ESP32-CAM PIR MOTION DETECTOR WITH PHOTO CAPTURE

A

Mini Project Report

Submitted in the Partial Fulfillment of the

Academic Requirements

for the Award of the Degree of

Bachelor of Technology

in

Electronics and Communication Engineering

By

N.AISHWARY (19AG1A0489)

N.ROHIT REDDY (19AG1A0490)

K.MANIKANTA (19AG1A0480)

Under the esteemed guidance

MR.Y.V.S.DURGA PRASAD

Associate professor

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ACE Engineering college

(NBA ACCREDITED B. TECH COURSES,ECE,EEE,CSE,MECH,CIVIL)

(NAAC "A" GRADE)

An Autonomous Instution

Ankushapur(V), Ghatkesar(M),R.R.Dist-501301

2022-2023

ACE

Engineering College

(NBA ACCREDITED B. TECH COURSES,ECE,EEE,CSE,MECH,CIVIL)

(NAAC "A" GRADE)

An Autonomous Instution

Ankushapur (V), Ghatkesar (M), Medchal.Dist – 501 301

CERTIFICATE

This is to certify that the project work entitled **"ESP32-CAM PIR MOTION DETECTOR WITH PHOTO CAPTURE"** done by

N. AISHWARYA	**19AG1A0489**
N.ROHIT REDDY	**19AG1A0490**
K.MANIKANTA	**19AG1A0480**

Department of Electronics and Communication Engineering, is a record of Bonafide work carried out by them. This mini project is done as partial fulfilment of obtaining **Bachelor of Technology degree** to be awarded by **Jawaharlal Nehru Technological University, Hyderabad** during the academic year 2022-2023

MR.Y.V.S DURGA PRASAD
(Associate Professor)
(Project Supervisor)

DR. P. SATISH KUMAR
(Professor & HOD-ECE)

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We take this opportunity to thank one and all who have helped in making this project possible. We are thankful to **JNTUH** for giving us this opportunity to work on a project as a part of our curriculum.

We the students of the **ECE department** of **ACE Engineering College** would like to convey our heart full thanks to **Prof. Y. V. GOPALA KRISHNA MURTHY**, for his moral support and provided facilities and **Dr. B. L. RAJU, Principal** of the college for the wonderful guidance and encouragement given to us to move ahead in the execution of this project.

We are highly grateful to the great personality in the field of electronics, none other than **Dr. P. SATISH KUMAR, Head of the Department of Electronics and Communication Engineering** of ACE for guiding and taking care of our career in this field, we are thankful to Sir.

We wish to express our deep sense of gratitude to our Project coordinator, **Prof. B. GIRIRAJU** who has given continuous support throughout work successfully.

We are happy for being guided by **Mr.Y.V.S DURGA PRASAD, Associate Professor,** for his guidance given to us to complete our technical project work successfully.

Above all, we are very much thankful to the management of **ACE Engineering College** which was established by the high-profiled intellectuals for the cause of Technical Education in the modern era. We wish that ACE sooner should become a deemed university and produce uncountable young engineers and present them to the modern technical world.

WITH REGARDS

N.AISHWARAYA (19AG1A0489)

N.ROHIT REDDY (19AG1A0490)

K.MANIKANTA (19AG1A0480)

ABSTRACT

The ESP32-CAM is in deep sleep mode with external wake up enabled. When motion is detected, the PIR motion sensor sends a signal to wake up the ESP32.The ESP32-CAM takes a photo and saves it on the micro-SD card. Insert a formatted micro SD card and power your circuit – you can use a portable charger, after uploading code and building the circuit. Then click the reset (RST) button to get it to operate again. It activates the flash, snaps a photo, and saves it to the micro SD card when it senses motion. It goes back to deep sleep mo de until a new signal from the PIR motion sensor is received. After that, connect the micro SD card to your computer to view the photographs. We can easily design this motion sensor with a photo capture circuit using ESP 32 - CAM, PIR sensor, and some basic electronics components. And we can easily supply this circuit with a 5V DC charger.

CONTENTS

CHAPTERS

FIGURES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION:

A security framework that can have the option to recognize and screen the zone and respond viably to security danger is in incredible need. Because of the expanding number in wrongdoing and theft, the requirement for a proficient security framework is fundamental. There are now loads of security frameworks in the realm of innovation as of now, in the market for both indoor and outdoor applications, for example, Ultrasonic identifiers, CCTV, microwave indicators, photoelectric finders, infrared locators and so forth. Anyway, the greater part of these frameworks of being costly in the market, or they require increasingly electrical force development, more memory space of usage of the account framework and complex circuitry, and so on. Accordingly, an answer for conquer these issues could be by utilizing a sensor of minimal effort which can identify the interlopers, and other security astonishingly inside the sensor's discovery run and creates and yield. This yield is can likewise be utilized to additionally flagging and actuating other security gadgets like caution framework, helping framework and other comparable security danger gadgets. Which means this framework can spare force utilizations on the grounds that these segments get activated just when there are gate crashers and security dangers in the sensor's discovery run. A Passive Infrared radiation motion sensor is security-based system that saves the power consumption and the memory space of the recording system. Passive Infrared Radiation (PIR) sensor detects the change in infrared radiation of warm-blooded moving object in its detection range. The use of motion detectors goes back to ancient societies that developed agriculture and motion detection of people and things can be traced back to the early decades of the 20th century, with many of the same principles still in use today. The objective of this work is to develop a simplified version of a PIR sensor which can be installed on campus and houses for lightening systems, shops and malls for security systems and other major applications and places all over the globe.

1.2 BLOCK DIAGRAM

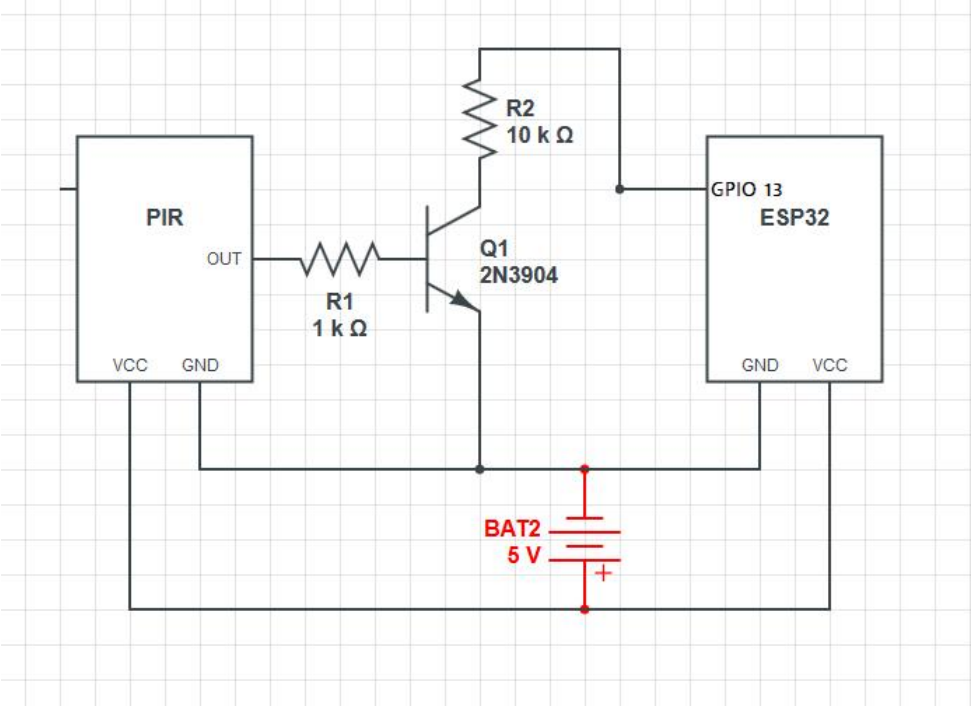


Fig1.2: BLOCK DIAGRAM

1.2.1 ESP32 CAM:

ESP32-CAM is a low-cost ESP32-based development board with onboard camera, small in size. It is an ideal solution for IoT application, prototypes constructions and DIY projects. The board integrates Wi-Fi, traditional Bluetooth and low power BLE, with 2 high performance 32-bit LX6 CPUs. It adopts 7-stage pipeline architecture, on-chip sensor, Hall sensor, temperature sensor and so on, and its main frequency adjustment ranges from 80MHz to 240MHz. Fully compliant with Wi-Fi 802.11 and Bluetooth 4.2 standards, it can be used as a master mode to build an independent network controller, or as a slave to other host MCUs to add networking capabilities to existing devices. It is suitable for home smart devices, industrial wireless control, wireless monitoring, QR wireless identification, wireless positioning system signals and other IoT applications.

