WIBO-THINK Wi-Fi Robot Controller Module

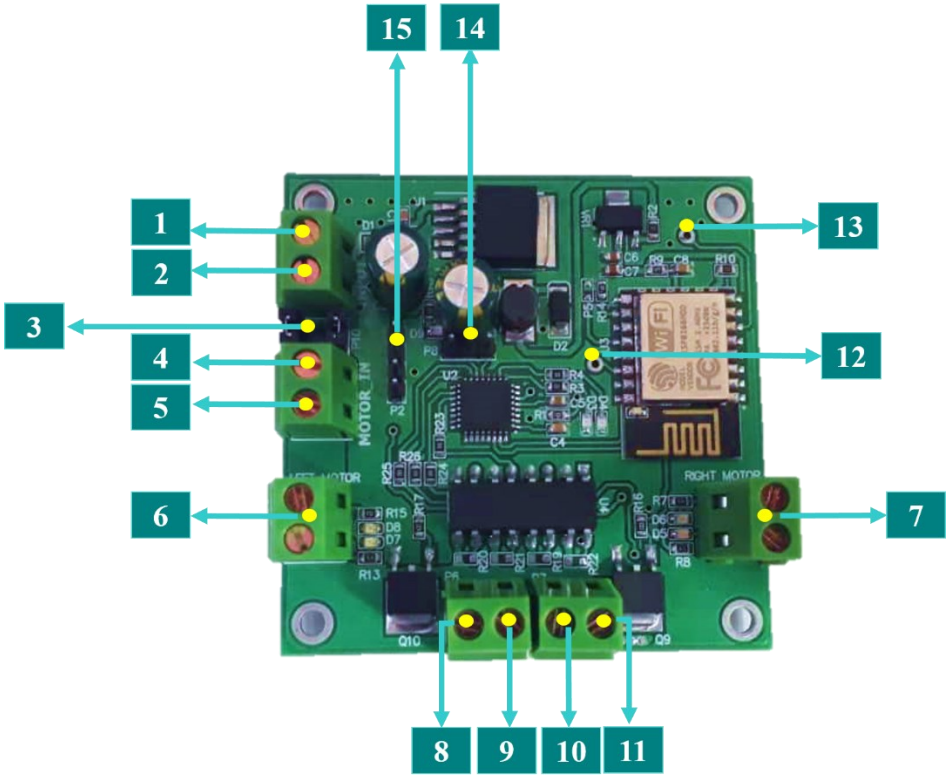
1 Introduction

Thank you for purchasing WIBO-THINK!

WIBO-THINK is a wireless motor driver/robot controller module which controls two DC motors through Wi-Fi. It is perfect for IoT, robotics, home automation and other projects with wireless motor control. This module uses ESP-12F module which is based on ESP8266EX for Wi-Fi connection. WIBO-THINK comes with an android app which offers the opportunity to connect to WIBO-THINK and control a robot remotely. Besides, it is possible to monitor the output of two analog sensors connected to two ADC channels. In addition to two motors, two PWM outputs of the board are also controlled by the WIBO-THINK app.

2 Description

2-1: Pin Description



#1 is the power supply for the logic section of the board which can be anywhere between 6V to 30V (12V is recommended).

#2 is the ground pin for the logic supply.

#3 is two jumpers to connect the logic power terminal to motor power terminal and power all parts of the board with a single power supply.

#4 is the power pin for motor supply.

#5 is the ground pin for motor supply.

#6 is the left motor terminal.

#7 is the right motor terminal.

#8 is a negative supply pin of the first PWM output.

#9 is a positive supply pin of the first PWM output.

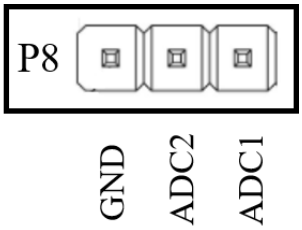
#10 is a positive supply pin of the second PWM output.

#11 is a negative supply pin of the second PWM output.

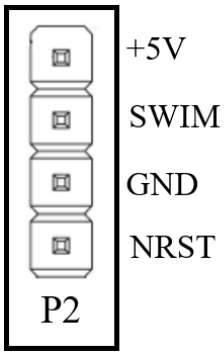
#12 is GPIO5 pin of the ESP-12F Wi-Fi module.

#13 is GPIO12 pin of the ESP-12F Wi-Fi module.

