# kidsbits WiKi

kidsbits WiKi

Apr 25, 2024

## KD3021 KIDSBITS SMART FARM KIT COMPATIBLE WITH LEGO

1	1. Download	ling the code:	3
2	2. Introduct	ion:	5
3	3. Kit List		7
4	KidsBlock 7	utorial	15
	4.1 1. Ma	inboard_Introduction:	15
	4.2 2. Kid	sBlock Development Environment Configuration:	15
	4.3 3. Pro	jects:	27
	4.3.1	Project 01: Lighting System	27
	4.3.2	Project 02: Light Controlled System	78
	4.3.3	Project 03: Automatic Feeding System	96
	4.3.4	Project 04: Anti-theft Alarm System	129
	4.3.5	Project 05Rainwater Control System	166
	4.3.6	Project 06Temperature and Humidity Control System	190
	4.3.7	Project 07Soil Moisture Detection System	240
	4.3.8	Project 08Water Level Detection System	264
	4.3.9	Project 09Automatic Irrigation System	279
	4.3.10	Project 10 WiFi Web Page Controls Smart Farm System	291

# STEM Education Smart Farm Kit for Arcluino Compatible with Lego



CHAPTER

ONE

### 1. DOWNLOADING THE CODE:

KidsBlock\_Codes

#### CHAPTER

### 2. INTRODUCTION:

Based on the ESP32 IoT, the smart farm integrates multiple sensors such as a photoresistor, a soil moisture sensor, a water level sensor as well as a WiFi wireless communication module in a way that achieves automation, wireless operation and intelligent management. What's more, it is capable of realizing intelligent functions such as sensing, early warning, decision-making and analysis.

Therefore, the product contributes to helping you master how to use sensors to build an IoT system, and how to realize smart farm management via KidsBlock graphical programming.

By the way, detailed projects, sample code and LEGO building are provided in it, thus enhancing our hands-on ability, creativity, scientific and technological innovation awareness as well as problem-solving ability.

### CHAPTER

### THREE

### 3. KIT LIST

#	Components	QTY	Picture
1	kidsIOT Mainboard	1	kiakion kiakion kiakion
2	Button Sensor	1	
3	Steam Sensor	1	Steam sensor
4	PIR Motion Sensor	1	
5	Photoresistor	1	Photoresistance

		able 1 – continued norn previous page
#	Components	QTY Picture
6	Ultrasonic Adapter Board	I VCC I Trig I Echo I GND Ultrasonic
7	Illtrasonic Sensor	
7		
8	White LED Module	
9	Relay Module	
10	Passive Buzzer	
11	Motor	
12	Temperature and Humidity Sensor	Humidity temperature

able 1 – continued from previous page

		Table 1 – continued from previo	us page
#	Components	QTY	Picture
13	GPIO Shield	1	S3 S1 S G G S1 S4 S2 V G G G C UD Expander
14	Servo	1	
15	Water Level Sensor	1	A C C C C C C C C C C C C C C C C C C C
16	Soil Moisture Sensor	1	
17	Weter D	1	
17	Water Pump	1	
18	Water Pipe USB Cable	1	

#	Components	QTY	Picture
			AAAA AAAA AAAAA AAAAAAAAAAAAAAAAAAAAAA
20	Battery Holder	1	
21	20cm Wire	3	
22	30cm Wire	8	
23	F-F DuPont Wire	1	
24	M-F DuPont Wire	1	
25	Slotted Screwdriver	1	<ul> <li>Interface on control induction in the factor of the factor</li></ul>
26	Sink	2	
27	Fan	1	
28	AA BatteryNot provide	6	
20	The Bullet flot plottee	0	

#### Table 1 – continued from previous page

#	Components	QTY Picture
29	Lighting System LEGO Pieces	
30	Light Controlled System LEGO Pieces	
31	Anti-theft Alarm System LEGO Pieces	

### Table 1 – continued from previous page

#	Components		Picture
32	Automatic Feeding System LEGO Pieces	1	
33	Rainwater Control System LEGO Pieces Temperature and Humidity System LEGO Pieces	1	

#### Table 1 – continued from previous page

			do page
#	Components	QTY	Picture
35	Soil Moisture/Water Level/Automatic Irrigation System LEGO Pieces	1	
36	Lego Board	2	

Table 1 – continued from previous page

### **KIDSBLOCK TUTORIAL**

### 4.1 1. Mainboard\_Introduction:

Refer to the linkhttp://kd2076-kidsbits-stem-electronic-building-block-programming.readthedocs.io/

### 4.2 2. KidsBlock Development Environment Configuration:

Please refer to the link to install and use the KidsBlock software https://kidsblocksite.readthedocs.io/en/latest/

**Note:** The control board used in this kit is the kidsIOT board. For importing the kidsIOT board, libraries and sample codes, please refer to the following content.