Fig 1.2.1: ESP32-CAM

1.2.2 PIR SENSOR

All living objects, whose body temperature is more than 0 degree C, emit the heat in form of infrared radiation through their body, also called as thermal radiations. This Radiated energy is invisible to human eye. These Signals can be detected by using PIR sensor which is specially designed for such purpose. In Passive Infrared (PIR) Sensor, passive word indicates PIR Sensor does not generate or radiate any energy for detection purposes. PIR Sensors don't detect or measure "HEAT"; they detect the infrared radiation emitted or reflected from objects. They are small, inexpensive, low power and easy to use. They are commonly found at home, medical, factories etc. areas.

Fig 1.2.2: PIR SENSOR

1.2.3 FTDI

232 USB to Serial interface board This cable is used to transmit and receive data between computer and external devices such as micro controllers, Arduino, development

modules (Bluetooth, GPS, GSM, etc.). Most importantly, FTDI cable is used to connect RS232 standard based devices to Personal computers and laptops. GROUND: Connect to the ground pin of the device to which you want to connect with the computer. CTS: Clear to Send = This is control input and is used to clear the send request of Data. VCC: Connect it with Vcc TxD: Transmit Asynchronous Data = This is an output pin and used to transmit output data asynchronously RxD: Receive Asynchronous Data = This is an input pin and used to receive

input data asynchronously. RTS: This is a control output pin and is used to make a request for sending the data. Standard interface layout, compatible with a variety of Arduinos such as the Pro Mini Original FTDI FT232 chip, stable performance USB power has current protection, using 500MA self-restore fuse RXD/TXD transceiver communication indicator With power, sending, receiving indicator, working status LED indicators Mini USB Port Connection, Support 3.3V, 5V

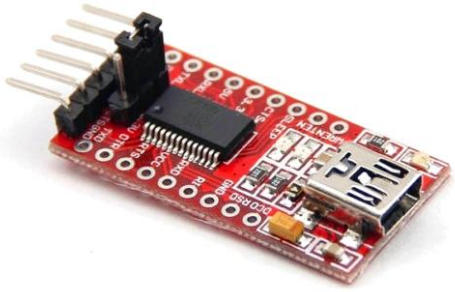


Fig 1.2.3: FTDI PROGRAMMER

1.2.4 TRANSISTOR

A transistor has three terminals, namely emitter (E), base (B) and collector (C). We have two types of transistors, npn and pnp. The emitter is heavily doped and injects a large number of majority carriers into the base. The emitter is always forward biased with respect to the base. In pnp transistors, majority carriers are holes and in npn transistors, majority carriers are electrons.

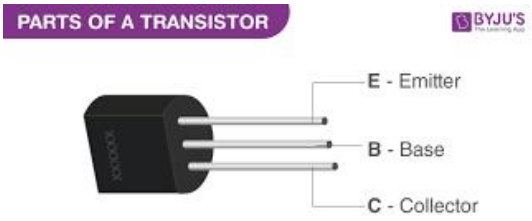


Fig 1.2.4: Transistor

1.2.5 RESISTOR

The resistor is a passive electrical component that creates resistance in the flow of electric current. In almost all electrical networks and electronic circuits they can be found. The resistance is measured in ohms (Ω). An ohm is the resistance that occurs when a current of one ampere (A) passes through a resistor with a one volt (V) drop across its terminals. The current is proportional to the voltage across the terminal ends.

Fig 1.2.5: Resistor

CHAPTER 2

HARDWARE DESCRIPTION

2.1 HARDWARE COMPONENTS

1. ESP32-CAM

2. PIR SENSOR

3. FTDI PROGRAMMER

4. MICROSD CARD

5. TRANSISTOR

6.RESISTOR

7. JUMPER WIRES

8. 3.7V BATTERY

9. BREAD BOARD

2.2 ESP32-CAM

2.2.1 ESP32-CAM

The ESP 32 CAM module may be a full-featured microcontroller that also has an integrated video camera and microSD card socket. It's inexpensive and simple to use, and is ideal for IoT devices requiring a camera with advanced functions like image tracking and recognition. The sample software distributed by Express if includes a sketch that permits you to create a web-based camera with a classy electrical device. After you get the hang of programming the device, you'll find that it's very easy to use. The ESP32-CAM module has

fewer I/O pins than the previous ESP-32 module we checked out. Many of the GPIO pins are used internally for the camera and also the microSD card port. Another thing missing from the ESP32-CAM module could be a USB port. so as to program this device, you'll must make use of an FTDI adapter

Fig2.2.1 :ESP32-CAM

2.2.2 GENERAL PIN FUNCTION:

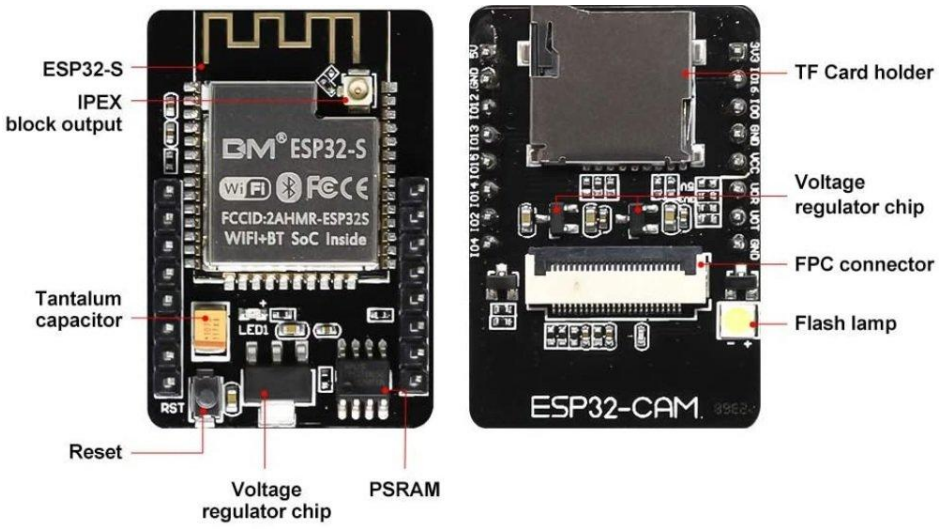


FIG 2.2.2:ESP32-CAM AI-THINKER

ESP32-S CHIP: The module is a main chip contains two high-performance 32-bit LX6 CPU's with a 7-stage pipeline architecture and used for all the processing and functioning.

IPEX block output: The printed IPEX connects GSM antennas to transmit signals.

Tantalum capacitor: The tantalum capacitor is majorly used on small size modules. They are durable and provide power supply filtering for fine signal quantity.

Reset button: When pressed, the reset button restarts the code executed on the module.

Voltage regulator chip: The voltage regulator chip on the module maintains the output voltage despite the fluctuations in the input supply. It regulates the voltage to 3.3 volts.

PSRAM: A low-power Pseudo-Random Access Memory of 4MB is incorporated in the module for fast processing of the instructions. It helps the camera to run smoothly.

TF Card Holder: ESP32 series are embedded with a micro-SD card holder to store the data. All the transmission takes place through the Serial Peripheral Interface.

FPC connector: To mount the camera, the ESP32 module contains a flexible printed circuit connector. Their fine pitch is responsible for signal reliability.

Flash Light: The flash lamp produces electric pulses which work as a flash for the camera so that it can capture clear images.

2.2.3 ESP32-CAM PINOUT:

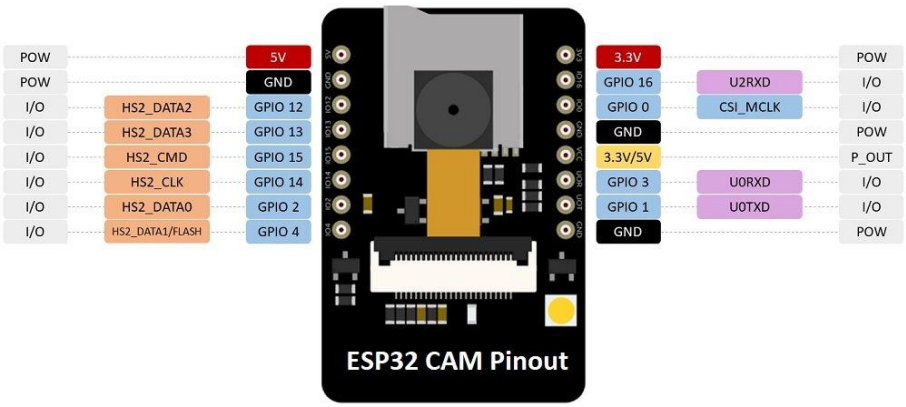


Fig 2.2.3: ESP32-CAM PINOUT

This section describes the pinout of the ESP32-CAM AI-Thinker Module. The ESP32-S chip has a total of 34 pins but only 16 pins are exposed to the pinout headers.