#14 provides two ADC inputs. The pin arrangement is as follows:



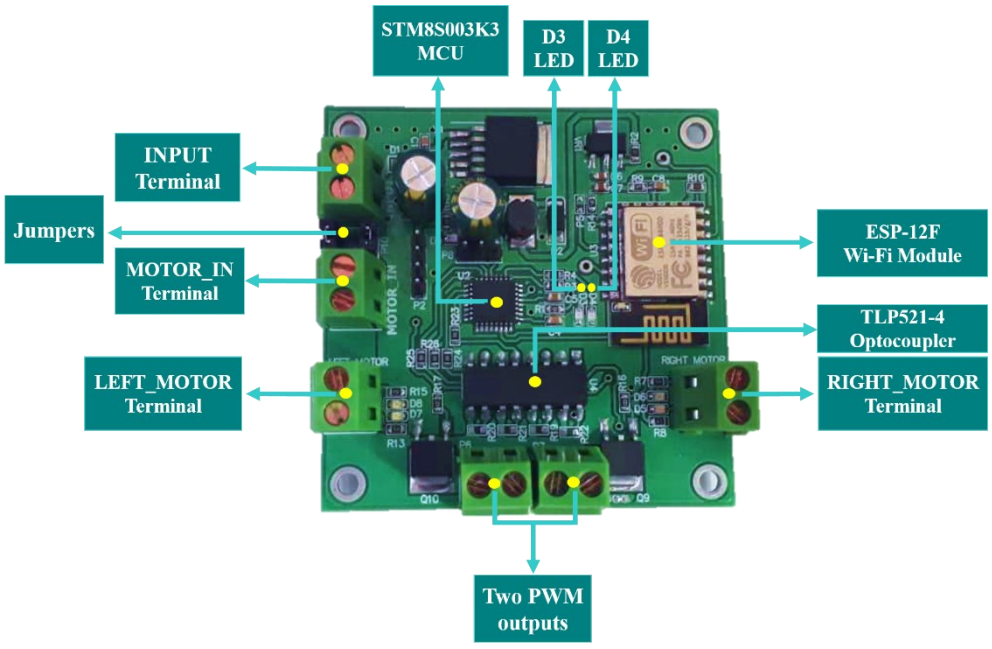
#15 is a SWIM connector to program the STM8S003K3. The pin arrangement is as follows:



2-2: Features

- Has two separate terminals for logic and motor supplies which can be connected with two jumpers.
- Supports 6~36V DC logic supply and 0~18V DC motor supply.
- Has two ADC inputs (0-5V) represented with/without applying a mathematical function in WIBO-THINK app.
- Has a H-bridge motor driver with maximum current of 10A.
- Supports two DC motors rated at up to 18V at 10A.
- Has ESP-12F Wi-Fi module.
- Can be completely controlled over a TCP/IP connection.
- Has STM8S003K3 microcontroller as main chip.
- Has an android application which is also named WIBO-THINK.
- Low power consumption.
- Has two PWM outputs.
- Has status LEDs to indicate motors rotation direction and speed.

2-3: Hardware Configuration



To isolate the logic and motors supplies, WIBO-THINK is powered from two screw terminals which are labeled “INPUT” and “MOTOR-IN”. These two terminals are used for logic and motor supplies, respectively. The “INPUT” terminal is protected from reverse polarity and can be applied power up to 36V DC. DC motors with up to 18V nominal voltage can be powered by the “MOTOR-IN” terminal. There are two jumpers on the board to disable the MCU and motors isolation and apply a single power supply for both.

The WIBO-THINK uses STM8S003K3 microcontroller to implement certain functions. The STM8S003K3 value line 8-bit microcontrollers offer 8 Kbytes of Flash program memory, plus integrated true data EEPROM. To program the MCU, there is a 4-pin header marked “P2” on the board which can be connected to SWIM connector of ST-Link/V2 programmer. Two ADC inputs of the STM8S003K3 are accessible through 3-pin header labeled “P8”.

Wireless internet access is added to the WIBO-THINK by using ESP-12F. The core of ESP-12F is ESP8266EX which is a high integration wireless SoC. ESP-12F interacts with STM8S003K3 with the UART interface. Commands are sent to WIBO-THINK over a TCP/IP link. Commands are defined to read/write four PWM outputs, read two ADC inputs and set/read status of two LEDs. GPIO12 and GPIO5 of the Wi-Fi module are brought out to make further use of the capabilities of it. It is noteworthy that the maximum current allowed to draw per pin of the ESP-12F is 15mA.