1. Click No device selected to enter the main page, and select the control board needed. In this project, we select the kidsIOT mainboard and click **Connect**, then it is connected. Click Go to Editor to return the code editor. Icon No device selected will change into Will change into USB-SERIAL CH340 (COM5). This means the kidsIOT mainboard and portsCOM)are connected.





			-							_			_					- 0	×
kids	block	•	Edit	🔛 kid	sIOT	Ų	USB-SE	RIAL CH	340 (CC	DM5)	Kid	ls		File	Ø	Download firmware	·∳ Tutorials	Upload 🔵	Ö
Sa Cod	ie 🦪 Cost	umes	() Sounds	3												📩 Upload			
Events	Events														Ç.	<pre>1 // generated by Ki 2 void setup() { 3 }</pre>	dsBlock		
Control	when Arduino	begin														4 5 void loop() { 6 }			-
Operato	Control															7			
/ariable:	wait 1 se	econds																	
My Blocks	repeat 10																		
Pins		<u>و</u>																	
Serial	forever																		
Sensor		3																	00
Data	ii 🕐	hen																	
Type																			
TEXT		hen												Q					
OLED	else																	Send	2
2.	If th	e ki	idsIOT	' m	aint	oar	d is	co	nne	cted	1,		but	ic	on	<b>N</b> onconnected	doesn't	change	in
ψı	JSB-SEF	RIAL	CH340	) (C(	M5	)	You	nee	d to	o cl	lick	to	co	nne	ct t	the COM port.	Click	Jnconne	cted
Then	vou will	find	a nage	non	un	show	ving	Com	hect	ed						•			

Then you will lind a page pop up, snowing Connected.



			_										-		×
kids	block	<b>-</b>	Edit	<b>kidslO</b> T	🖶 USB-SERIAL C	CH340 (COM5)		File	O			🔆 Tutorials			Ø
Coc	je 🚽 Costu	mes	() Sound	s					<b>±</b>	Upload					
	Events								, .	1 // 2 voi	<pre>generated by id setup() {</pre>	KidsBlock			
Events									2 -	3 } 4					
Control	when Arduino b	egin								5 voi 6 }	id loop() {				-
Operator	Control									7					
Variable:	wait 1 so	conde													
M		Contas													
Blocks	repeat 10														
Pins															
Serial															
	torever														
Sensor		٦													UU 面
Data	if 🛛 th	en													
Variable															
	i di														
TEXT															
OLED	else														
<b>~</b>														Send	4
1000											C	language a	rea cize e	wite	6
kid	sblock 🌐	)- E	a 🖀	kidsUno	🖶 USB-SERIAL CH	40 (COM5) 0	🕞	File D	0	Download	d firmware	<ul> <li>Tutorials</li> </ul>	Upload	C	
	Cada	Costure	an Tak	enne Uploa	ad teh code v	well edite	d Ne					± Upload			
	Events	Coolan		i i	n your comp	uter	- Lo	ad from your c	omputer			111	generated by	KidsBl	la
Eve	nts	-	_	Save	code to the	compute	r 🔶 🔽	re to your con	iputer		(D)p	load code	detup() {		
Con	trol when Ard	kino be	gin ad								crite	5 voic	1 loop() {		
Oper	rator bloc	k are	ea									7			
		-										C land	quage are	а	
Varia	iDies with 1	seco	ands												
Blo	y cks repeat	10													
Pir						6	de edi	ing area							
			-			Co	de ean	ing area						00	
Ser	rial torever													Đ	
Da	ta 🚺		2												
Varia	able											Serial	port moni	tor	
									2	zoom	in + 🖸				
Blo	xt cks								zo	o mo	ut + Q				
		the													
-			Se	ensors/mo	dules exten	sion libra	ry .				0		Send	2	

To disconnect the port, just click USB-SERIAL CH340 (COM5) and Disconnect.



NoteIf you want to update libraries of KidsBlock, click then Clear cache and restart.





esp32 Pas	sive buzzer
esp32 Pas	sive buzzer
Version 1.0.0	Author keyes
Help	
	Not loaded

esp32 Pas	sive buzzer
Version 1.0.0 Help	Author keyes
	Loaded

Click back to return to the code editor. Then you can view the passive buzzer in the blocks area.