2.2.4 POWER PINS

The module has three ground pins and two positive power supply pins such as 5V and 3.3V pins. These pins can be used to power the ESP32-CAM AI-Thinker module.But it is recommended not to power tis development bord with 3.3V pin, it does not provide stable power to board

ESP32-CAM also provide one power output pin as shown in yellow color in the pinout diagram above. This is a VCC pin that can output either 5V or 3.3V. According to Jumper connection on ESP32-CAM, VCC pin provides 3.3V output.

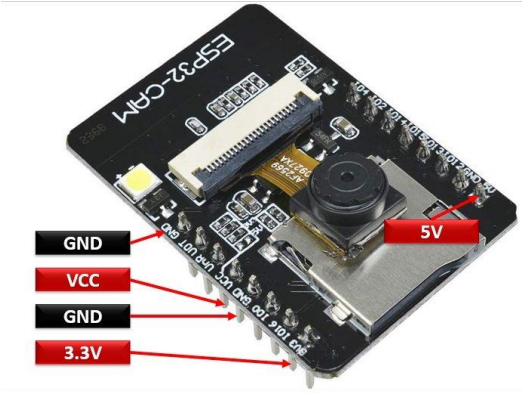


Fig 2.2.4: ESP32-CAM POWER PINS

2.2.5 GPIO33 – Built-in Red LED

AI-Thinker board also has one onboard RED LED. You can find this LED next to the reset button. This built-in red LED is connected with GPIO 33 in logic low active state. That means if we want to turn on LED, we drive GPIO 33 to logic low level. Similarly, to turn off LED, we drive GPIO 33 to active high state.

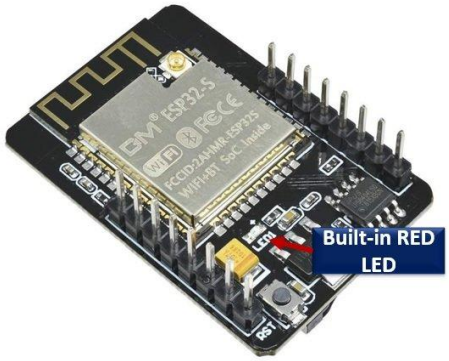


Fig 2.2.5: Built-in RED LED

2.2.6 UART PINS

Almost all the GPIO pins of ESP32-CAM are multi-purpose pins. GPIO1 and GPIO3 has alternate functions for the serial transmission and reception of the data for UART port, respectively. AI-Thinker board does not come with onboard programmer. Therefore, these UART pins are used to program and communicate with the PC to upload the code.

PIN NAME	FUNCTION
GPIO1	U0TXD (UART Transmission pin)
GPIO3	UORXD (UART Reception pin)

2.2.7 GPIO0 PIN- FLASH MODE SELECTION

This pin determines the mode of the module such as flash or normal mode. In flashing mode, GPIO0 is pulled low which means it should be connected to the ground. When GPIO0 is connected to ground, ESP32-CAM goes into flashing mode and we can flash the code. After flashing the code to board, we should disconnect the GPIO0 from the ground to run the module normally or we say it enters the normal mode

- GPIO0 connected to ground à ESP32-CAM in Flash Mode
- GPIO0 not connected to ground a ESP32-CAM in normal program execution mode

2.2.8 SD CARD PIN

ES32-CAM board has built-in SD card connector which can used to connect an SD card. These GPIO pins use for connections with micro SD cards while reading and writing data to SD card. These GPIO pins can be used as typical I/O pins if the SD card is not in use.

2.2.9 FLASH LED PIN

ESP32-CAM also has built-in high brightness flashlight. This flashlight can be used with a Camera while taking picture in darkness. GPIO4 is connected to an embedded flash lamp which (if programmed so) flashes when the camera captures the pictures. GPIO4 is also connected to the SD card, so it might cause difficulty accessing both at the same time. That means flashlight may glow unwanted while using SD card.

Fig 2.2.9: Flash Light

2.2.10 Camera Connector pins

OV2640 CAMERA	ESP32	Variable name in code
D0	GPIO5	Y2_GPIO_NUM
D1	GPIO18	Y3_GPIO_NUM
D2	GPIO19	Y4_GPIO_NUM
D3	GPIO21	Y5_GPIO_NUM
D4	GPIO36	Y6_GPIO_NUM
D5	GPIO39	Y7_GPIO_NUM

D6	GPIO34	Y8_GPIO_NUM
D7	GPIO35	Y9_GPIO_NUM
XCLK	GPIO0	XCLK_GPIO_NUM
PCLK	GPIO22	PCLK_GPIO_NUM
VSYNC	GPIO25	VSYNC_GPIO_NUM
HREF	GPIO23	HREF_GPIO_NUM
SDA	GPIO 26	SIOD_GPIO_NUM
SCL	GPIO 27	SIOC_GPIO_NUM

2.2.11 ESP32-CAM Features

- supports Smart Config/AirKiss Technology and has capacitive GPIO pins.
- 18 high-resolution analog-to-digital converters.
- 2 digital-to-analog converters with a resolution power of 8-bits.
- Consumes power up to 600 DMIPS at 160 MHz of clock frequency.
- The module has high-speed WiFi with a speed of 2.4 GHz.
- ESP32 module contains 2 I2C, 3 SPI, and 3 UART for fast serial transmission.
- It has a TF holder and supports WiFi image upload.
- ESP32 supports 4 different operating modes i.e s STA/AP/STA+AP
- Embedded Lwip and FreeRTOS.
- RTC supported to wake up the module from Deep-sleep mode.
- A built-in flashlight is provided in the module to produce flashes to capture bright clear images.
- A CMOS technology IC has an adjustable output power amplifier
- ESP32 has a low duty cycle to reduce power consumption making it power efficient

2.3 PIR SENSOR

2.3.1 PIR SENSOR

All living objects, whose body temperature is more than 0 degree C, emit the heat in form of infrared radiation through their body, also called as thermal radiations. This Radiated energy is invisible to human eye. These Signals can be detected by using PIR sensor which is specially designed for such purpose. In Passive Infrared (PIR) Sensor, passive word indicates PIR Sensor does not generate or radiate any energy for detection purposes. PIR Sensors don't detect or measure "HEAT"; they detect the infrared radiation emitted or reflected from objects. They are small, inexpensive, low power and easy to use. They are commonly found at home, medical, factories etc. areas.

Fig 2.3.1: PIR SENSOR

2.3.2 PIR SENSOR PINOUT

The PIR sensor stands for Passive Infrared sensor. It is a low cost sensor which can detect the presence of Human beings or animals. This **HC-SR501 PIR sensor module** has three output pins Vcc, Output and Ground as shown in the pin diagram above. Since the output pin is 3.3V TTL logic it can be used with any platforms like Arduino, Raspberry, PIC, ARM, 8051 etc..

Pin Name	Description
Vcc	Input voltage is +5V for typical applications. Can range from 4.5V- 12V
High/Low Ouput (Dout)	Digital pulse high (3.3V) when triggered (motion detected) digital low(0V) when idle(no motion detected
Ground	Connected to ground of circuit

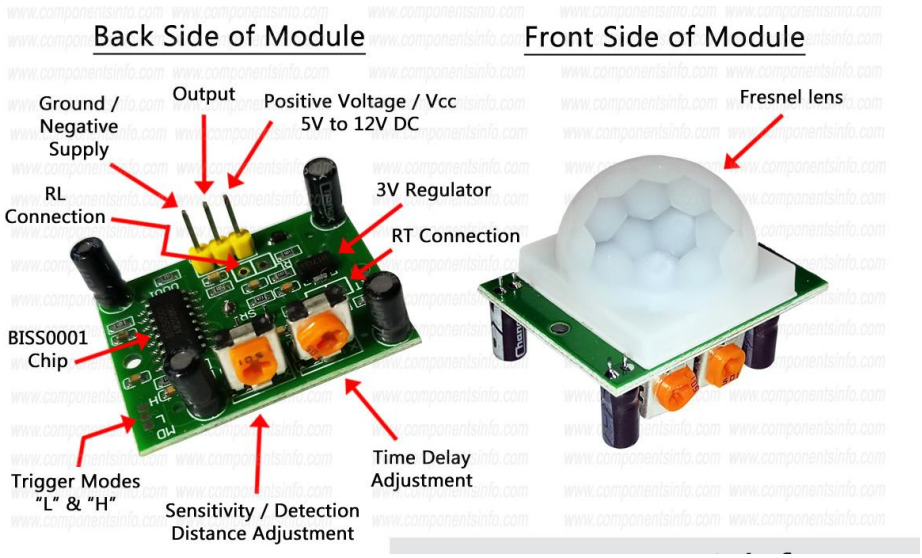


Fig 2.3.2: PIR SENSOR PINOUT

1.Module sensitivity & time delay adjustments

The back side circuitry also contains two variable resistors / adjusters, one variable resistor is given to adjust the sensitivity of the module or in other words we can say the detection range of the module and other one is used to adjust how much time the output will remain high after the PIR module has detected an activity.