In addition to separate power terminals, there is an optocoupler on the board to prevent noise interference on digital signals. TLP521-4 provides 4 isolated channels for 4 PWM signals. A H-bridge motor driver uses the isolated PWM signals to drive two motors. The H-bridge driver consists of 4 N-channel and 4 P-channel MOSFETs with maximum current of 10A. Two screw terminals marked “LEFT_MOTOR” and “RIGHT_MOTOR” are motor outputs which can be connected to motors rated at up 18V and 10A.

Two TIM1 channels of MCU are in access as two PWM outputs which are marked “P7” and “P8” on the board. These two screw terminals would be used to control devices with maximum voltage of module supply voltage.

The board has seven LEDs. D9 indicates that 5V is regulated correctly. D3 and D4 are controlled by two I/O pins of MCU (PB7 and PB6) with regard to user application. D5 and D6 indicate the right motor rotation direction and speed. One of the LEDs is for clockwise the other is for counter-clockwise rotation. The light intensity of LEDs shows the motor speed i.e., as the motor rotates faster the intensity increases. Similar to right motor, two LEDs are for the left motor which are marked D7 and D8.

2-4: Specifications

Parameter	Min.	Typ.	Max.	Unit
Logic power supply	6	12	36	V
Motor power supply		12	18	V
Output current			10	A
Operating temperature	-20		85	°C

3 Instruction

3-1: Getting Started

Here, there is a simple example to get started with WIBO-THINK. After following 9 steps WIBO-THINK will connect to the app and be ready to wirelessly work.

1. Install the WIBO-THINK app on your mobile phone.
2. Connect a 12V DC power supply to INPUT terminal.
   Note: Here a single power supply is used for both controller and motors. Be noticed, the power supply should provide the starting current of the motors.
3. Connect two supply terminals with jumpers.
4. Connect two 12V DC motors to left and right motor terminals.
5. Turn the power supply on.
6. Search for available networks with your mobile phone and connect to "WiboThink". The password is "88888888". Static IP of the WIBO-THINK is 192.168.4.1.
7. Open the app. To add a new panel, press the plus icon "+" at the bottom-right corner of the screen.
   Finally, if you take all previous steps correctly, two numbers at the top of the screen will change and now you can control your board by WIBO-THINK app.

3-2: Commands

As said before, WIBO-THINK can be controlled remotely by sending and receiving specific commands using TCP/IP protocol. The port number is **8080** and the module IP address is **192.168.4.1**. All commands have a general format consisting of 9 bytes which are as follows:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Data1	Data2	Data3	Data4	Checksum	EoF
0xaa								0x55

Byte 1 **Header:** The command frame start byte. This byte is constant and equals to 0xaa.

Byte 2 **Reserved:** The reserved byte. This byte is not used currently and must kept clear.

Byte 3 **Command:** The command byte. This byte is specific for each command.

Byte 4…7 **Data1…4:** The command required data. These 4 bytes are used according to the command and might be not used in some commands.

Byte 8 **Checksum:** The checksum byte. This byte is XOR of 7 previous bytes.

Byte 9 **EoF:** The command frame last byte. This byte is constant and equals to 0x55.

Five commands were defined to set and read the PWM channels, set and read the status of two LEDs and read the ADC inputs. The format of each default command is explained below.

1. **Setting PWM Output Channels**
   This command is used to set the four PWM outputs. Its command byte is **0x14**.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Ch1_PWM	Ch2_PWM	Ch3_PWM	Ch4_PWM	Checksum	EoF
0xaa	0x00	**0x14**						0x55

Byte 4 **Ch1_PWM:** Determines the rotation direction and speed of the right motor.

This byte is a number between 0 and 255.

0-127: determines the right motor CW rotational speed.

128-255: determines the right motor CCW rotational speed.

Byte 5 **Ch2_PWM:** Determines the rotation direction and speed of the left motor.

This byte can be a number between 0 and 255.

0-127: determines the left motor CW rotational speed.

128-255: determines the left motor CCW rotational speed.

Byte 6 **Ch3_PWM:** Determines PWM voltage of the Q10 gate.

The first PWM output is controlled by Q10 MOSFET.

This byte is a number between 0 and 255.

Byte 7 **Ch4_PWM:** Determines PWM voltage of the Q9 gate.

The second PWM output is controlled by Q9 MOSFET.

This byte is a number between 0 and 255.

2. Reading PWM Output Channels

This command is used to read the four PWM outputs. Its command byte is **0x19**.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Reserved	Reserved	Reserved	Reserved	Checksum	EoF
0xaa	0x00	**0x19**	0x00	0x00	0x00	0x00		0x55

Once the above frame is sent to WIBO-THINK, the module returns a frame with value of four PWM channels.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Ch1_PWM	Ch2_PWM	Ch3_PWM	Ch4_PWM	Checksum	EoF
0xaa	0x00	**0x19**						0x55

By reading byte 4 to byte 7 of the above frame, the value of four PWM channels is achieved respectively.

