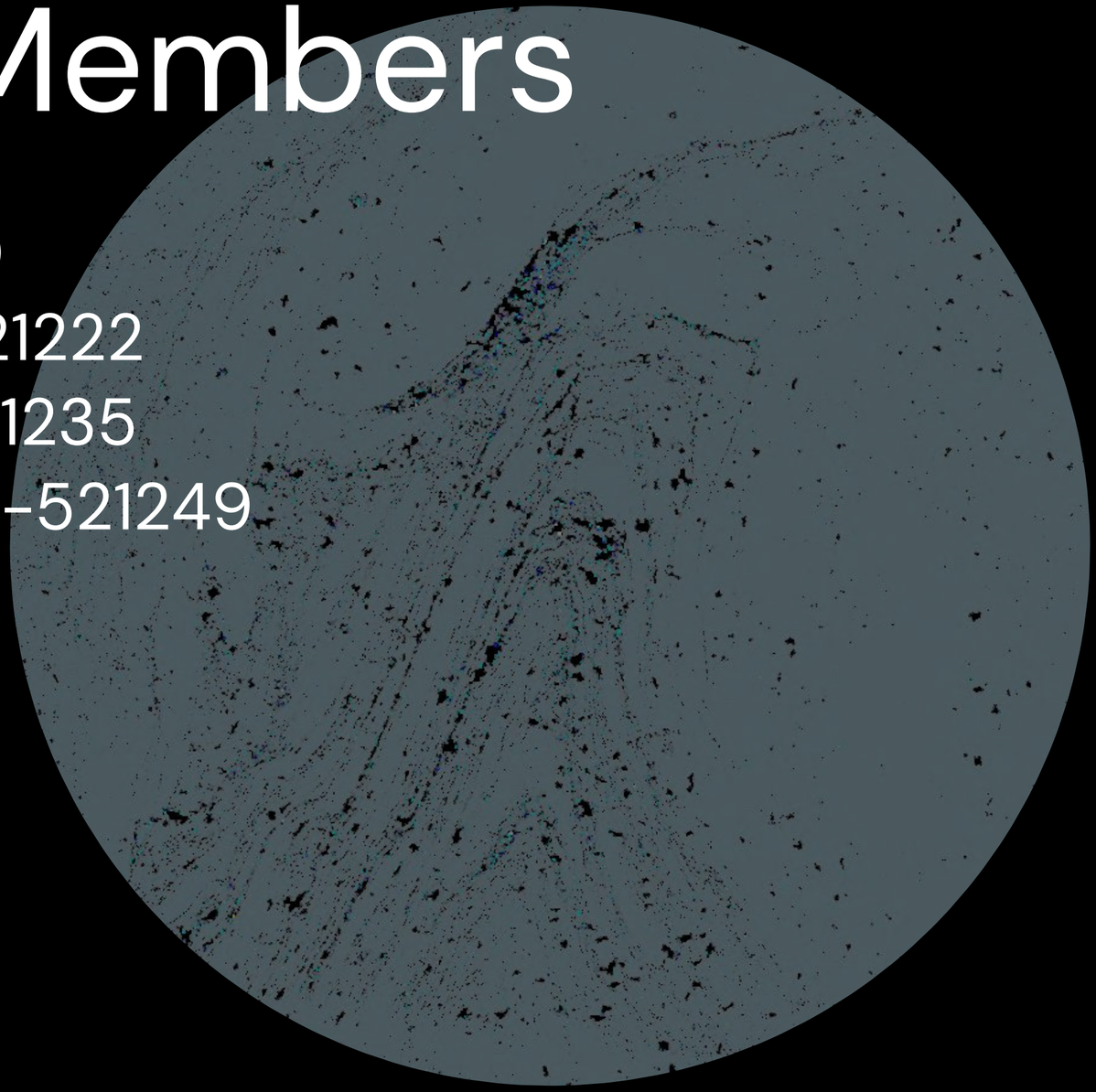
National Institute of Technology

Andhra Pradesh

EE301 MINOR PROJECT

Smart Car Parking System

Group Members



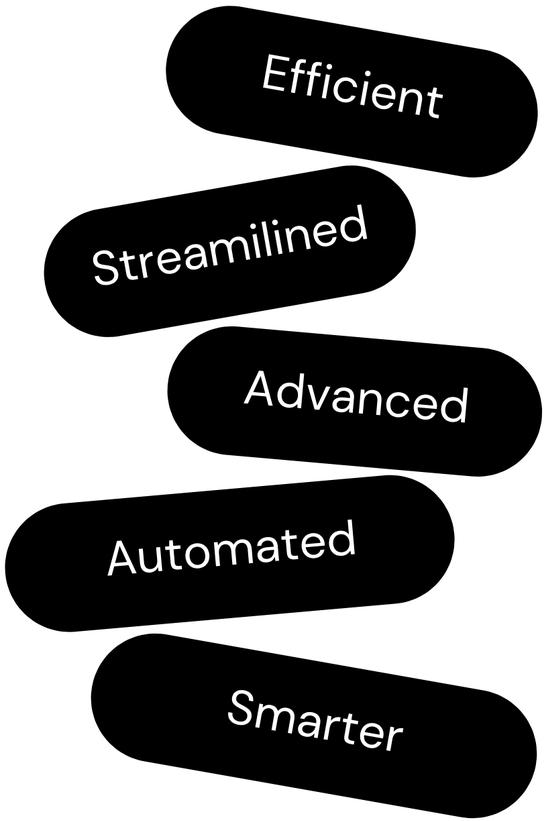
Nehil Patil-521210
Rakesh Kumar-521222
Saurav Kumar-521235
Suryanshu Kumar-521249

Table of contents

01

Why This Project?

Why this project??



Smart Parking systems can help optimize parking space usage, improve the efficiency of parking operations, and enable smoother traffic ow. By utilizing advanced technologies such as sensors, cameras, and real-time data analytics, Smart Parking systems can provide drivers with accurate information on parking availability, eliminating the need for time-consuming searches.

Additionally, by automating the entire process, Smart Parking systems eliminate the need for manual intervention, reducing the time and resources required to manage parking facilities.

Overall, Smart Parking systems provide a faster, more streamlined parking experience that benefits both drivers and parking operators.