2.Trigger Modes (Optional Modes)

There are two trigger modes in the module marked with the letter “L” and “H” that is also shown in the image above. These are optional modes and there is no need to work on these modes to use the module. By default the module is set on its most common trigger mode. But in case you want to use them then the points are given. To select any mode just connect it with the middle point, you can also use a jumper by soldering three pins on these points. But first it is important to understand what each mode does.

3.Trigger Mode “L”

Trigger mode “L” is called single trigger mode. By selecting trigger mode “L” the output will goes high only one time until the time delay period is completed. No matter how many time an object is moved in front of the sensor’s detection range.

ESP32-CAM PIR MOTION DETECTOR WITH PHOTO CAPTURE

4.Trigger Mode “H”

Trigger mode “H” is called repeat trigger mode. By selecting trigger mode “H” the output will goes high as many times as the object moves in the detection range of the sensor hence, the time delay started again each time a single motion is detected.

ESP32-CAM PIR MOTION DETECTOR WITH PHOTO CAPTURE

5.Temperature & light sensing (Optional Modes)

To further enhance the module performance you can also use the temperature sensing and light sensing functions / features of the module. For doing so you don’t have to do any complex thing only you have to add a sensor for each that’s it and after that all the measurement work is done by the BSS0001 chip. By closely looking at the PCB back side you will find 2 holes marked RT and 2 holes marked as RL.

6.RT

The 2 holes / points marked RT is given to solder a thermistor. A thermistor is a temperature sensing resistor whose resistance changes when the temperature around it changes. Using a thermistor increases the module performance when the temperature goes above +32 Celsius. The thermistor should be mounted from front side of the PCB.

7.RL

Using the RL feature makes the module to only work in dark which saves battery life a lot by only activating the module in dark / night. For using this feature you only have to solder a LDR or light dependent resistor. An LDR is a light sensing resistor whose resistance changes when the amount of light changes on its surface. The LDR should be mounted from front side of the module.

2.3.3 Working of the Module

The PIR motion sensor module or PIR motion sensing technology works by sensing the change occurs in the infrared radiation or heat generated by a human or animal body in movement or motion. When the pyroelectric sensor detects this phenomenon, it sends an output signal that is further processed by the BISS0001 chip. BISS001 is a low power, high quality and reliable PIR signal processing chip which is built with CMOS technology. After the signal is processed by the chip and if the signal is true then the output of the chip goes high for a certain amount of time that can be adjusted with the variable resistor used in this module.

2.3.4 APPLICATIONS

- Home Security
- Office / Work Place Security
- Saving Power (Switch ON appliances only when someone is present)
- Human Detection
- Animal Detection
- Industrial Equipment Automation

2.4 FTDI PROGRAMMER

2.4.1 FTDI PPROGRAMMER

The USB to TTL serial adapter is based on the high quality and very popular FTDI FT232RL chipset and is an excellent way to connect TTL serial devices to a PC through a USB port .

This USB to TTL serial adapter is ideal for many uses, including:

- Programming microprocessors such as ARM, AVR, etc ..
- Working with computing hardware such as routers and switches.
- Serial communication with many devices such as GPS devices.
- Serial terminals on devices like the Raspberry Pi.

Unlike most USB to TTL serial adapters, this adapter supports both 5V AND 3.3V operation! Simply set the jumper as required to choose between 5V and 3.3V as labelled on the board. The adapter comes with a right-angle connector fitted allowing you to use it straight away. If you need to access any of the other inputs or outputs of the FT232RL, all the useful signals are provided as through-hole solder pads - ideal for use with straight headers into a breadboard.

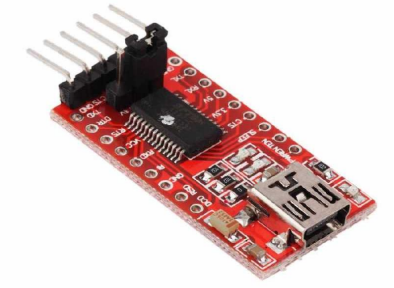


Fig 2.4.1:FTDI PROGRAMMER

2.4.2 FTDI PROGRAMMER PINS

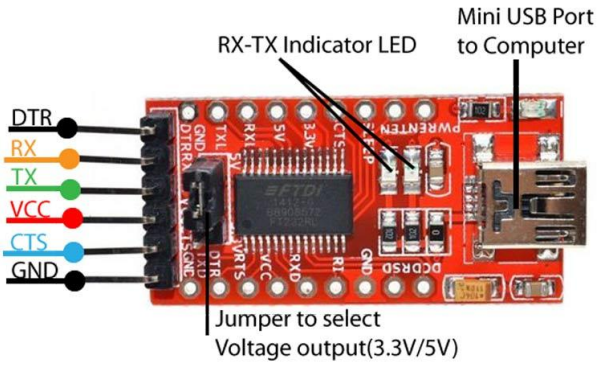


Fig 2.4.2: FTDI PROGRAMMER PINS

- **DTR:** Data Terminal Ready - an output used for flow control.
- **RX:** Serial data Receive pin.
- **TX:** Serial data Transmit pin.
- **VCC:** Positive voltage output - this is controlled by the jumper. If the jumper is set to 5V, this will provide a 5V output. If the jumper is set to 3.3V, this will provide a 3.3V output.
- **CTS:** Clear To Send - an input used for flow control
- **GND:** Ground or 0V
- **RX** on this board to the **TX** pin on your device
- **TX** on this board to the **RX** pin on your device
- **GND** on this board to **GND** on your device

The VCC pin is ideal for powering small devices such as homemade circuits. This pin should not be connected when a device has a separate power supply as this may damage both devices. Please note that in 5V mode the maximum current draw on this pin is approximately 500mA. In 3.3V mode the maximum current draw on VCC is approximately 50mA.

There are also several pins available as solder pads. These pins are labelled on the board. Connecting to these pins is not usually required and you should check the FTDI datasheet before doing so.

2.4.3 Features of USB to serial UART interface IC

- Single chip USB to asynchronous serial data transfer interface.
- Entire USB protocol handled on the chip. No USB specific firmware programming required.
- Fully integrated 1024 bit EEPROM storing device descriptors and CBUS I/O configuration.
- Fully integrated USB termination resistors.
- Fully integrated clock generation with no external crystal required plus optional clock output selection enabling a glue-less interface to external MCU or FPGA.
- Data transfer rates from 300 baud to 3 Mbaud (RS422, RS485, RS232) at TTL levels.
- 128 byte receive buffer and 256 byte transmit buffer utilising buffer smoothing technology to allow for high data throughput.

2.4.4 Features of FT232RL USB TO TTL 5V 3.3V Convertor

- Material: PCB + Electronic Component
- Support 3.3V, 5V
- Main Colour: Red
- Chipset: FT232RL
- USB power has over current protection, using 500MA self-restore fuse
- RXD/TXD transceiver communication indicator
- Pin definition: DTR,RXD,TX,VCC,CTS,GND
- Pitch:2.54mm
- Module Size: About 36mm(length)*17.5mm(width)

- Interface : Mini USB

2.5 MICROSD CARD

2.5.1 MICROSD CARD

Fig 2.5.1: MICROSD CARDS

The first thing to consider when choosing a memory card is figuring out what type of card your device requires, which can usually be found in the instruction manual or on the manufacturer's website. This will usually indicate what SD standard the device requires. Both SD and microSD cards share the same standards: SD, SDHC, SDXC and SDUC, and microSD, microSDHC, microSDXC and microSDUC.

For the average user, a 32GB or 64GB card is more than enough. A mid-sized card should be able to capture hundreds, or even thousands of photos and video clips.

2.5.2 TYPES OF SD CARD

Type	Capacity
SD	2GB and under

SDHC	More than 2GB, up to 32GB
SDXC	More than 32GB, up to 2TB
SDUC	More than 2TB, up to 128TB

The SD standards represent a timeline of the evolution of SD cards, with newer cards allowing higher capacities and speeds. SDUC was announced in June 2018 and it will be a while before the SDUC cards are available and widely adopted in the industry. Hardware devices that host the memory cards are backward compatible, meaning you can use older standard memory cards in devices that support newer standards like SDXC.

For example, an SDXC-compatible device can use SDXC, SDHC and SD standard cards. An SDHC-compatible device can use SDHC and SD standard cards but will not be able to use SDXC standard cards. SD-compatible devices are compatible only with SD standard cards. To put it more simply, hardware devices that support newer standards are backward compatible with older standard cards, but newer standard cards can't be used in hardware devices that only support older standards.