3. **Setting LED status**
   This command is used to set the status of two LEDs. Its command byte is **0x31**.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	LED1_State	LED2_State	Reserved	Reserved	Checksum	EoF
0xaa	0x00	**0x31**			0x00	0x00		0x55

Byte 4 **LED1_State:** Determines the state of the LED marked D4.

0x00: The LED is off.

0xFF: The LED is on.

Byte 5 **LED2_State:** Determines the state of the LED marked D3.

0x00: The LED is off.

0xFF: The LED is on.

Byte 6 and byte 7 **Reserved:** Reserved. These bytes should be cleared.

4. **Reading LED Status**
   This command is used to read the status of two LEDs. Its command byte is **0x38**.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Reserved	Reserved	Reserved	Reserved	Checksum	EoF
0xaa	0x00	**0x38**	0x00	0x00	0x00	0x00		0x55

Byte 4…7 **Reserved:** Reserved. These bytes should be cleared.

Once the above frame is sent to WIBO-THINK, the module returns a frame with status of two LEDs. This frame is as follows.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	LED1_State	LED2_State	Reserved	Reserved	Checksum	EoF
0xaa	0x00	**0x38**			0x00	0x00		0x55

Byte 4 **LED1_Stat:** Determines the status of the LED marked D4.

0x00: The LED is off.

0xFF: The LED is on.

Byte 5 **LED2_Stat:** Determines the status of LED marked D3.

0x00: The LED is off.

0xFF: The LED is on.

Byte 6 and byte 7 **Reserved:** Reserved. These bytes should be cleared.

5. **Reading ADC Inputs**
   This command is used to read the ADC inputs. Its command byte is **0x25**.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	Start/Stop	Single/Continues	Reserved	Reserved	Checksum	EoF
0xaa	0x00	**0x25**			0x00	0x00		0x55

Byte 4 **Start/Stop:** ADC reading start/stop.

0x00: Start ADC reading.

0xFF: Stop ADC reading in Continues mode.

Byte 5 **Single/Continues:** Single or Continues ADC reading mode.

Single and Continues modes are started by clearing the Start/Stop byte.

0x00: Single mode. In this mode, just one conversion of ADC is received.

0xFF: Continues mode. In this mode, the ADC conversion is received one after another. To stop receiving data, set all bits of the Start/Stop byte.

Byte 6 and byte 7 **Reserved:** Reserved. These bytes should be cleared.

Once the above frame is sent to WIBO-THINK, the module returns a frame with two ADC channels data. The response frame is as follows.