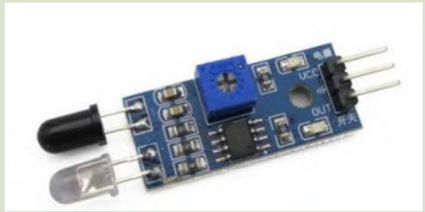
02

Components Used

Components Used

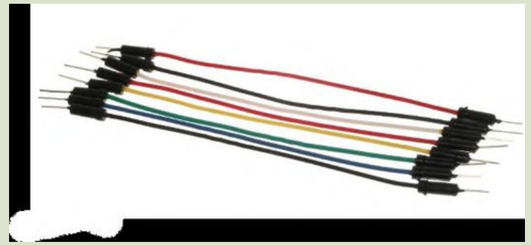
STM32F407VG

IR sensors

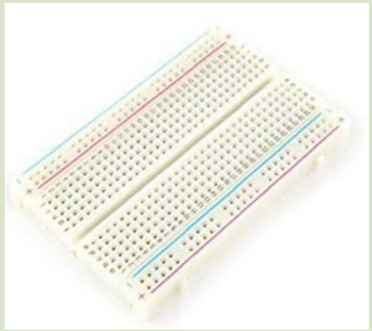


Servo motor

Jumper wires

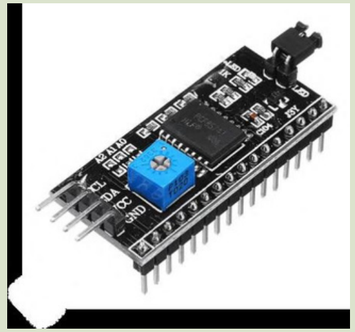


Breadboard



16×2 LCD

I2C module

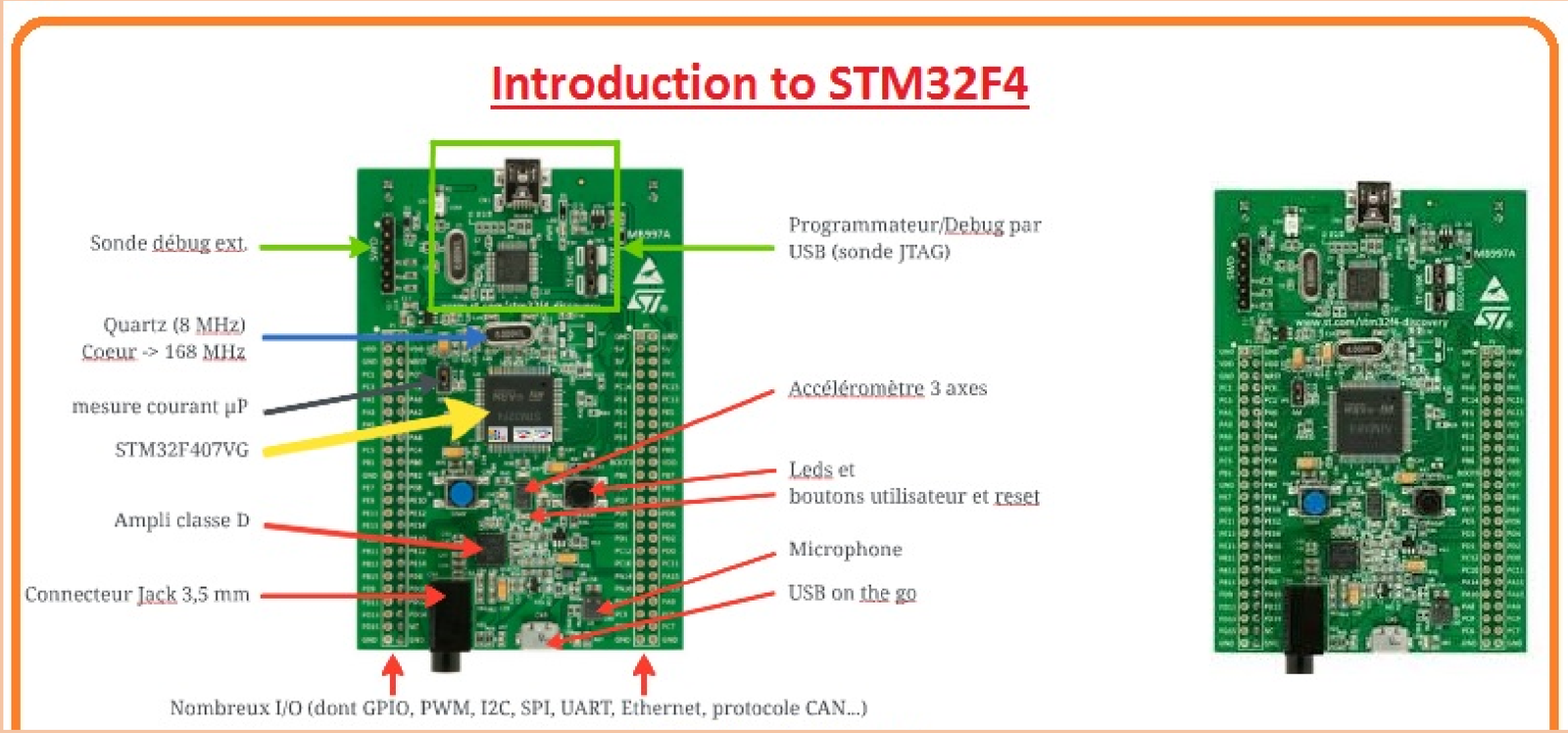


