Joy-Pi

TABLE OF CONTENTS

The credentials are:
Username: pi
Password: 12345

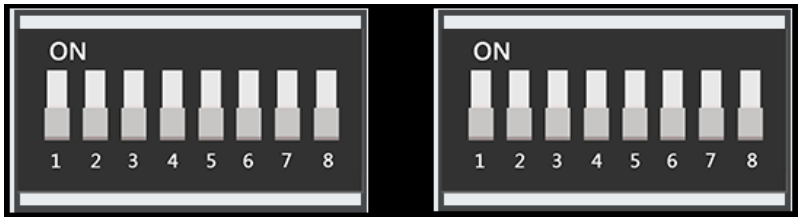
1. OVERVIEW

Dear customer,
thank you very much for choosing our product.
In the following we will show you what has to be observed during start-up and use. Should you encounter any unexpected problems during use, please feel free to contact us.

The following lessons are designed so that, no matter how much previous knowledge you have, you can easily complete all lessons. For the different lessons you have to download sample files and run them on the Joy-Pi. How to do this can also be found in this manual.

But these tutorials are only the beginning.
We look forward to see what you will do with our Joy-Pi.

2. SWITCHING BETWEEN MODULES



On the Joy-Pi board are two switch units with 8 switches. The switches make it possible to switch between different sensors and modules. Since the Raspberry Pi has only a limited number of GPIO pins, these switches are needed to use more sensors and modules than GPIO pins are available.

Using these switches is quite simple and is required in some of the following lessons.

In the table you can see which switch switches which sensor or module:

Sensors / Modules	**Switching unit**	**Switch**
Button Matrix	Left	1 - 8
Independent buttons	Left	5 - 8
Vibration module	Right	1
Tilt sensor	Right	2
Stepper motor	Right	3, 4, 5, 6
Servomotor	Right	7, 8