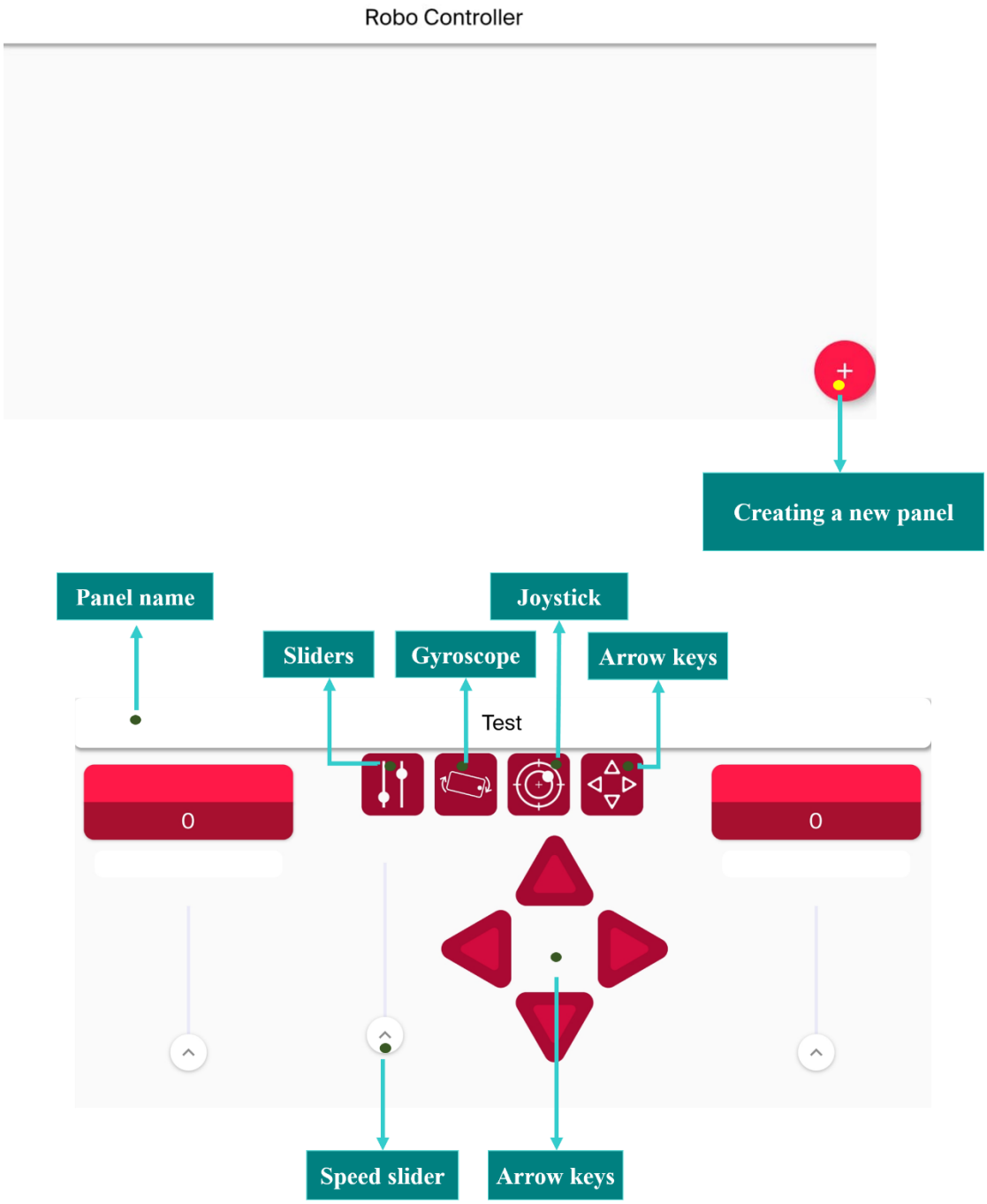
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Header	Reserved	Command	ADC1_L	ADC1_H	ADC2_L	ADC2_H	Checksum	EoF
0xaa	0x00	**0x25**						0x55

3-3: Android App

The WIBO-THINK module comes with an android application to control the module over Wi-Fi. The main platform of the WIBO-THINK app is designed for controlling a robot. In this platform, a robot can be guided with arrow keys, gyroscope, joystick or sliders. Besides, two ADC values are shown and two PWM outputs can be changed with two sliders. By changing the setting of the platform, a new panel is created on the home page of the app. Therefore, you can control each robot with its own panel.

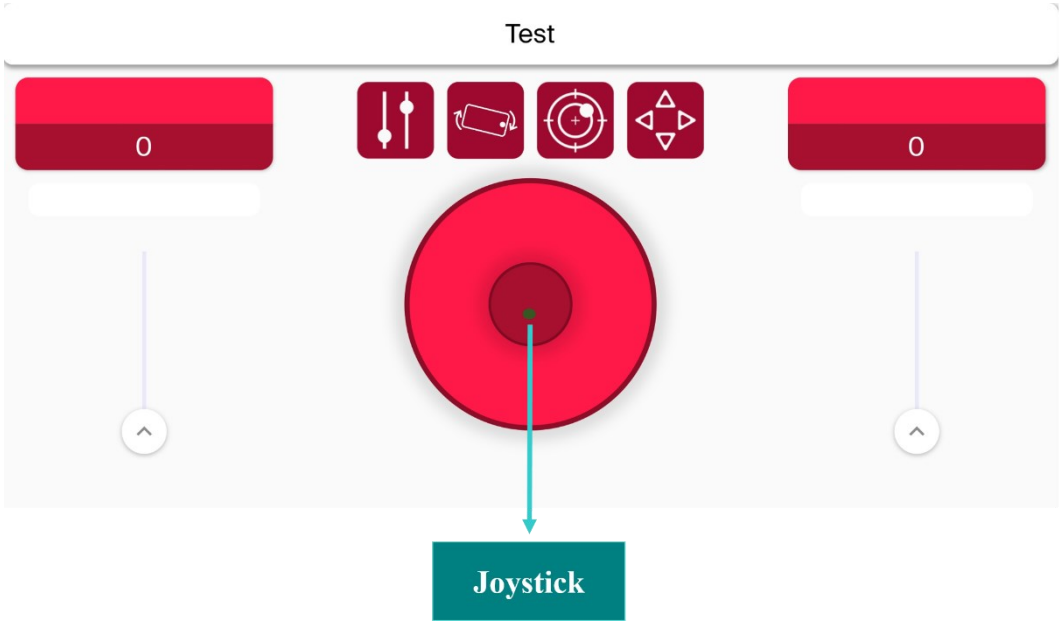
After installing the app, 

its icon appears among your mobile apps. Once you press the icon, after a splash page, the home page appears. By pressing the plus icon "+" at the bottom-right, the platform screen displays. The created panels will be added in the home page.