USB cable

Introduction TO STM32F4

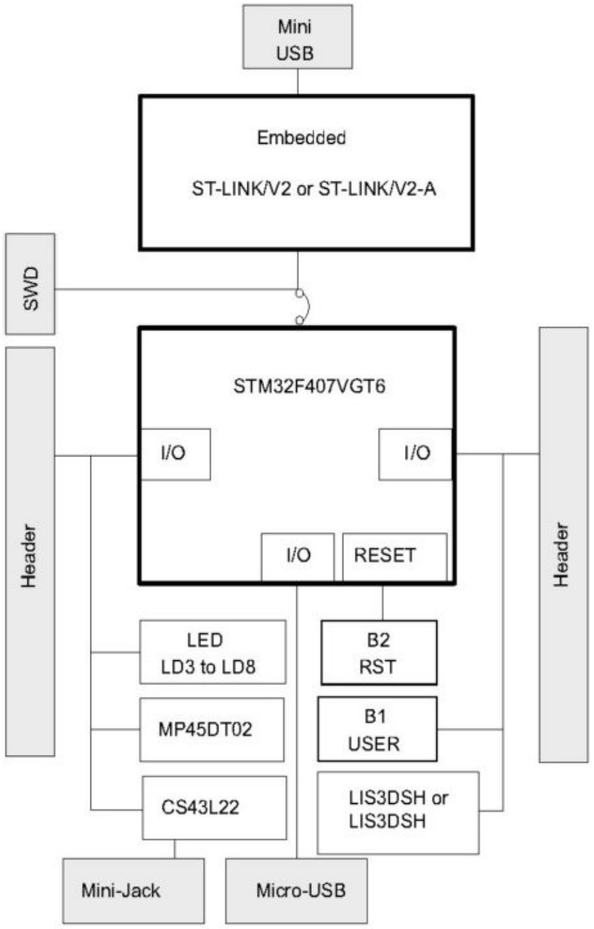
- The STM32F4 belongs to the family of thirty-two-bit ARM Microcontrollers which is created by the STmicrocontroller.
- The STM32F4 belongs to the family of thirty-two-bit ARM Microcontrollers which is created by the microcontroller.