2.5.3 TYPES OF MICROSD CARDS

Type	Capacity
microSD	2GB and under
microSDHC	More than 2GB, up to 32GB
microSDXC	More than 32GB, up to 2TB
microSDUC	More than 2TB, up to 128TB

Hardware devices are also backward compatible with microSD cards, just like full-sized SD cards. The same rules apply for microSD cards as they do for SD cards. A microSDXC-compatible device will work with microSDXC, microSDHC and microSD cards. A microSDHC-compatible device will work with microSDHC and microSD cards. A microSD-compatible device will only work with a microSD card. Again, hardware devices that support newer standards are backward compatible with older standard microSD cards, but newer standard microSD cards can't be used in hardware devices that only support older standards.

2.6 TRANSISTOR

2.6.1 TRANSISTOR

A transistor has three terminals, namely emitter (E), base (B) and collector (C). We have two types of transistors, npn and pnp. The emitter is heavily doped and injects a large number of majority carriers into the base. The emitter is always forward biased with respect to the base. In pnp transistors, majority carriers are holes and in npn transistors, majority carriers are electrons.

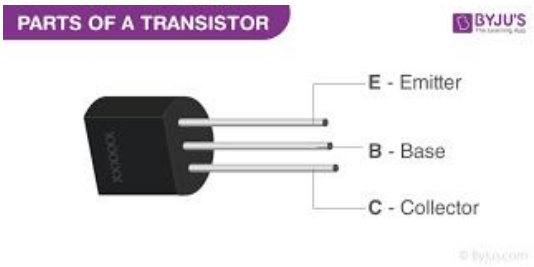


Fig 2.6.1: TRANSISTOR

2.6.2 PNP TRANSISTOR

A pnp transistor behaves exactly the same way as an npn transistor, with the difference that the majority carriers are holes. Here too, the base is lightly doped compared to the emitter and collector. The BE junction is forward biased and CB junction reverse biased.

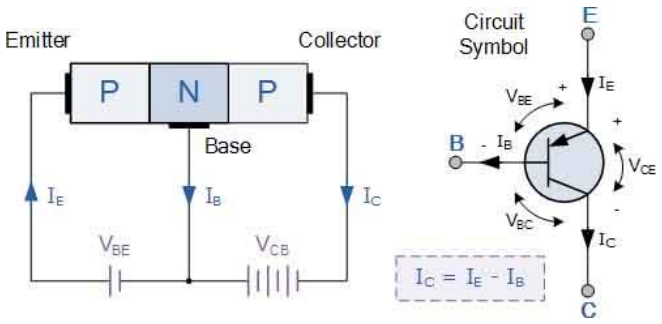


Fig 2.6.2: PNP TRANSISTOR

Holes emitted from the p-type emitter are injected into the base. Some of the holes flow out through the base. Most of them are collected by the collector. The BE junction bias voltage controls the large emitter and collector current.

2.6.3 NPN TRANSISTOR

In the previous tutorial we saw that the standard **Bipolar Transistor** or BJT, comes in two basic forms. An **NPN** (**N**egative-**P**ositive-**N**egative) configuration and a **PNP** (**P**ositive-**N**egative-**P**ositive) configuration. That is: an NPN transistor and a PNP transistor types.

The most commonly used transistor configuration is the **NPN Transistor**. We also learnt that the junctions of the bipolar transistor can be biased in one of three different ways – **Common Base**, **Common Emitter** and **Common Collector**.

In this tutorial about bipolar transistors we will look more closely at the "Common Emitter" configuration using the **Bipolar NPN Transistor** with an example of the construction of a NPN transistor along with the transistors current flow characteristics is given below.

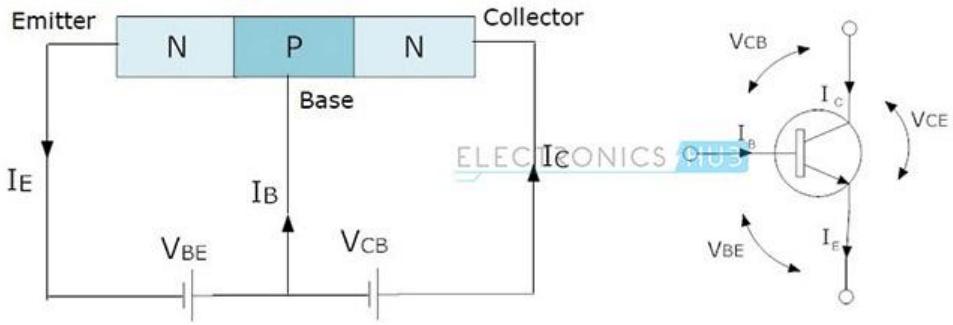


Fig 2.6.3: NPN TRANSISTOR

2.6.4 APPLICATIONS

- Well sutable for Tv and Home apppliance equipment
- Small load switch transistor with high gain low saturation voltage

2.7 RESISTOR

2.7.1 RESISTOR

The resistor is a passive electrical component that creates resistance in the flow of electric current. In almost all electrical networks and electronic circuits they can be found. The resistance is measured in ohms (Ω). An ohm is the resistance that occurs when a current of one ampere (A) passes through a resistor with a one volt (V) drop across its terminals. The current is proportional to the voltage across the terminal ends. This ratio is represented by Ohm's law

$$R=V/I$$

Fig 2.7.1 RESISTOR

Dependent on the application, the electrical engineer specifies different properties of the resistor. The primary purpose is to limit the flow of electrical current; therefore the key parameter is the resistance value. The manufacturing accuracy of this value is indicated with the resistor tolerance and is expressed as a percentage of the resistance value (for example ±5%). Many other parameters that affect the resistance value can be specified, such as long term stability or the temperature coefficient. The temperature coefficient, usually specified in high precision applications, is determined by the resistive material as well as the mechanical design.

In high frequency circuits, such as in radio electronics, the parasitic capacitance and inductance can lead to undesired effects. Foil resistors generally have a low parasitic reactance, while wirewound resistors are among the worst. For accurate applications such as audio amplifiers, the electric noise of the resistor must be as low as possible. This is often specified as microvolts noise per volt of applied voltage, for a 1 MHz bandwidth. For high power applications the power rating is important. This specifies the maximum operating power the component can handle without altering the properties or damage. The power rating is usually specified in free air at room temperature. Higher power ratings require a larger size and may even require heat sinks. Many other characteristics can play a role in the design specification. Examples are the maximum voltage or the pulse stability. In situations where high voltage surges could occur, this is an important characteristic.

Sometimes not only the electrical properties are important, but the designer also has to consider the mechanical robustness in harsh environments. Military standards sometimes offer guidance to define the mechanical strength or the failure rate.

2.7.2 RESISTOR COLOUR CODE

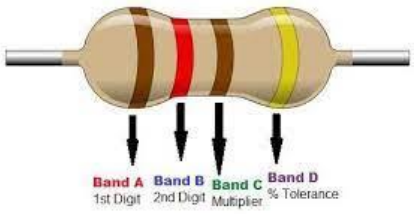


Fig 2.7.2: Resistor colour code

The resistance value and tolerance are indicated with several colored bands around the component body. This marking technique of electronic components was already developed in the 1920s. Printing technology was still not far developed, what made printed numerical codes too difficult on small components. Nowadays, the color code is still used for most axial resistors up to one watt. In the figure above, an example is shown with four color

bands. In this example the two first bands determine the significant digits of the resistance value, the third band is the multiplying factor and the fourth band gives the tolerance. Each color represents a different number and can be looked up in a resistor color code chart or using a resistor color code calculator.

2.8 JUMPER WIRES

2.8.1 JUMPER WIRES

Handy for making wire harnesses or jumpering between headers on PCB's. These premium jumper wires are 6" (150mm) long and come in a 'strip' of 40 (4 pieces of each of ten rainbow colors). They have 0.1" male header contacts on one end and 0.1" female header contacts on the other. They fit cleanly next to each other on standard-pitch 0.1" (2.54mm) header. The best part is they come in a 40- pin ribbon cable. You can always pull the ribbon wires off to make individual jumpers, or keep them together to make neatly organized wire harnesses.This male to female jumper wires are used robotics and embedded projects for interfacing devices.

