Overview

This project demonstrates how an **IR sensor** can be used with a **CH32V003 microcontroller** to detect objects and control an **LED** accordingly. When the sensor detects an obstacle, the LED turns ON; otherwise, it remains OFF.

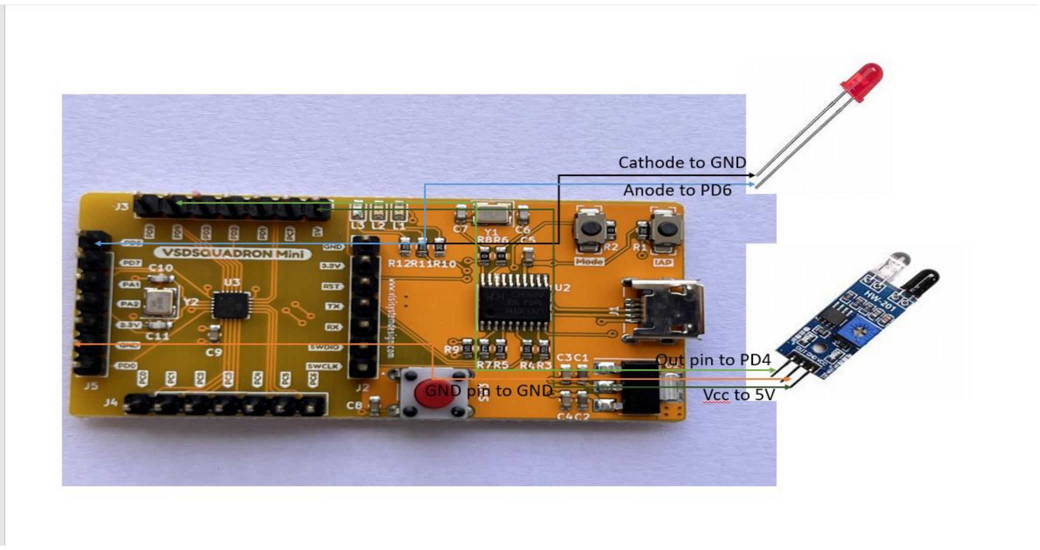
Objectives

- To interface an **IR sensor** with the **CH32V003** microcontroller.
- To control an **LED** based on IR sensor detection.
- To demonstrate an object detection system using an IR sensor.

Components Required

components	quantity
microcontroller	1
IR SENSOR	1
LED	1

Circuit Diagram



Circuit Connections

IR Sensor Connections

IR Sensor Pin	CH32V003 Pin	Description
VCC	5V	Power supply to IR sensor
GND	GND	Ground connection
OUT	PD4	Sensor output

LED Connections

LED Pin	CH32V003 Pin	Description
Anode	PD4	LED control output
Cathode	GND	completes circuit

CODE:

```
#include <ch32v00x.h>
#include <debug.h>

#define IR_SENSOR_GPIO_PORT GPIOC
#define IR_SENSOR_GPIO_PIN GPIO_Pin_4
#define IR_SENSOR_CLOCK_ENABLE
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOC, ENABLE)

#define LED_GPIO_PORT GPIOD
#define LED_GPIO_PIN GPIO_Pin_6
#define LED_CLOCK_ENABLE RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOD,
ENABLE)

void NMI_Handler(void) __attribute__((interrupt("WCH-Interrupt-fast")));
void HardFault_Handler(void) __attribute__((interrupt("WCH-Interrupt-fast")));
void Delay_Init(void);
void Delay_Ms(uint32_t n);

int main(void)
{
   NVIC_PriorityGroupConfig(NVIC_PriorityGroup_1);
   SystemCoreClockUpdate();
   Delay_Init();
```

```
    GPIO_InitTypeDef GPIO_InitStructure = {0};

    // Enable clocks for LED and IR sensor GPIO ports
    LED_CLOCK_ENABLE;
    IR_SENSOR_CLOCK_ENABLE;

    // Configure LED GPIO as output
    GPIO_InitStructure.GPIO_Pin = LED_GPIO_PIN;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP;
    GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
    GPIO_Init(LED_GPIO_PORT, &GPIO_InitStructure);

    // Configure IR sensor GPIO as input
    GPIO_InitStructure.GPIO_Pin = IR_SENSOR_GPIO_PIN;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_IN_FLOATING;
    GPIO_Init(IR_SENSOR_GPIO_PORT, &GPIO_InitStructure);

    while (1)
    {
        // Read IR sensor state
        if (GPIO_ReadInputDataBit(IR_SENSOR_GPIO_PORT, IR_SENSOR_GPIO_PIN))
        {
            // IR sensor detected something → Turn LED OFF
            GPIO_ResetBits(LED_GPIO_PORT, LED_GPIO_PIN);
        }
        else
        {
            // No detection → Turn LED ON
            GPIO_SetBits(LED_GPIO_PORT, LED_GPIO_PIN);
        }

        Delay_Ms(100); // Small delay to avoid bouncing issues
    }
}

void NMI_Handler(void) {}
void HardFault_Handler(void)
{
    while (1)
    {
    }
}
```