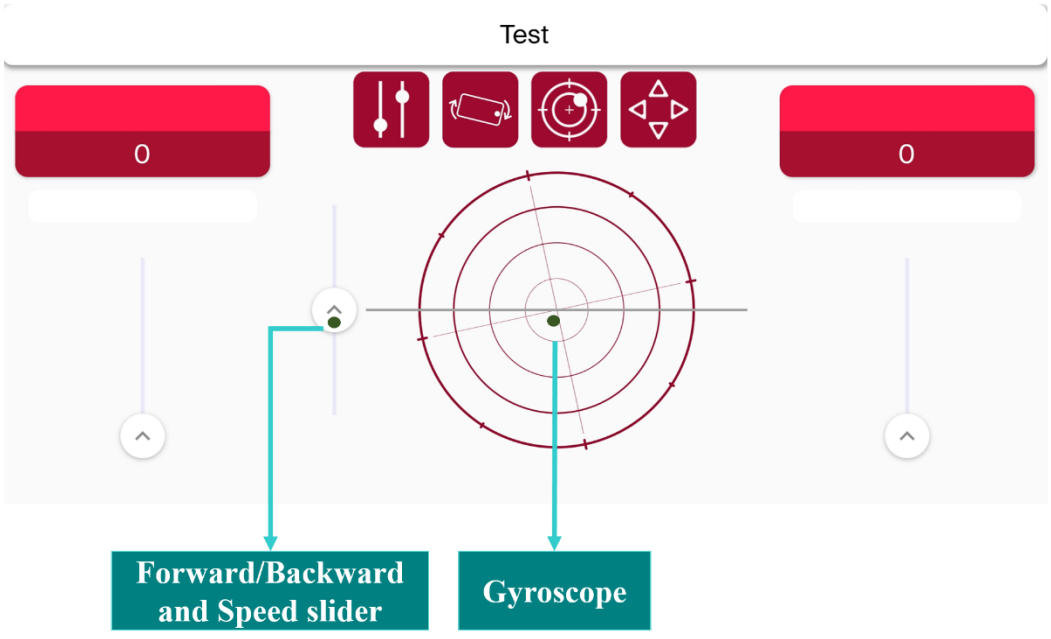


The default name of the platform is Test. You can change the name for your robot panel. You can choose the way of controlling the robot by pressing one of four icons at the top of the screen. The first one is arrow keys. The robot is guided to four directions by four keys and the robot speed is