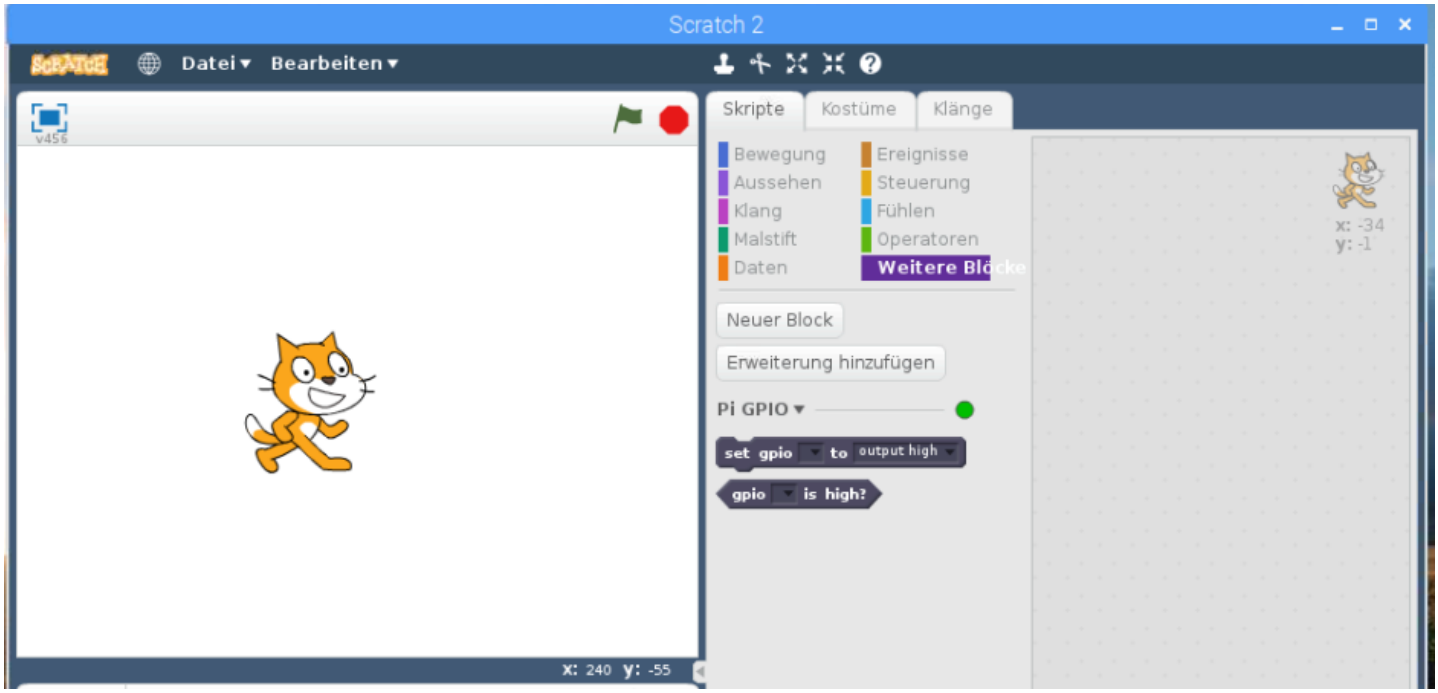
3. SCRATCH

Scratch is a graphical programming language and belongs to the so-called educational programming languages. It was developed at MIT and has the goal to teach people the basics of programming. In the center stands the very simple operation. With an editor, scratch projects can be created and published on the scratch website.
The editor is divided into 3 main areas. The stage, the object catalogue and a detailed view in which the script area is hidden. You can program the stage and any object through this script area using prefabricated blocks. The commands range from movements, appearance and sounds to data, control and feeling.

Scratch is pre-installed on the Rasbian operating systems.

You can go to the official Scratch website here.

4. SCRATCH WITH GPIOS

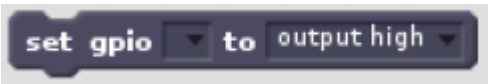


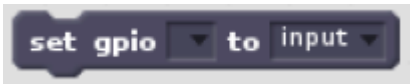
This introduction explains how to add the GPIO function to Scratch and how to use it.

First open Scratch and then click "More Blocks". Now click on "Add Extension" and then on "PI GPIO".

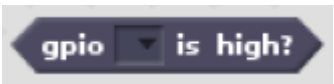
You have now successfully added the GPIO function. Now you should be able to see a new subgroup named "PI GPIO" on the right, as you can see in the picture above. There are two functions:

With the first function you can set pins to "high" or "low".
You can also set a pin as input.





The second function is to query the state of a pin.

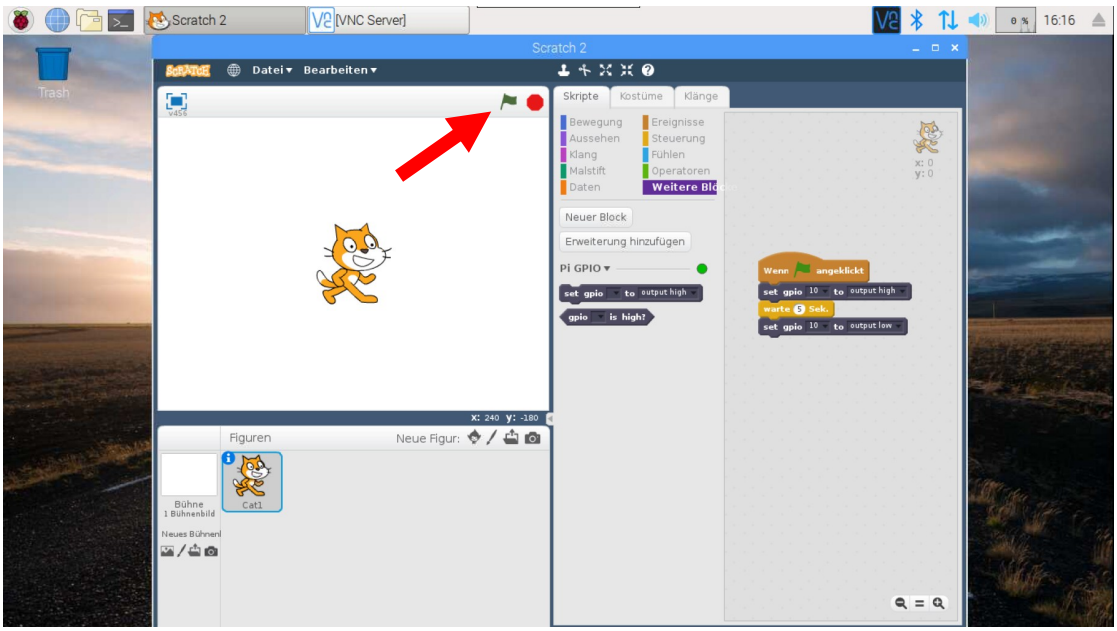


Make sure that you always specify the GPIO.BCM number in both functions and not the pin number.