Wires suitable for prototyping these wires can be connected to any header with a 2.54mm (0.1") pitch. Handy for making wire harnesses or jumpering between headers on PCB's they have 0.1" sockets on either end or fit cleanly next to each other on standard-pitch header. For best results, when plugging these in a line, have the sides with the 'silver latch bit' sticking out since that side is a tiny bit wider. A jump wire (also known as jumper, jumper wire, jumper cable, DuPont wire, or DuPont cable – named for one manufacturer of them) is an electrical wire or group of them in a cable with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump

wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

Fig 2.8.1: JUMPER WIRES

2.8.2 FEATURES

- Capable of voltage: 5-12v
- Dimensions: 0.00 in x 0.00 in x 0.00 in (0.0 cm x 0.0 cm x 0.0 cm)
- Weight: 1.38 oz (39 g)
- Easy to use
- Easy to interface
- Long life
- Best for Prototyping

2.8.3 APPLICATIONS

- End to end connection in controller and other peripherals.
- It is used in robotics.
- It is used in AVR/8051/PIC/ARM/Arduino/Raspbery-pi based projects.

2.9 BREAD BOARD

2.9.1 BREAD BOARD

A breadboard is a solderless construction buse used for developing an electronic circuit and wiring for projects with microcontroller hoards like Arduino

The middle section of the board has two columns, ench with 30 strips of connector, like the one pulled out and to the side of the breadboard These connect together mything that is pushed through from the front into one of those five holes

On either edge of the board are much longer sections of clip that join together the columns of holes marked by the blue and red lines on the front of the breadboard. These are generally used for GND (blue) and SV (red).

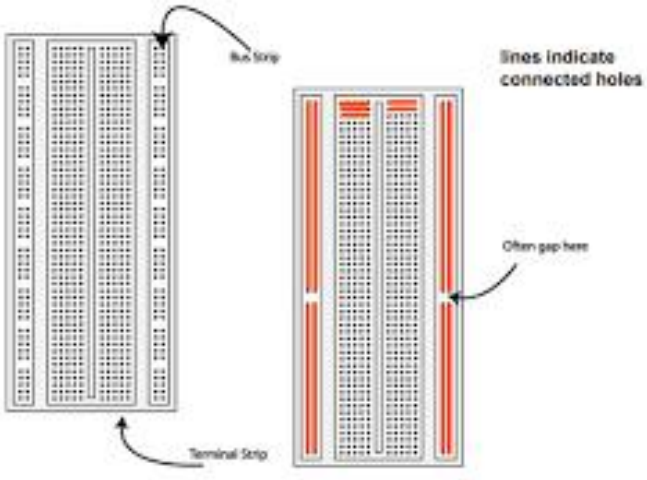


Fig 2.9.1: BREAD BOARD

2.9.2 APPLICATIONS

- Distribution Strips are two
- Wire Size is 21 to 26 AWG wire
- Tie Points are two hundred

- Withstanding Voltage is 1,000V AC
- Tie points within IC are 630
- Insulation Resistance is DC500V or 500MΩ
- Dimension is 6.5*4.4*0.3 inch
- Rating is 5Amps
- ABS plastic through color legend

CHAPTER 3

SOFTWARE DESCRIPTION

3.1 ARDUINO IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (te Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards,

The source code for the IDE in released under the GNU General Public License, version 2. The Anduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program tub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Fig 3.1.1 ARDUNIO IDE SOFWARE

3.2 Installation of ESP8266 board to Arduino IDE:

Step 1. Start Arduino and open the Preferences window.

Step 2. Enter **https://dl.espressif.com/dl/package_esp32_index.json** into the Additional Board Manager URLs field. You can add multiple URLs, separating them with commas.

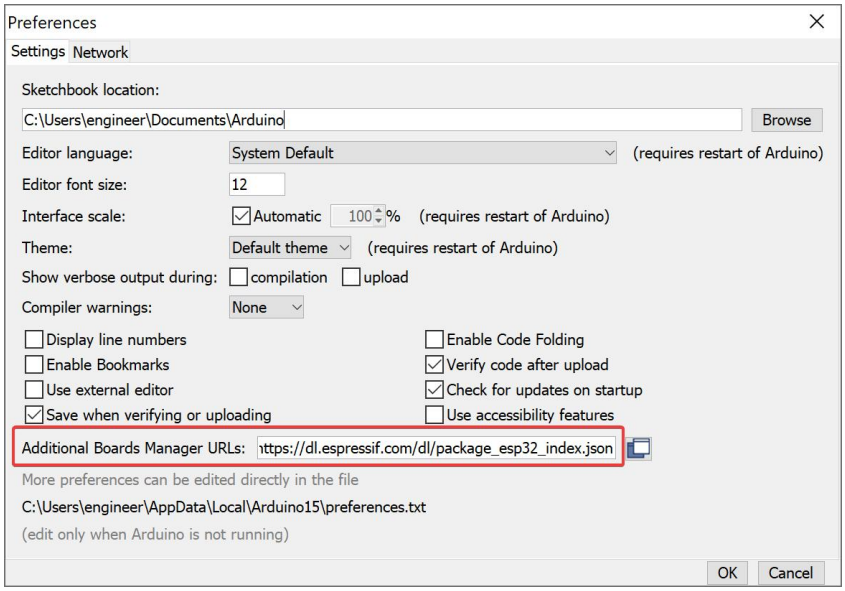


Fig 3.2(2):Arduino IDE Preferences Window

Step 3. After that open Boards Manager from Tools > Board menu and install *esp32* platform.

Select ESP32 board from Tools > Board menu after installation.

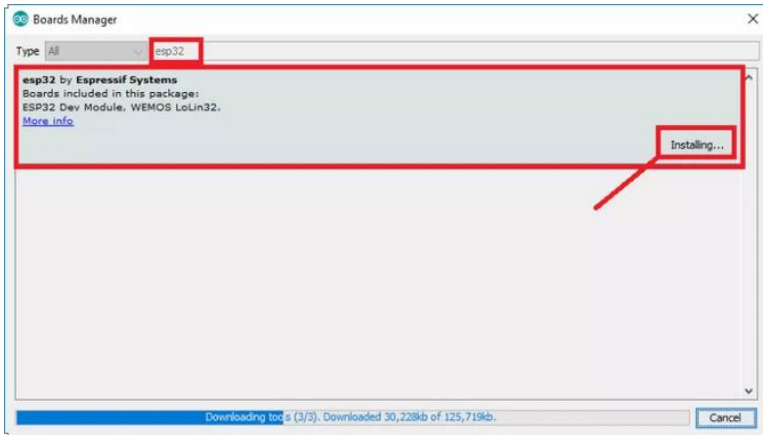


Fig 3.2(3): Boards Manager Arduino IDE

3.3 RESULT WINDOW :

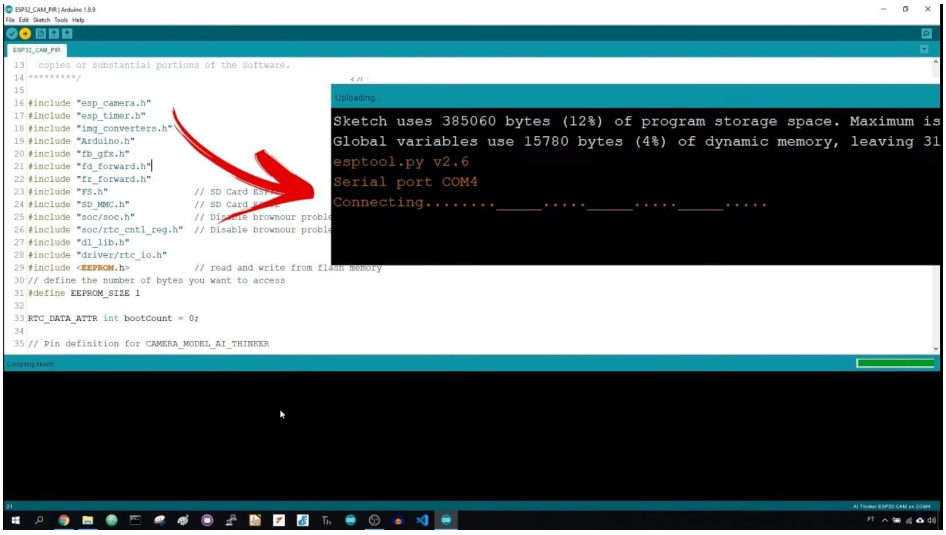


Fig 4.4: RESULT WINDOW

3.4 PROGRAM FLOW CHART

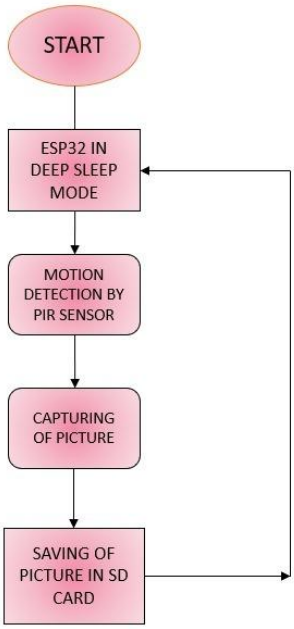


Fig 3.5: Flow Chart

CHAPTER 4

CIRCUIT DESCRIPTION & RESULT

4.1 TECHNICAL DESCRIPTION:

The sensor can detect the presence of intruders. Upon detection of IR, PIR sensor generates the output in the form of electrical signal. Although the output from the sensor is of few volts, it could be amplified to required voltage using amplifier circuit and could be used for actuating lighting system and the webcam. The lamp and webcam could be turned ON when the PIR sensor is activated and could remain OFF when the sensor is idle. This way, the energy consumed by the overall system could be minimized. Also, the cost of system could be far less than the security system available in the market. With this hypothesis, we have proposed a simp le low power PIR based security system.

