Proje Raporu: Ele Giyilebilir Mouse

**Giriş**

Bu projemizde, mikrodenetleyici dersi kapsamında ele giyilebilir bir mouse tasarlayıp gerçekleştirdik. Projemizde ESP32CAM, MPU6050, USB-TTL programlayıcı, 2 adet buton, mini breadboard ve jumper kablolar kullandık. ESP32CAM ile Bluetooth üzerinden bir bilgisayara bağlanarak, MPU6050 sensöründen gelen hareket verilerini kullanarak fare hareketlerini taklit ettik. Bu raporda projemizin detayları, kullanılan malzemeler, yazılım geliştirme süreci ve karşılaşılan zorlukları aşağıda anlatacağız.

**Kullanılan Malzemeler**

ESP32CAM: Bu modül hem Wi-Fi hem de Bluetooth özelliklerine sahip bir mikrodenetleyicidir.

MPU6050: İvmeölçer ve jiroskop sensörleri içeren, hareket ve ivme verilerini ölçmek için kullanılan bir modüldür.

USB-TTL Programlayıcı: ESP32CAM'i programlamak için kullanılan bir ara birimdir.

Butonlar: Sol ve sağ tıklama işlevleri için iki adet buton kullandık.

Mini Breadboard: Devre bağlantılarını düzenlemek için kullanıldı.

Jumper Kablolar: Modüller arasındaki bağlantıları sağlamak için kullanıldı.

**Devre Şeması**



Devreyi oluşturma sürecimiz:

MPU6050:

VCC - 5.0V (ESP32CAM)

GND - GND (ESP32CAM)

SCL - GPIO 13 (ESP32CAM)

SDA - GPIO 12 (ESP32CAM)

Buton 1:

Bir ucu GPIO 15'e, diğer ucu GND'ye bağlandı.

Buton 2:

Bir ucu GPIO 14'e, diğer ucu GND'ye bağlandı.

USB-TTL Programlayıcı:

RX - TX0 (ESP32CAM)

TX - RX0 (ESP32CAM)

GND - GND (ESP32CAM)

5V - 5V (ESP32CAM)

**Yazılım Geliştirme Süreci**

Projemizde Arduino IDE'yi kullanarak ESP32CAM için kod yazdık. Kodumuzda öncelikle gerekli kütüphaneleri dahil ettik:

#include <Wire.h>

#include <Adafruit\_MPU6050.h>

#include <Adafruit\_Sensor.h>

#include <BleMouse.h>

MPU6050 sensörünü ve BLE Mouse (Bluetooth Low Energy Mouse) nesnelerini tanımladık

Adafruit\_MPU6050 mpu;

BleMouse bleMouse("ESP32 Mouse");

Buton pinlerini ve son durum değişkenlerini tanımladık:

const int button1Pin = 15;

const int button2Pin = 14;

int lastButton1State = HIGH;

int lastButton2State = HIGH;

setup() fonksiyonu içerisinde, sensör ve BLE mouse başlatma işlemlerini gerçekleştirdik:

void setup() {

Serial.begin(115200);

Serial.println("Starting BLE work!");

Wire.begin(13, 12);

pinMode(button1Pin, INPUT\_PULLUP);

pinMode(button2Pin, INPUT\_PULLUP);

if (!mpu.begin()) {

Serial.println("Failed to find MPU6050 chip");

while (1) {

delay(10);

}

}

Serial.println("MPU6050 Found!");

mpu.setAccelerometerRange(MPU6050\_RANGE\_2\_G);

mpu.setGyroRange(MPU6050\_RANGE\_250\_DEG);

mpu.setFilterBandwidth(MPU6050\_BAND\_21\_HZ);

bleMouse.begin();

Serial.println("Bluetooth cihazı başladı: ESP32 Mouse");

}

loop() fonksiyonu içerisinde, sensörden okunan verileri işleyerek mouse hareketlerini ve tıklama işlemlerini gerçekleştirdik:

void loop() {

static unsigned long lastUpdate = 0;

unsigned long currentMillis = millis();

if (bleMouse.isConnected()) {

if (currentMillis - lastUpdate >= 20) {

lastUpdate = currentMillis;

sensors\_event\_t a, g, temp;

mpu.getEvent(&a, &g, &temp);

int deltaX = a.acceleration.y \* 10;

int deltaY = a.acceleration.x \* 10;

int button1State = digitalRead(button1Pin);

int button2State = digitalRead(button2Pin);

if (button1State != lastButton1State) {

if (button1State == LOW) {

bleMouse.press(MOUSE\_LEFT);

} else {

bleMouse.release(MOUSE\_LEFT);

}

lastButton1State = button1State;

}

if (button2State != lastButton2State) {

if (button2State == LOW) {

bleMouse.press(MOUSE\_RIGHT);

} else {

bleMouse.release(MOUSE\_RIGHT);

}

lastButton2State = button2State;

}

bleMouse.move(deltaX, deltaY);

Serial.print("X: ");

Serial.print(deltaX);

Serial.print(" Y: ");

Serial.println(deltaY);

}

} else {

Serial.println("Mouse not connected!");

}

}

**Sonuçlar ve Değerlendirme**

Projemiz başarılı bir şekilde tamamlandı ve ESP32CAM ile MPU6050 kullanarak bir ele giyilebilir mouse oluşturduk. Bu proje sayesinde Bluetooth teknolojisi ve sensör verilerini kullanarak cihazlar arası etkileşimin nasıl gerçekleştirileceğini öğrendik. Ayrıca, butonların tıklama işlevlerini programlayarak interaktif bir kullanıcı deneyimi sağladık.

**Karşılaştığımız Zorluklar**

Bağlantı Sorunları: MPU6050 sensöründen doğru verileri almak için doğru bağlantıları sağlamak ve I2C iletişimini sorunsuz hale getirmek başlangıçta biraz zorlayıcı oldu.

Hareket Hassasiyeti: İvmeölçer ve jiroskop verilerini fare hareketlerine dönüştürürken hassasiyeti ayarlamak gerekti. Bu yüzden bazı ayarlamalar yaparak uygun çarpanları belirledik.

USB-TTL Programlayıcıyla Yaşanan Sorunlar: USB-TTL programlayıcı, kod yüklemesi sırasında istikrarsız davranışlar sergiledi. Bağlantı kesintileri ve hatalı veri iletimi gibi sorunlarla karşılaşıldı. Bu sorunları çözmek için USB-TTL kabloların bağlantılarını kontrol ettik.

**Gelecekte yapmayı planladığımız Çalışmamızı geliştirmek için izleyeceğimiz bazı adımlar**

Öğretmenlerimizden aldığımız çok olumlu tepkiler sayesinde motivasyonumuzun çok arttı ve projemizi daha da geliştirmek için çalışmalar yapmaya karar verdik.

Ek Butonlar ve Fonksiyonlar: Daha fazla buton eklenerek farklı fonksiyonlar atanabilir.

Daha iyi bir dış görünüm için 3d tasarımlar.

Kullanıcı Arayüzü Geliştirme: Bir mobil uygulama veya masaüstü uygulaması ile fare ayarlarının kullanıcılar tarafından yapılması sağlanabilir.

**Sonuç**

Projemiz Ele giyilebilir mouse, hareket algılama ve Bluetooth iletişimi konularında bilgi ve deneyim kazanmamıza yardımcı oldu. pratik olarak deneyim kazandık ve takım çalışması ,sorun çözme gibi çok öenmli konularda kendimizi geliştirdik.