- There are numerous applications provided by this module different categories of communications devices can be attached to this controller for linking of different types of electronic instruments such as detectors, and motors. etc.
- This board comprises numerous other devices used for communication and interfacing of different projects without any use of the external device
- This module also comes with some other components such as digital to analog converter, analog-to-digital converter, the port of sound, all these components make is the complete board.

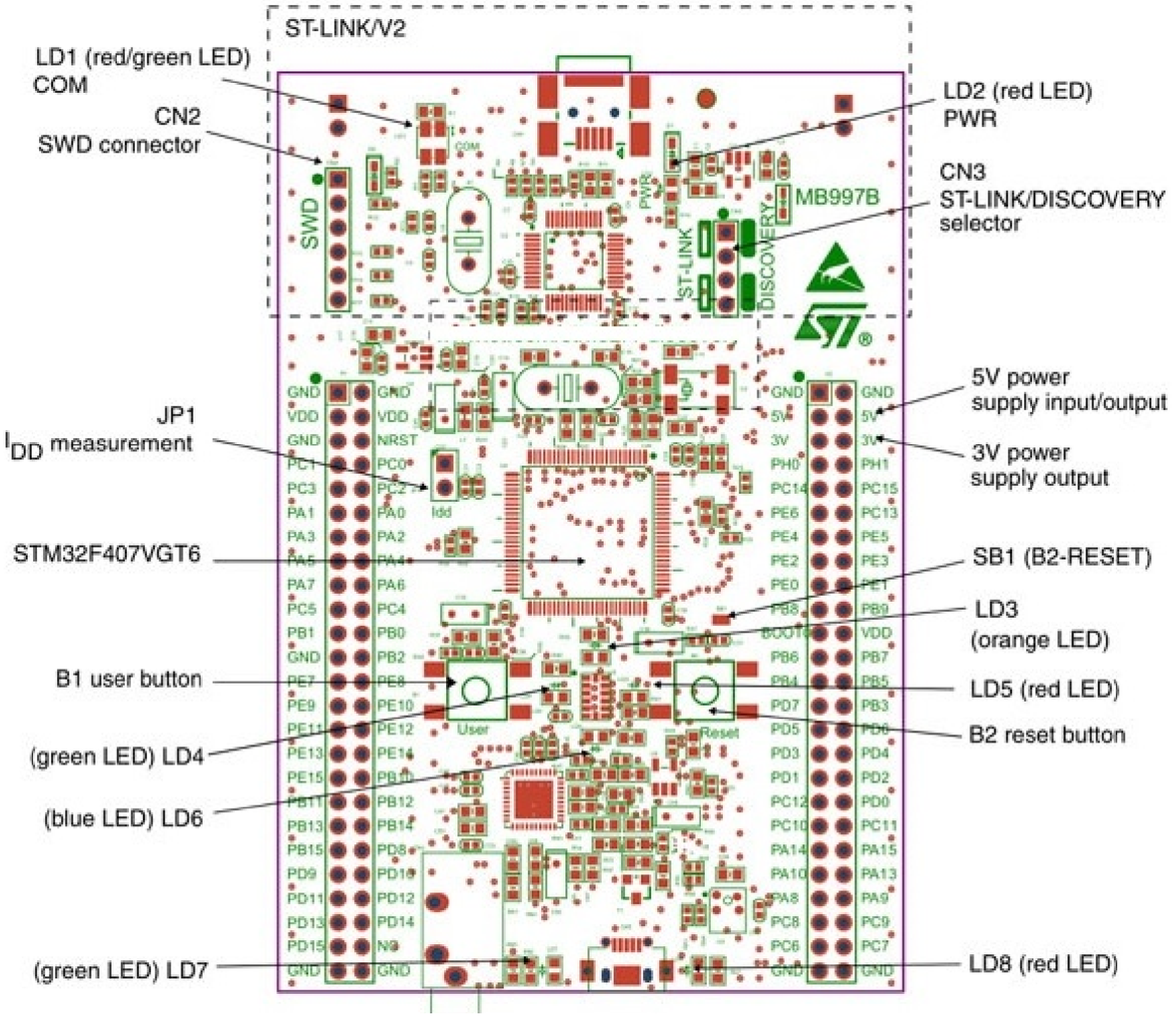


Block Diagram

STM32F4 Block Diagram



STM32F4 Onboard Devices

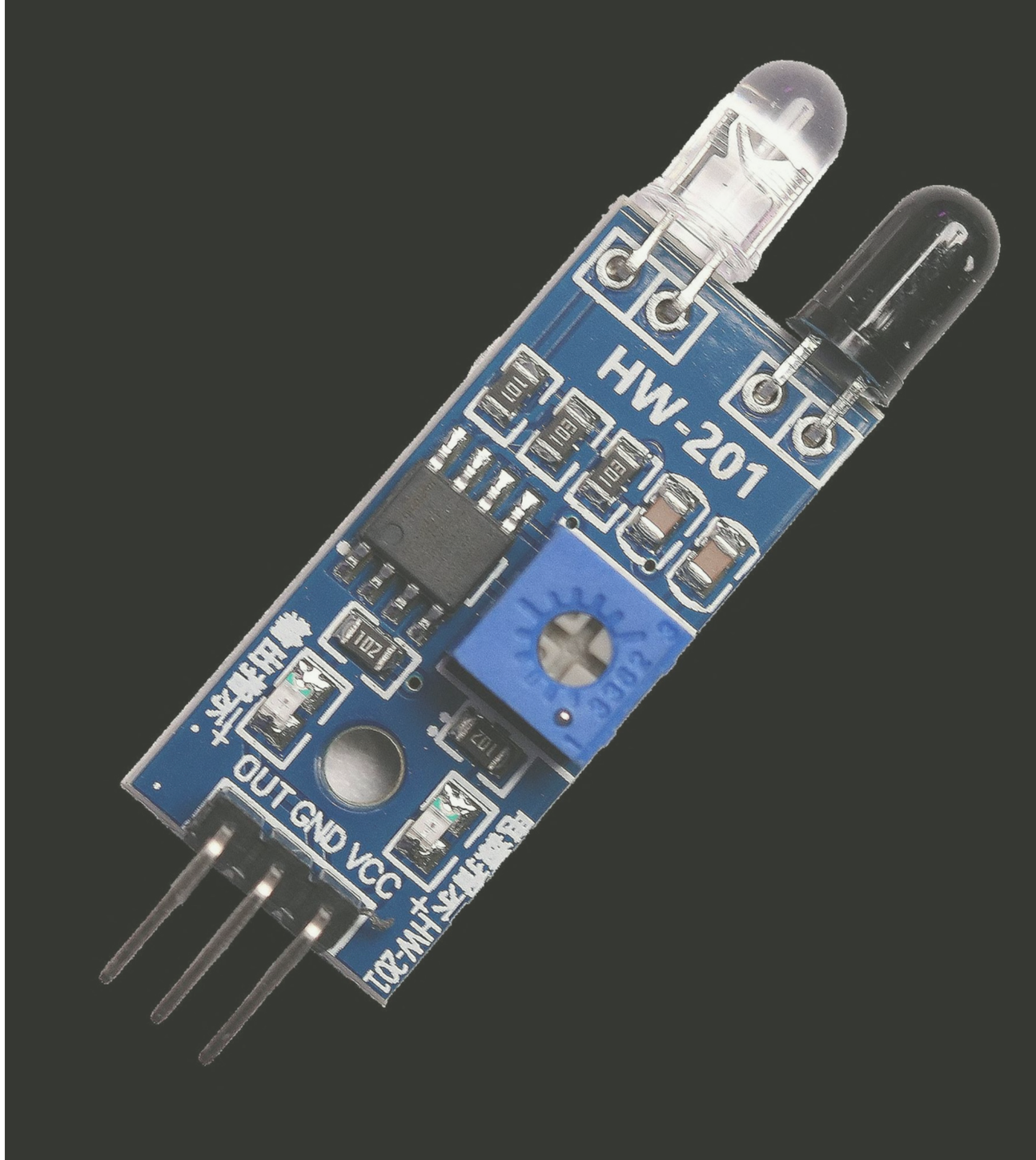


04

IR Proximity Sensors

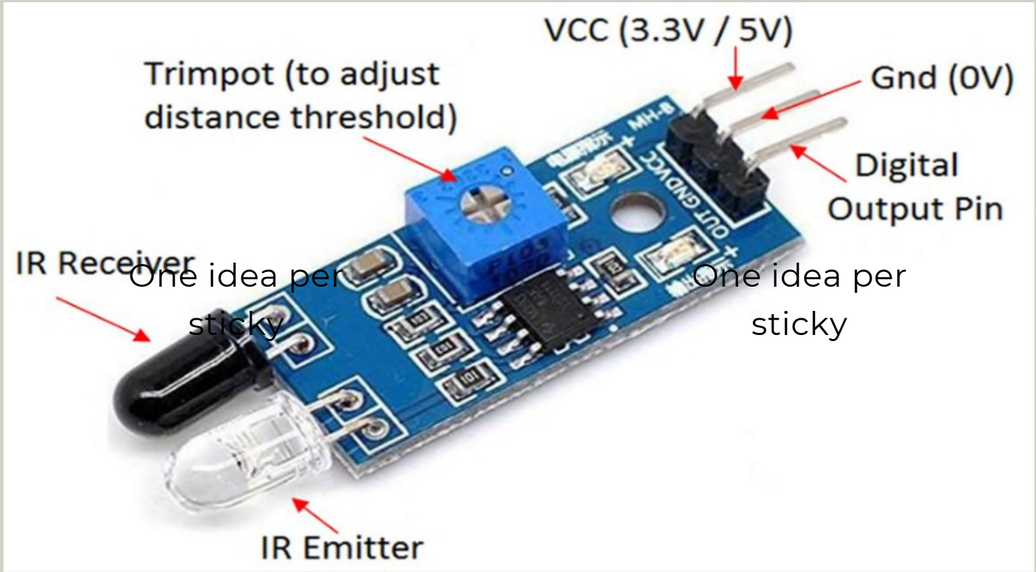
IR Sensors

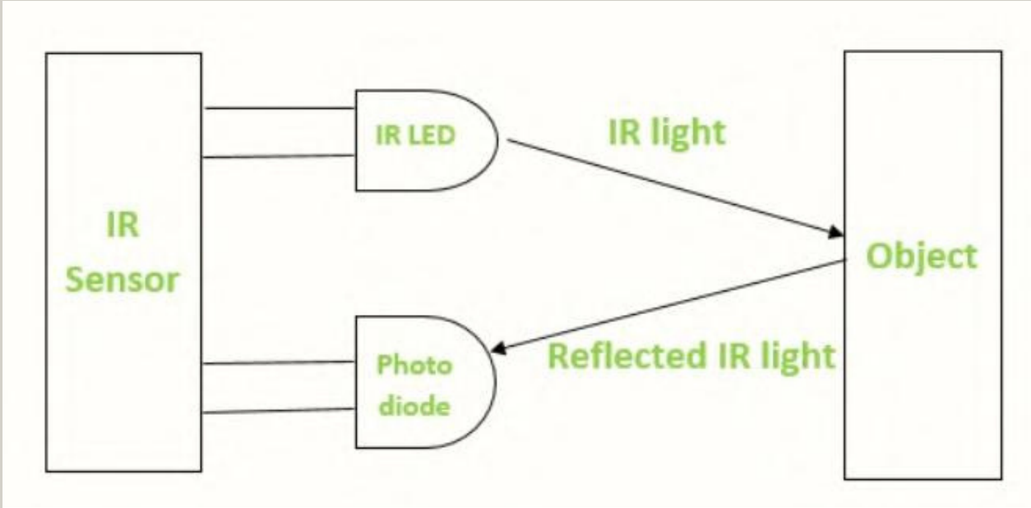
Infrared (IR) Sensor Module is a distance proximity sensor “switch”. When there is an object or obstacles that are close enough to block the view in front of 2 LEDs, it triggers the infrared trans-receiver module. The clear LED is the IR emitter while the black LED is the IR receiver. It uses the electromagnetic reflection principle where when the reflective surface (object) is closer, the receiver will receive stronger signal from transmitter due to shorter distance traveled of reflected of wave.



