begin(9600);

  display.begin(SSD1306_SWITCH
CAPVCC, 3); // Clear the
buffer.

  display.clearDisplay();
  display.display();

}

void loop() {
  display.drawBitmap(0,0,
umbrella, 32, 32, WHITE);
  // display.drawPixel(10, 10,
WHITE);
  // display.drawCircle(80, 20, 10
WHITE);
  // display.fillCircle(80, 20, 5,
WHITE);
  // display.setCursor(25, 50);
  // display.setTextFont(WHITE,
BLACK);
  // display.setTextSize(1);
  // display.print("Hello World");
  display.display();
}
```

Map and Constrain

```
int led = 5;
int flexled;

void setup() {
  // put your setup code here, to
  run once:
  Serial.begin(9600);
  pinMode(led, OUTPUT);
}

void loop() {
  // put your main code here, to
  run repeatedly:
  int flex = analogRead(A1);
  Serial.println(flex);
  flexled = map(flex, 0, 500, 0,
  255);
  flexled = constrain(flexled, 0,
  255);
  analogWrite(led, flexled);
  delay(10);
}
```

Sensor Reader

```
int led = 5;
int flexled;

void setup() {
  // put your setup code here, to
  // run once:
  Serial.begin(9600);
  pinMode(led, OUTPUT);
}

void loop() {
  // put your main code here, to
  // run repeatedly:
  int flex = analogRead(A1);
  Serial.println(flex);
  flexled = map(flex, 100, 400, 0,
  255);
  analogWrite(led, flexled);
  delay(10);
}
```

Sensor Reader (temp)

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire
oneWire(ONE_WIRE_BUS);
DallasTemperature
sensors(&oneWire);

void setup(){
  Serial.begin(9600);
  sensors.begin();
}

void loop() {
  sensors.requestTemperatures();
  float c =
  sensors.getTempCByindex(0);
  float f = (c * (9.0/5.0)) + 32;
  Serial.println(f);
  delay(1000);
}
```

Image to Cpp link

<https://javl.github.io/image2cpp>

NeoPixel Code

```
#include <Adafruit_NeoPixel.h>

#define PIN 6
#define LED_COUNT 3

Adafruit_NeoPixel leds = Adafruit_NeoPixel(LED_COUNT, PIN, NEO_GRB + NEO_KHZ800);

\\

void setup()
{
  leds.begin();
  leds.show();
}

void loop()
{
  leds.setPixelColor(0,255,0,0);
  leds.setPixelColor(1,0,255,0);
  leds.setPixelColor(2,0,0,255);
  leds.show();
}

\\or

void setup()
{
  leds.begin();
  leds.show();
}

void loop()
{
  for(int x=0; x<LED_COUNT; x++)
  {
    leds.setPixelColor(x,255,255,0);
  }
  leds.show();
}
```