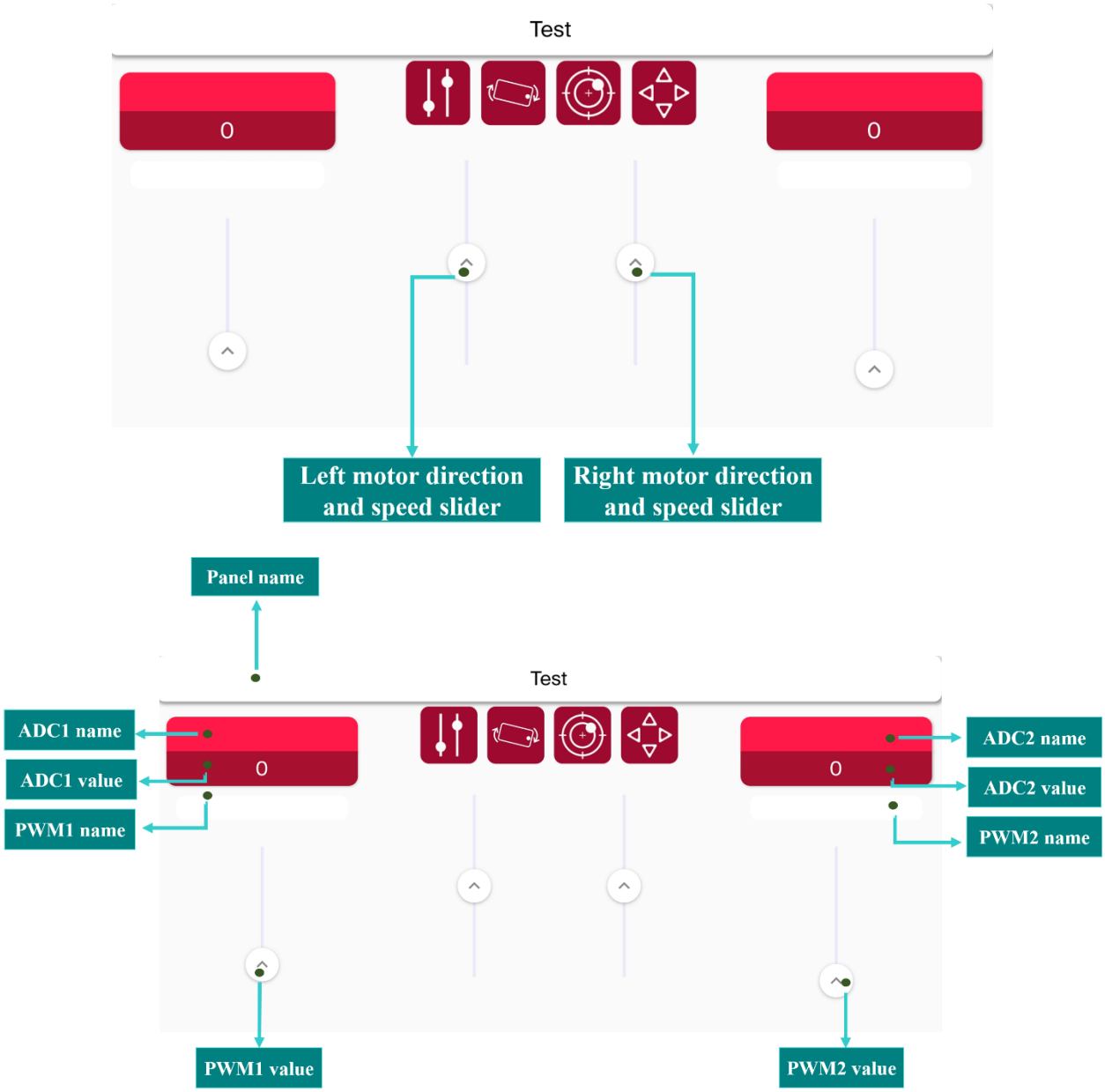
changed by a slider. The next icon is joystick. The robot is simply controlled by moving the central circle.



The third icon is gyroscope. By choosing this icon, you can guide your robot to the left and right with gyroscope. The robot moves forward/backward by pressing the slider toward up/down. The slider is also used to change the speed of the robot.