To get to know Scratch a little better before the first lesson begins, here is a small example.

In this example, if the green flag is clicked, GPIO 10 is set to "high" for 5 seconds and then to "low" again. If, for example, an LED is connected to the pin, the LED, after the flag has been clicked, lights up for 5 seconds and then goes out again.

In the following picture you can see where the flag is, with which, in this example, the program is started.



5. LESSON 1: CONTROL THE BUZZER

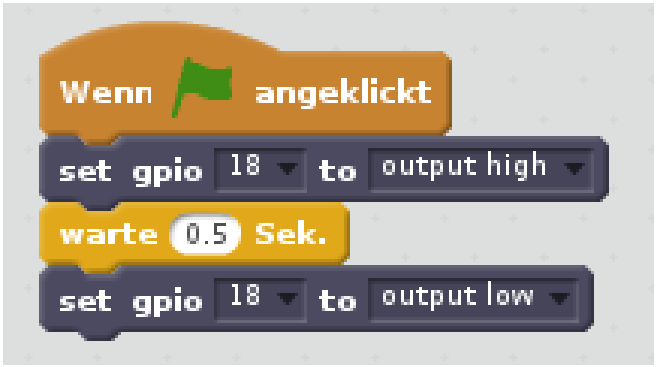
In this lesson you will learn how the buzzer works and how to program it with scratch.
The buzzer on the Joy-Pi is connected to the Raspberry Pi via GPIO18.
In this example, when the flag is clicked, the buzzer pin is set to HIGH for half a second and then to LOW again.

Remove this sticker from the buzzer:

If you want the program to run endlessly, you can also add a "Repeat continuous" loop. In this case the program can only be stopped by clicking on the red circle next to the green flag in Scratch.

Block Structure

1. Click on the green flag to start the program.
2. Set the buzzer to GPIO18 on HIGH
3. Wait 0.5 seconds
4. Set the buzzer to GPIO18 on LOW

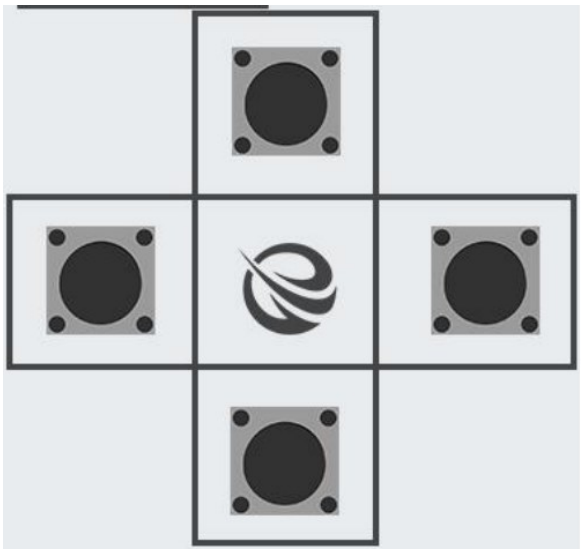


5. LESSON 2: CONTROL THE BUZZER WITH A BUTTON

In this lesson, you will learn how to control the buzzer push button.

The buzzer is connected to pin GPIO18 (output) and the upper button to GPIO26 (input). When the button is pressed (input), the output is set to HIGH (buzzer beeps). When the button is released, the output signal returns to LOW (buzzer off).

The upper button of the button cross is used:



Attention:

The Joy-Pi Board sets the signal to LOW when the button is pressed and to HIGH again when the button is released. Take this into account when creating your own programs.

The switches of the left switch block must all be set to "on".

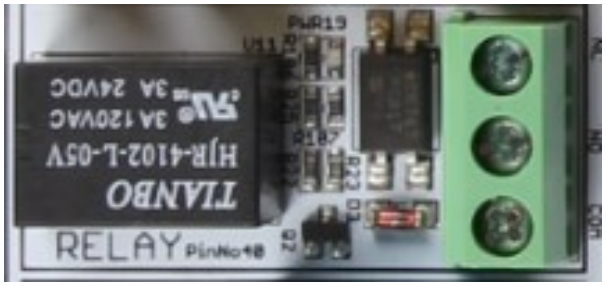
Block Structure

1. Click on the green flag to start the program.
2. GPIO 26 (push-button) as input
3. infinite loop
4. When GPIO26 HIGH (button not pressed)
5. Set GPIO18 LOW (buzzer no sound)
6. Otherwise GPIO18 HIGH (buzzer beeps)

5. LESSON 3: CONTROLLING RELAYS

In this lesson, you will learn how to control the relay using Scratch.