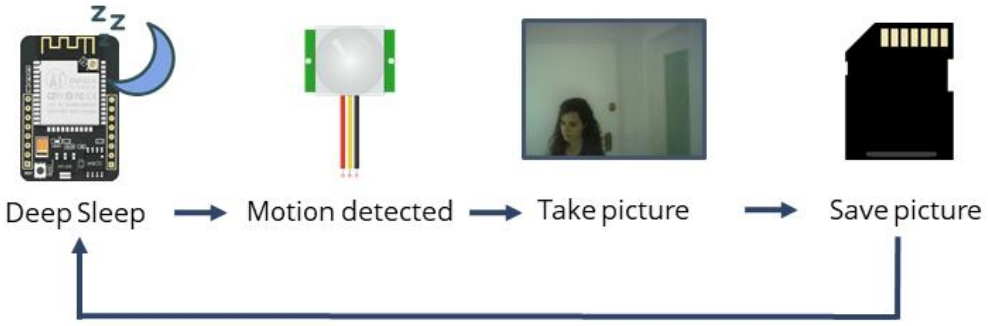


Fig 4.1: Flow chart

4.2 WORKING

I. The software developed is kept running and checks if the cam is turned ON.
II. When an intruder comes in the detection range of the PIR sensor, the sensor generates an outpu t of 3.3 volts.

III. This output is further amplified and is used for activating the relay of the lighting system and the webcam.
IV. Once the lamp and webcam are actuated with the output from the amplifier, software finds the webcam is turned ON.
V. The software starts to capture the photo by the esp32 cam Vol-8 Issue-4 2022 IJARIIE-ISSN(O)-2395-4396 17851 www.ijariie.com 1529.
VI. After the intruder leaves the detection range of the sensor, there is no output from the sensor. Therefore, it turns OFF the webcam. The photo captured will be saved to micro-SD card.
VII. Every time when the intruders come in the detection range of the sensor, the above steps from step 2 to step 6 repeats. PIR sensors have ranges of up to 10 meters (30 feet).

4.3 PROCEDURE

- Take all the components required.
- Connect Rx and Tx pins of FTDI PROGRAMMER Module to U0T and U0R pins of ESP32-CAM
- And connect GPIO 0 of ESP32-CAM to GND of FTDI programmer
- To upload code to the ESP32-CAM board, connect it to your computer using an FTDI Programmer
- Many FTDI programmers have a jumper that allows you to select 3.3V or 5V. Make sure the jumper is in the right place to select 5V.
- Go to **Tools** > **Board** and select **AI-Thinker ESP32-CAM**.
- Go to **Tools** > **Port** and select the COM port the ESP32 is connected to.
- Then, click the upload button to upload the code.
- When you start to see these dots on the debugging window as shown below, press the ESP32-CAM on-board RST button.
- Now connect the circuit to breadboard as per the circuit diagram.

4.4 SCHEMATIC DIAGRAM

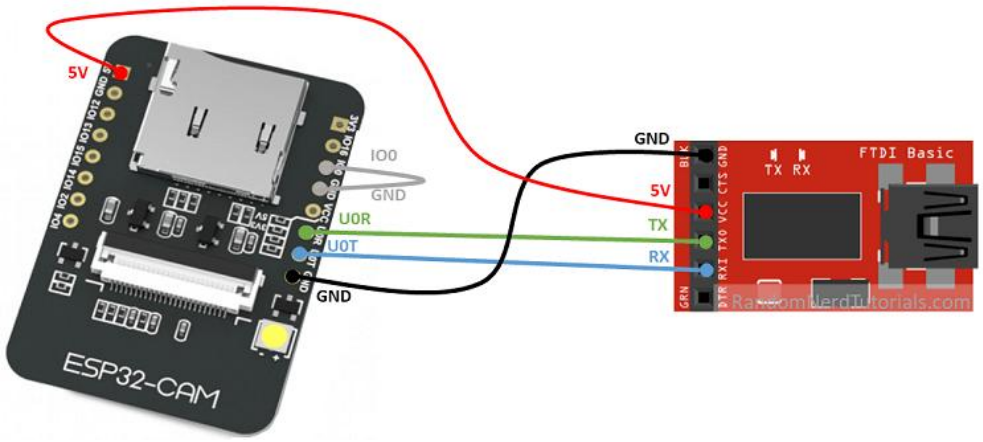


Fig 4.4a: Circuit for programming

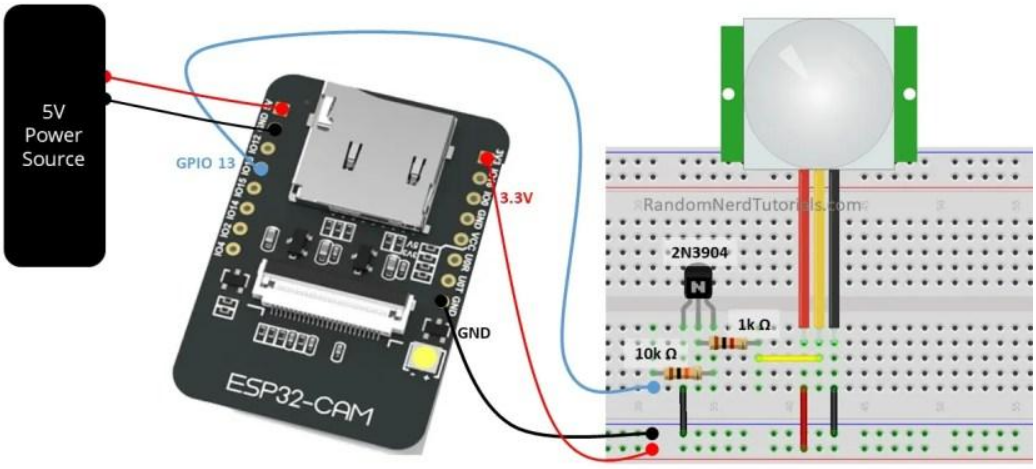


Fig 4.4b: Circuit to Breadboard

4.5 RESULT

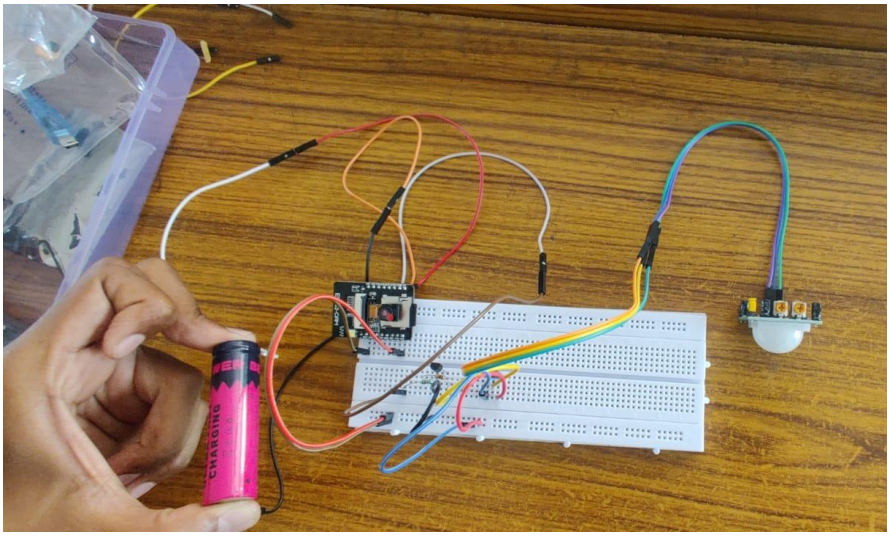
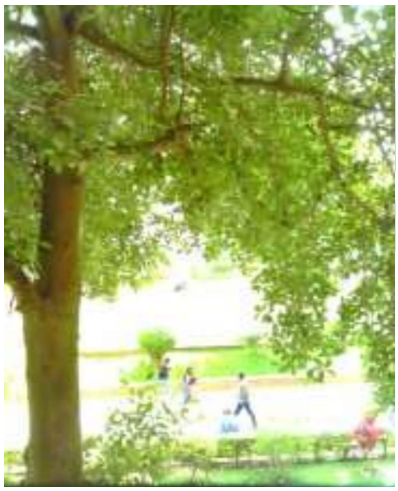
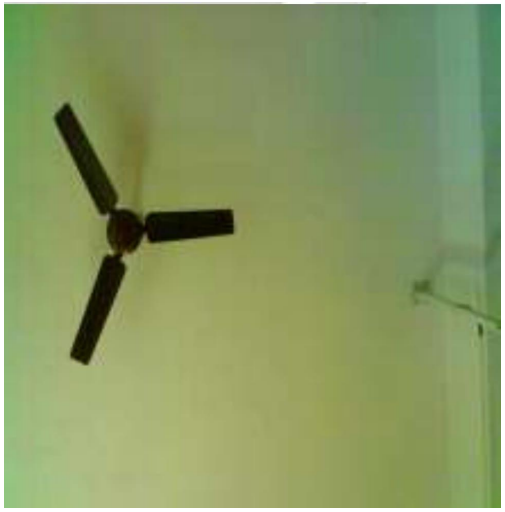