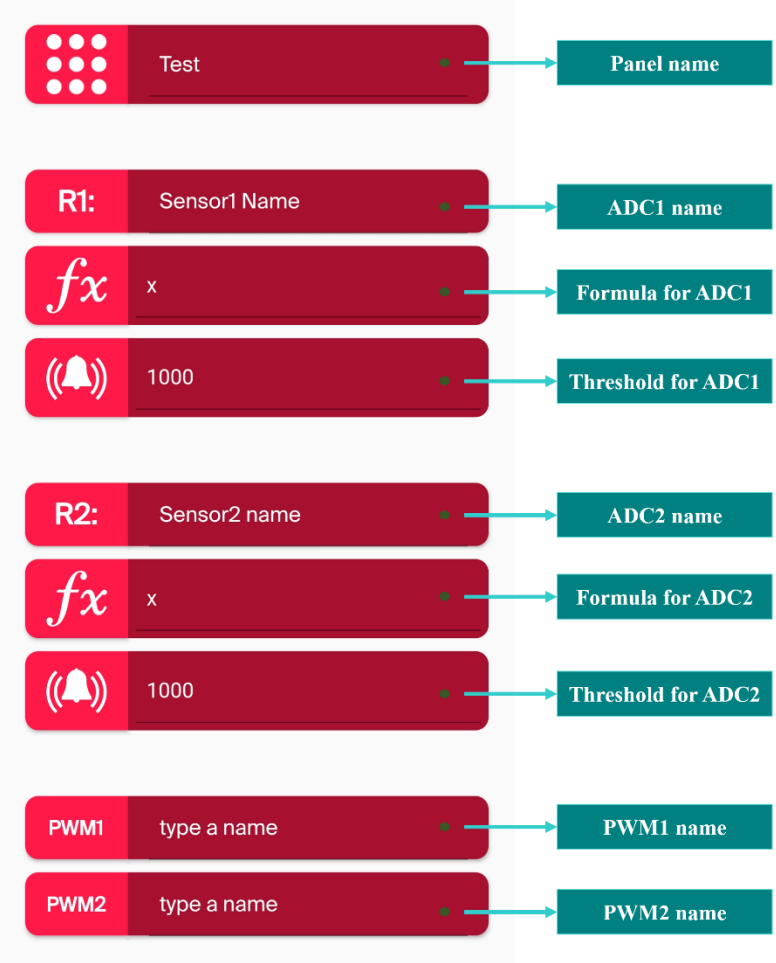


The fourth icon is sliders. There is one slider to change the direction and speed of each motor. By pressing two sliders upward/downward, two motors move forward/backward. To turn the robot left, it is needed to move just the right motor. To turn the robot right, it is needed to move just the left motor.

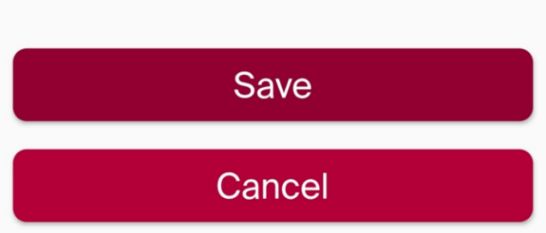


Two analog inputs and outputs are shown at two sides of the screen. Two analog inputs are shown in two red boxes at the top-right and top-left of the screen. Two PWM outputs are set by two sliders at the bottom of the ADC boxes. Each ADC and PWM channels can have a name.

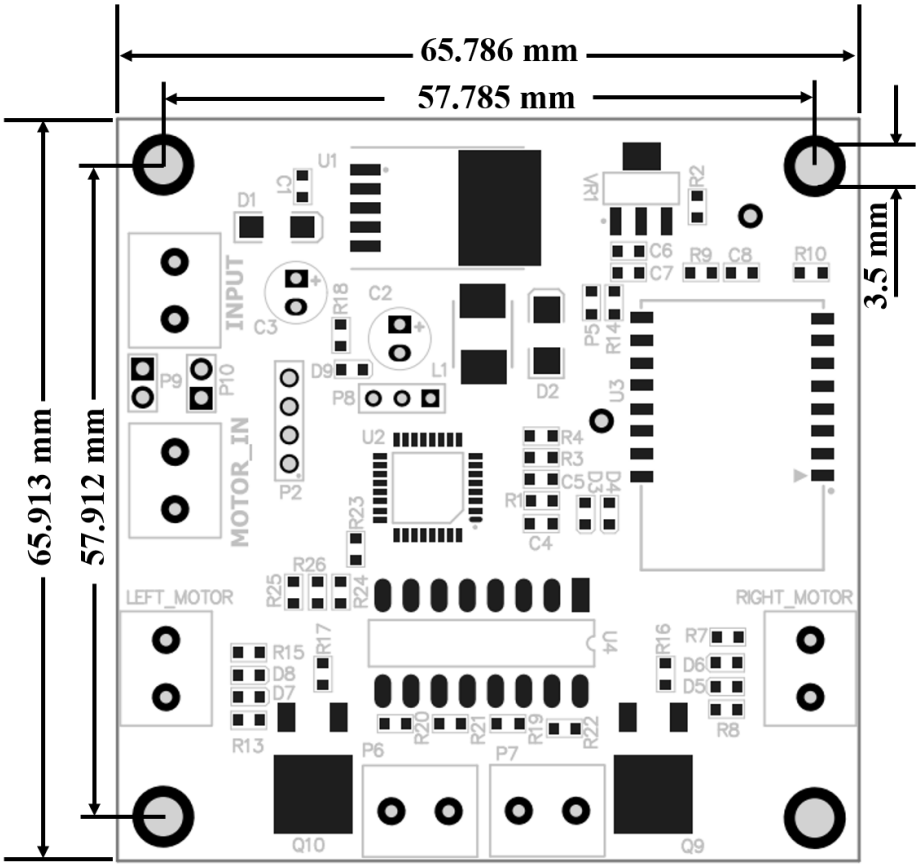
By pressing one of the ADC boxes the setting menu will open. In this menu, you can choose a name for the panel, each ADC channels and each PWM outputs. Besides, you can apply a formula to the analog inputs. For example, by typing “x+100” in the formula box, the ADC value plus 100 is shown in the value box. In addition, it is possible to set a threshold for two analog inputs and if the ADC value crosses the threshold the mobile phone will vibrate.



Once the setting is saved, the panel is added to the home page.



4 Mechanical Dimensions



5 Contact Information

Email: info@robothinkco.com

Web: www.robothinkco.com