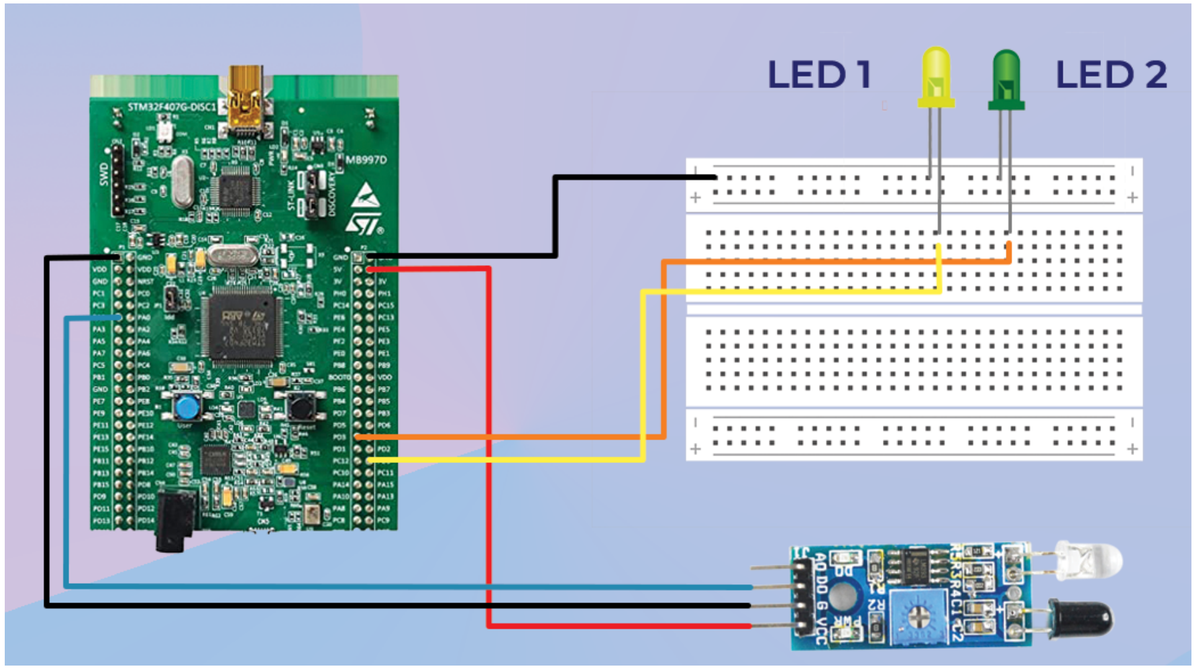
Working of an IR Sensor

Working of IR proximity sensor is same as working of IR sensors.
IR proximity sensor works by applying voltage to a pair of IR light emitting diode(LEDS) which in turn emits infrared light Radiation transmitted by the transmitter is received by the receiver after being reflected from an object
The reflected light is stronger when the object is close
Distance calculated based on the intensity radiation.





Connections with STM



Pinout:

LED1– PD0 (Pin 0 of Port D)

LED2– PD1 (Pin 1 of Port D)

IR (5V)– 5V of STM Board

IR (Gnd)– GND of STM

IR (Dout)– PA1 (Pin 1 of Port A)

Advantages of IR proximity sensors

1.Their lower power consumption
2.Simple design
3. Useful features
4.Infrared sensor can detect movement as well as heat of an object.
5.Extremely cheap.
6.No interference with electrical devices .

Applications of IR Sensors

An IR sensor module widely used in various electronics application such as Remote control, motion detection, proximity sensing and more

It is used in gas warning device, gas analyzers, medical gas measurement technology, flame detectors and in contactless precision temperature.

It is used in IR imaging devices, optical power meters, sorting device, missile guidance, remote sensing flame monitors, moisture analyzers .

IR sensors are also used in Radiation thermometers, night vision devices, photobio modulation, climatology

05

I2C LCD

Introduction to LCD

An LCD (Liquid Crystal Display) screen is an electronic display module and has a wide range of applications.

A 16x2 LCD display is very basic module and is very commonly used in va rious devices and circuits.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines.
In this LCD each character is displayed in 5x7 pixel matrix

The 16 x 2 intelligent alphanumeric dot matrix display is capable of displ aying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.

Introduction to LCD continued

I2C_LCD is an easy-to-use display module

An I2C LCD advantage is that wiring is straightforward, requiring only two data pins to control the LCD.

The character LCD is ideal for displaying text and numbers and special characters.

Atypical I2C LCD display consists of an HD44780-based character LCD display and an I2C LCD adapter. Let’s learn more about them.

LCDs incorporate a small add-on circuit (backpack) mounted on the back of the LCD module.

The module features a controller chip handling I2C communications and an adjustable potentiometer for changing the intensity of the LED backlight.

06

Servo Motors

Introduction

A **servo motor** can rotate with great precision. This type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision.

It is just made up of a simple motor which runs through a **servo mechanism**. If motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor.

It consists of three parts:

1.Controlled device

2.Output sensor

3.Feedback system

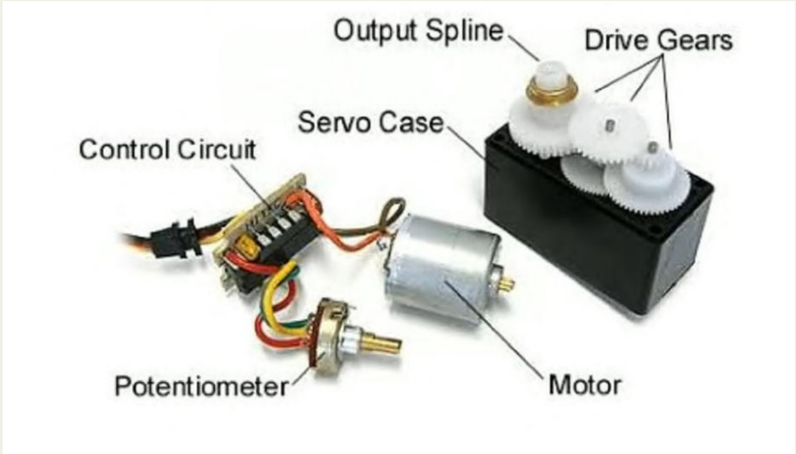
You can control the servo motor by sending a series of pulses to it. A typical servo motor expects a pulse every 20 milliseconds (i.e., the signal should be 50Hz). The length of the pulse determines the position of the servo motor.