The relay is connected to pin GPIO21. GPIO21 is set to LOW to open the relay. After a pause of one second, GPIO21 is set to HIGH to close the relay.



Attention:

It is very important not to attempt to connect high voltage equipment to the relay (e.g. table lamp, coffee machine, etc.). This could result in electric shock and serious injury.

Block Structure

1. Click on the green flag to start the program.
2. GPIO21 LOW (relay opens)
3. Wait a second.
4. GPIO21 HIGH (relay closes)

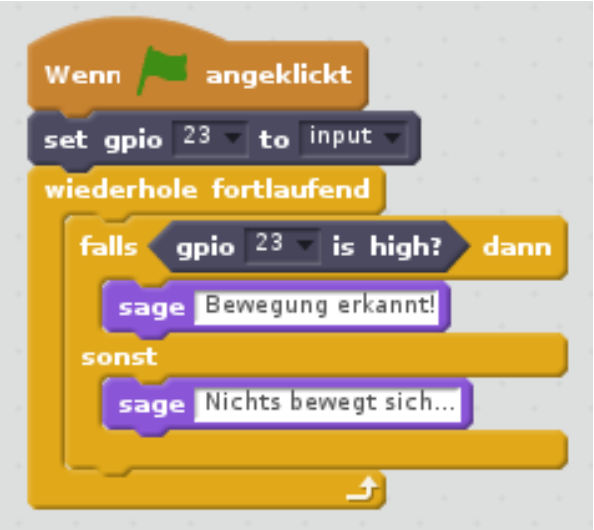
5. LESSON 4: DETECT MOTION

In this lesson, you will learn how to recognize movement using the motion sensor.

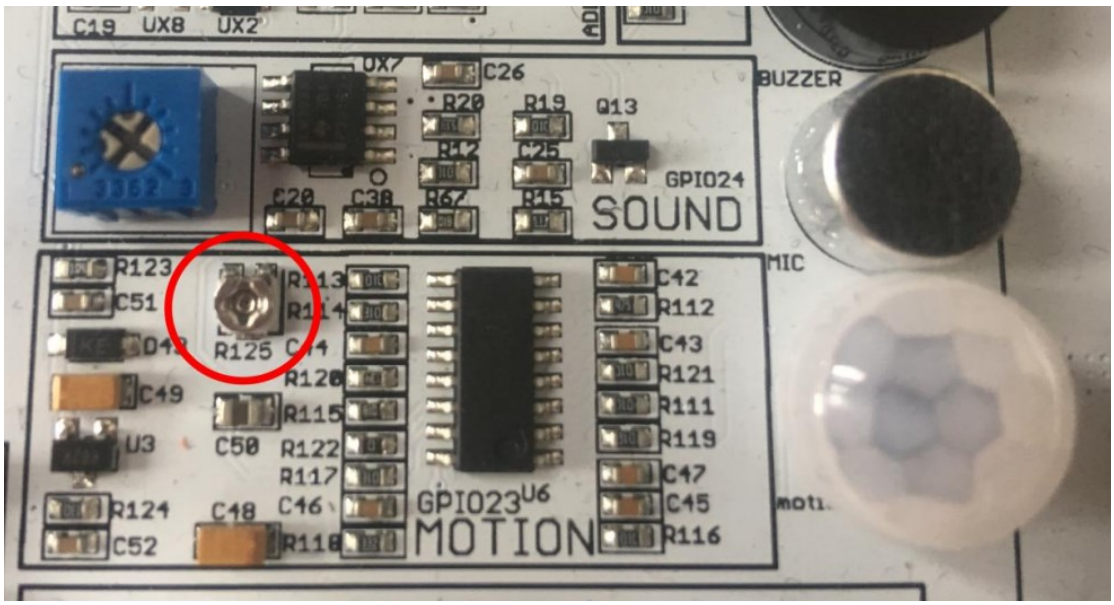
The motion sensor is connected to GPIO23. GPIO23 is configured as an input. The signal from the sensor is processed (as in previous lessons) by an if-then query and a corresponding message is output.