Fig 4.5: circuit on Breadboard





CHAPTER 5

ADVANTAGES & APPLICATIONS

5.1 ADVANTAGES

- By intalling this we can secure the things and reduce the crimes.
- Security of many places like banks,museum and of jewellery shop will be high.
- The chance of hacking is very less
- Very less electricity is needed
- Once it is installed it will work for many years.

5.2 APPLICATIONS

- Used in Home Smart Devices
- Used in Industrial wireless control
- Used in wireless monitoring
- Used in Jewellery Shops
- Used in Museum Security
- Used in Survillance

CHAPTER 6

CONCLUSION & FUTURE SCOPE

6.1 CONCLUSION

Motion capture (Mocap) is an effective 3D animation tool for realistically capturing human motion. In this PIR Sensor Based Security System, we have used low power, low cost PIR sensor that are easy to interface with other components. By using this system, we were able to reduce the power consumed and memory space of the system.Currently, we have used only one webcam in our project which could only capture the area facing to it. The software developed is for the recording of the photo captured by the web cam.Advancement in innovation and technology has significantly improved security in our regular day to day existence. This research focuses on how to configure a simple home security framework using a PIR sensor (Passive Infra-Red) in light of a microcontroller. This safety will work if the PIR (Passive Infra-Red) sensor recognizes any individual that would not like to go into the house, and afterward the micro-controller process and instructs. This work has shown that utilization of Sensors ought to be utilized in shops and homes in Nigeria and African nations in this way. This task can be of acceptable use and be mass produced economically to be applied in different houses, shops, and even in vehicles

6.2 FUTURE SCOPE

To make this extended for larger areas as now it is limited to smaller areas and to make the application much more effective. Integrating a 360 degree webcam to view all the angles.

REFERENCES

[1] ESP32-Cam Pir sensor and photo capture by MS.Jayashree cs, Pradum Singh, Shivanandan NM,Sreelekshmi C,IJARIIE-ISSN(O)-2395-4396,Vol-8 Issue-4 2022.
[2] Real time face detecting unauthorized human movement using ESP32 module by S. Dinakarasu,M.Baskar,C.Balaji,M.Gawtham,e-ISSN: 2395-0056 p-ISSN: 2395-0072Volume: 09 Issue: 06 | June 2022
[3] ESP32-WROOM-32 Datasheet v3.3,Espressif Systems
[4] Burange AW, Misalkar HD. Review of Internet of Things in development of smart cities with data management & privacy. IEEE International Conference on Advances in Computer Engineering and Applications. 2015 July 23;: p. 1.
[5] Wukkadada B, Wankhede K, Nambiar R, Nair A. Comparison with HTTP and MQTT In Internet of Things (IoT). In Proceedings of the
[6] International Conference on Inventive Research in Computing Applications (ICIRCA 2018); 2018; Coimbatore. p. 249-253.
[7] Vikram N, Harish KS, Nihaal MS, Umesh R, Kumar SAA. A Low Cost Home Automation System Using Wi-Fi Based Wireless Sensor Network Incorporating Internet of Things(IoT). In 2017 IEEE 7th International Advance Computing Conference; 2017; Hyderabad. p. 174-179.
[8] Alaa M, Zaidan AA, Zaidan BB, Talal , Kiah MLM. A Review of Smart Home Applications based on Internet of Things. Journal of Network and Computer Applications. 2017; 97.
[9] Agarwal A, Hada N, Virmani D, Gupta T. A Novel Design Approach for Smart Door Locking and Home Security using IoT. A High Impact Factor & UGC Approved Journal. 2017 August; 6(8): p. 1-5.
[10] M. N, Kamat , Shinde D. Smart Door Security Control System Using Raspberry Pi. International Journal of Innovations & Advancement in Computer Science. 2017 November; 6(11): p. 1-4.
[11] Gupta RK, Balamurugan S, Aroul K, Marimuthu R. IoT Based Door Entry System. Indian Journal of Science and Technology. 2016 October; 9: p. 1-5.

[12] http://ieeexplore.ieee.org/document/8256877/
[13] https://ieeexplore.ieee.org/abstract/document/9673226/
[14] https://randomnerdtutorials.com/esp32-cam-pir-motion-detector-photo-capture/
[15] https://youtu.be/LBoM_Uoq_nA
[16] .https://easyelectronicsproject.com/esp32-projects/esp32-cam-pir-motion-detector with-photo-capture-1/
[17] https://www.researchgate.net/publication/347062859_Smart_Home_Monitoring_System_Using_ESP32_Microc ontrollers

APPENDIX

SOURCE CODE

```
#include "esp_camera.h"

#include "Arduino.h"

#include "FS.h"

#include "SD_MMC.h"

#include "soc/soc.h"

#include "soc/rtc_cntl_reg.h"

#include "driver/rtc_io.h"

#include <EEPROM.h>

#define EEPROM_SIZE 1

RTC_DATA_ATTR int bootCount = 0;

#define PWDN_GPIO_NUM     32

#define RESET_GPIO_NUM    -1

#define XCLK_GPIO_NUM      0

#define SIOD_GPIO_NUM     26

#define SIOC_GPIO_NUM     27

#define Y9_GPIO_NUM       35

#define Y8_GPIO_NUM       34
```

```
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22
int pictureNumber = 0;
void setup() {
 WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //disable brownout detector
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
```

```
 config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
  config.pin_pclk = PCLK_GPIO_NUM;
 config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
 config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 20000000;
  config.pixel_format = PIXFORMAT_JPEG;
  pinMode(4, INPUT);
```

```
digitalWrite(4, LOW);
  rtc_gpio_hold_dis(GPIO_NUM_4);
  if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA; // FRAMESIZE_ +
      QVGA|CIF|VGA|SVGA|XGA|SXGA|UXGA
    config.jpeg_quality = 10;
    config.fb_count = 2;
  } else {
    config.frame_size = FRAMESIZE_SVGA;
 config.jpeg_quality = 12;
    config.fb_count = 1;
  }
  esp_err_t err = esp_camera_init(&config);


 if (err != ESP_OK) {
          Serial.printf("Camera init failed with error 0x%x", err);
    return;
  }
```

```
  Serial.println("Starting SD Card");
  delay(500);
  if(!SD_MMC.begin()){
    Serial.println("SD Card Mount Failed");
  }
  uint8_t cardType = SD_MMC.cardType();
  if(cardType == CARD_NONE){
    Serial.println("No SD Card attached");
    return;
    }
  camera_fb_t * fb = NULL;
fb = esp_camera_fb_get();
  if(!fb) {
    Serial.println("Camera capture failed");
    return;
  }
   EEPROM.begin(EEPROM_SIZE);
   pictureNumber = EEPROM.read(0) + 1;
```

```
String path = "/picture" + String(pictureNumber) +".jpg";

 fs::FS &fs = SD_MMC;

 Serial.printf("Picture file name: %s\n", path.c_str());

 File file = fs.open(path.c_str(), FILE_WRITE);

 if(!file){

   Serial.println("Failed to open file in writing mode");

 }

 else {

   file.write(fb->buf, fb->len); // payload (image), payload length

 Serial.printf("Saved file to path: %s\n", path.c_str());

   EEPROM.write(0, pictureNumber);

   EEPROM.commit();

 }

 file.close();

 esp_camera_fb_return(fb);

 delay(1000);

 pinMode(4, OUTPUT);

 digitalWrite(4, LOW);
```

```
  rtc_gpio_hold_en(GPIO_NUM_4);

  esp_sleep_enable_ext0_wakeup(GPIO_NUM_13, 0);

  Serial.println("Going to sleep now");

 delay(1000);

  esp_deep_sleep_start();

  Serial.println("This will never be printed");

}

void loop() {

}
```