More about Servo Motors

•We will be using an SG90 Micro Servo Motor in our experiments. It operates on 4.8-6VDC (5V typical) and can rotate 180 degrees (90 in each direction).

•It draws about 10mA when idle and 100mA to 250mA when moving, so we can power it with the Arduino's 5-volt output.

•Connect the red wire to the Arduino's 5V and the black/brown wire to ground. Finally, attach the Orange/Yellow wire to PWM enabled pin #9.



Code Snippet

STM32_workspace_9.3 - C/C++ - carparking/src/main.c - Atollic TrueSTUDIO for STM32

File Edit Source Refactor View Navigate Search Project Run Window Help

Quick Access

main.c main.c main.c *main.c

```c
#include "stm32f4xx_hal.h"
#include "lcd_library.h"  // Assume you have an LCD library for display

#define SENSOR_COUNT 4
#define SENSOR_PINS {GPIO_PIN_0, GPIO_PIN_1, GPIO_PIN_2, GPIO_PIN_3}
#define SENSOR_PORT GPIOA

#define MOTOR_PIN GPIO_PIN_4
#define MOTOR_PORT GPIOB

// Display initialization function (replace with actual initialization code)
void LCD_Init(void) {
    // Your LCD initialization code here
}

// Function to update the display with parking status
void UpdateDisplay(int* parkingStatus) {
    // Your display update code here, use LCD library functions
    // to show the status of each parking space
}

void GPIO_Init(void) {
    __HAL_RCC_GPIOA_CLK_ENABLE();
    __HAL_RCC_GPIOB_CLK_ENABLE();

    GPIO_InitTypeDef GPIO_InitStruct = {0};

    // Configure sensor pins
    GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
    GPIO_InitStruct.Pull = GPIO_NOPULL;

    for (int i = 0; i < SENSOR_COUNT; ++i) {
        GPIO_InitStruct.Pin = SENSOR_PINS[i];
        HAL_GPIO_Init(SENSOR_PORT, &GPIO_InitStruct);
    }

    // Configure motor pin
    GPIO_InitStruct.Pin = MOTOR_PIN;
    GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
    GPIO_InitStruct.Pull = GPIO_NOPULL;
    GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(MOTOR_PORT, &GPIO_InitStruct);
```

Code Snippet

STM32_workspace_9.3 - C/C++ - carparking/src/main.c - Atollic TrueSTUDIO for STM32

File Edit Source Refactor View Navigate Search Project Run Window Help

Quick Access

main.c | main.c | main.c | *main.c

```
    GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(MOTOR_PORT, &GPIO_InitStruct);
}

int main(void) {
    // Initialize the HAL Library
    HAL_Init();

    // Configure the system clock
    SystemClock_Config();

    // Initialize GPIO pins
    GPIO_Init();

    // Initialize LCD display
    LCD_Init();

    int parkingStatus[SENSOR_COUNT] = {0};  // 0: Empty, 1: Full

    while (1) {
        // Check status of each parking space
        for (int i = 0; i < SENSOR_COUNT; ++i) {
            if (HAL_GPIO_ReadPin(SENSOR_PORT, SENSOR_PINS[i]) == GPIO_PIN_SET) {
                // Parking space i is occupied
                parkingStatus[i] = 1;
            } else {
                // Parking space i is vacant
                parkingStatus[i] = 0;
            }
        }

        // Check if any parking space is occupied
        if (parkingStatus[0] || parkingStatus[1] || parkingStatus[2] || parkingStatus[3]) {
            HAL_GPIO_WritePin(MOTOR_PORT, MOTOR_PIN, GPIO_PIN_SET);  // Close the barrier
        } else {
            HAL_GPIO_WritePin(MOTOR_PORT, MOTOR_PIN, GPIO_PIN_RESET);  // Open the barrier
        }

        // Update the display with parking status
        UpdateDisplay(parkingStatus);
    }
}
```

The system automatically detects whether the parking slot is empty or not. If there is an empty parking slot, the automated gate will allow the new vehicle to enter. If in case the parking is full the entrance will be blocked by the servo barrier and the LCD screen will display "NO AVAILABLE SLOTS". The system also allows the visitors to see the number of available free spaces outside itself on the LCD screens present. The data gets updated as the vehicles move in and out of the parking.

Conclusion

The implementation of the Smart Car Parking System presents an opportunity to revolutionize parking solutions at college campuses, offering users a more efficient, sustainable, and user-friendly parking experience. This innovative project aligns with the college's commitment to technological advancement and environmental responsibility.