Block Structure

1. Click on the green flag to start the program.
2. GPIO23 as input (motion sensor)
3. infinite loop
4. When GPIO23 HIGH
5. Say "Movement detected!" (sth. moves)
6. Otherwise say, "Nothing moves..." (no movement)



The sensitivity of the motion sensor can be adjusted using the potentiometer circled in red.



5. LESSON 5: DETECT TILTS

In this lesson, you will learn how to use the tilt sensor to determine whether the Joy-Pi is tilted to the right or left.

The tilt sensor is connected to pin GPIO22. If the sensor is tilted to the left, it returns a HIGH signal. If the sensor is tilted to the right the signal is LOW. The signal is then processed in an if-then-loop.

Attention:

Switch 2 of the right switch block must be set to "on".

Block Structure

1. Click on the green flag to start the program.
2. GPIO22 as input (motion sensor)
3. infinite loop
4. When GPIO22 HIGH (Joy-Pi tilted to the left)
5. Say "left"
6. Otherwise say "right" (Joy-Pi tilted to the right)

5. LESSON 6: RECOGNIZE TOUCH

In this lesson you will learn how to recognize touch with the touch sensor.

The touch sensor is connected to GPIO17. As in previous lessons, GPIO 17 is configured as an input. In this example, the buzzer is set to HIGH when the touch sensor is no longer touched, when the sensor is touched, the buzzer is set to LOW and goes off.

Block Structure

1. Click on the green flag to start the program.
2. GPIO17 as input (touch sensor)
3. infinite loop
4. If GPIO17 HIGH (touch detected)
5. Set GPIO18 to LOW
6. Otherwise (no touch detected)
7. Set GPIO 18 to HIGH

5. LESSON 7: CONTROLLING THE VIBRATION MOTOR

In this lesson you will learn how to control the vibration motor.

The vibration motor is connected to pin GPIO27. For the vibration motor to vibrate, the signal must be HIGH. If the signal is set to LOW, the motor stops again.

To use the vibration motor, the switches of the right switch block must be set to "On".

Block Structure

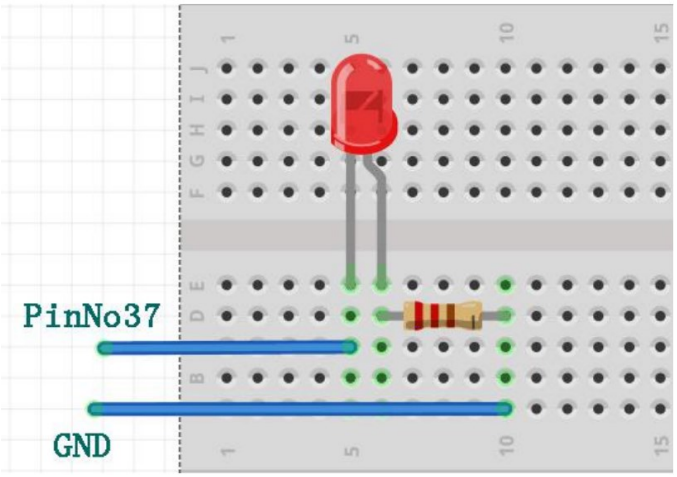
1. Click on the green flag to start the program.
2. GPIO27 HIGH (vibration motor on)
3. Wait 0.5 seconds
4. GPIO27 LOW (vibration motor off)

5. LESSON 8: FLASHING LED

In this lesson you will learn how to create a simple electronic circuit to control an LED. You will then use the Scratch program to make the LED flash.

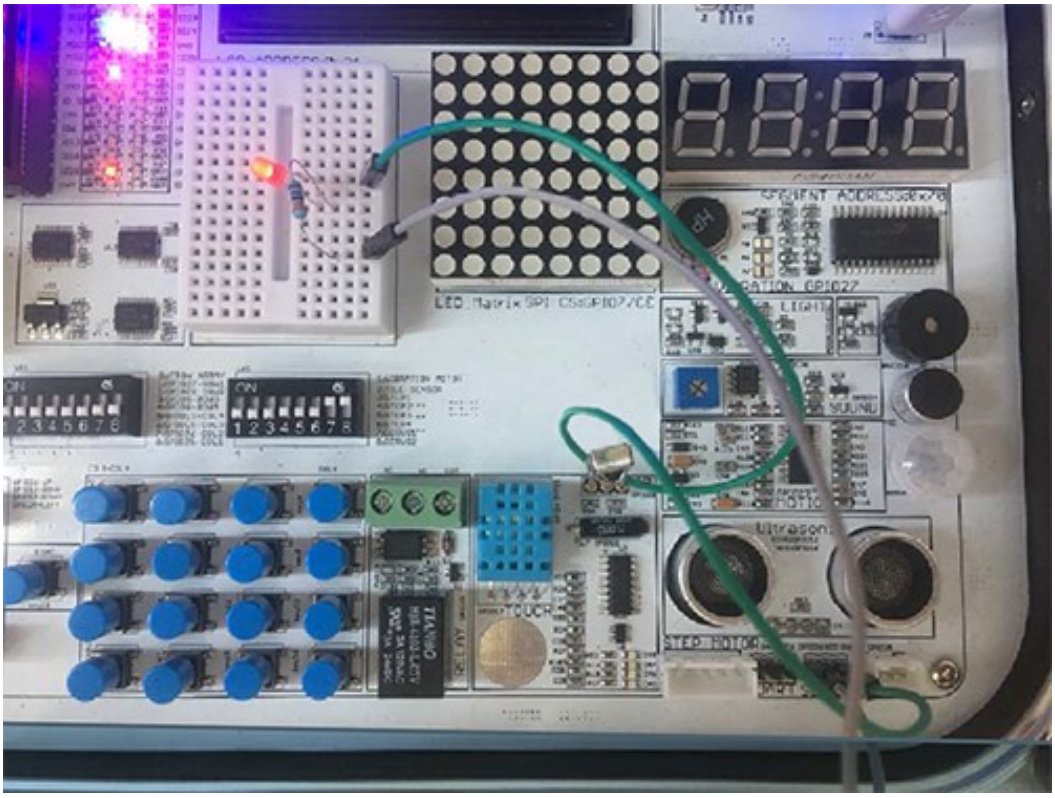
This lesson uses pin GPIO26. The peculiarity is that this pin actually belongs to the servo connector, but for this lesson it is used for other purposes.

The circuit is mounted on the experimental board as shown in the illustration on the right. Pin 37 is GPIO26 and can be found at servo 1 connection. GND is on the same connector. It must also be noted that the LED must be used with the correct polarity. The resistor serves to protect the diode from excessive currents.



For this lesson, the switches on the right switch block must be set to "ON".

The result should look like this:



Now only the code block has to be created. We will set pin GPIO26 HIGH within an endless loop to turn on the LED. This will be followed by a 0.3 second pause. Then the LED is switched off again by GPIO26 LOW and another 0.3 second pause follows. The result is a flashing LED.

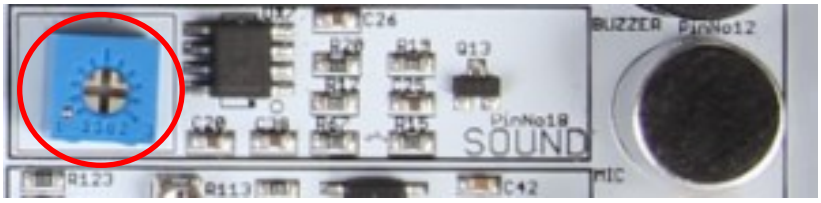
Block Structure

1. Click on the green flag to start the program.
2. infinite loop
3. GPIO26 HIGH (LED on)
4. Wait 0.3 seconds
5. GPIO26 LOW (LED off)
6. Wait 0.3 seconds

5. LESSON 9: DETECT NOISES

In this lesson, you will learn how to use the sound sensor to detect sounds in the environment.

The sound sensor is connected to pin GPIO24. The signal is LOW when a sound is detected. If the environment is quiet, the signal is HIGH.



The sensitivity of the sound sensor can be adjusted using the potentiometer circled in red. This can help if the sensor always or never detects a sound.

Block Structure

1. Click on the green flag to start the program.
2. GPIO24 as input (sound sensor)
3. infinite loop
4. When GPIO22 HIGH (no noise detected)
5. Say "No noise detected..."
6. Otherwise say "Noise detected!" (Noise detected)

6. SUPPORT

We also support you after your purchase. If you still have questions left or encounter any problems, we are also available by e-mail, telephone and ticket support system.

E-Mail: service@joy-it.net
Ticket-System: http://support.joy-it.net
Phone: +49 (0)2845 98469 – 66 (10 - 17 o‘clock)

Please visit our website for more information:

www.joy-it.net