OLED	Passiv	ve buzzei	r , ,					
WIFI	Å	Tone PIN	I# IO33 🔻	frequenty	NOTE_C3	duration	131	
Passive buzzer	<b>J</b>	Tone PIN	# IO33 <del>•</del>	play mu si	c Birthday 👻			
×.	Å	noTone	1033 🗸					
				s.				•
If you want	to delete	the passive	e buzzer, clic	k to	select the "esp	32 Passive b	uzzer"	. Then
Loaded wil	1 change	into <b>Not lo</b> a	aded. Then th	e passive buz	zzer is deleted.			

Paulan Contraction					
esp32 Pas esp32 Pass	esp32 Passive buzzer esp32 Passive buzzer				
Version 1.0.0	Author keyes				
Help					
Loaded					

esp32 Passive buzzer				
Version 1.0.0	Author keyes			
Help				
	Not loaded			

The way of deleting other sensors or modules is as same as the passive buzzer.

4. How to open SB3 type files

The first method Double-click SB3 type files to open them. For instance, open 1.1\_Blink, then we need to double-click it.

kide	sblock @	🕽 🗕 Edit 📲	kidslOT 🚽 USB-SERIAL CH340 (COM5)	1.1 📄 File	Download firmware	Upload 🔵 🧿
Co	de 🦪 Costumes	() Sounds			1 Upload	
Events	Events when Arduino begin		when Arduino begin		<pre>1 // generated by KidsBlock 2 #include <arduino.h> 3 4 void setup() { 5 pinMode(2, OUTPUT); </arduino.h></pre>	
Operato	Control		set pin IO2 ▼ mode outp	out 🗸	<pre>6    } 7 8    void loop() { 9        digitalWrite(2, HIGH); 10        delav(1 * 1000);</pre>	_
Variable: My Blocks	wait 1 second		set digital pin IO2 - out	high -	<pre>11 digitalWrite(2, LOW); 12 delay(1 * 1000); 13 } 14</pre>	
Pins			wait 1 seconds set digital pin IO2 - out			
Sensor	forever		wait 1 seconds	· · ·		
Data Variable	if then					
TEXT	if then				) -	
OLED	else					Send 🔑

The second method: Open Kidsblockclick **file** and **Load from your computer**then select the SB3 type file on the computer.for example 1.1\_Blink)

File	1 🖸		۵	
New				
Load from your computer 2				
Save to your computer				

🧕 Open		×
← → • ↑ <mark> </mark> •	KidsBlock_Codes	✓ O Search KidsBlock_Codes
Organize 🔻 New	folder	- 9= - ()
<ul> <li>Quick access</li> <li>Desktop</li> <li>Downloads</li> <li>Documents</li> <li>Pictures</li> <li>WPS</li> <li>This PC</li> <li>Network</li> </ul>	1.1_Blink.sb3         SB3 File         45.1 KB         1.3_Button.sb3         SB3 File         45.3 KB         1.5_Lighting-System.sb3         SB3 File         46.0 KB         2.2_Light-Control-System.sb3         SB3 File         45.6 KB         3.2_Ultrasonic-Sensor.sb3         SB3 File         45.1 KB	1.2_PWM.sb3         SB3 File         45.4 KB         1.4_Self-Locking-Button.sb3         SB3 File         45.8 KB         2.1_Photosensor.sb3         SB3 File         45.3 KB         3.1_Servo.sb3         SB3 File         45.1 KB         3.3_Intelligent-Feeding-System.sb3         SB3 File         45.7 KB
F	File name: 1.1_Blink.sb3	<ul> <li>✓ Custom Files ✓</li> </ul>
		2 Open Cancel
kidsblock @•	Edit 🏽 kidskOT 🕌 USB-SERUAL CH340 (COM5) 1.1 🕞 File	e 👩 🌐 Download firmware 🔅 Tutorials 🛛 Upload 🌑 🔅
😂 Code 🛹 Costumes	م) Sounds	typload
Events Control Variable: Pins Serial Events When Arduino begin Control Variable: Pins Serial Forever	when Arduino begin set pin 102 • mode output • forever set digital pin 102 • out high • wait 1 seconds set digital pin 102 • out low •	<pre>1 // generated by KidsBlock 2 #include <arduino.h> 3 4 void setup() { 5 pinMode(2, OUTPUT); 6 } 7 8 void loop() { 9 digitalWrite(2, HIGH); 10 delay(1 * 1000); 11 digitalWrite(2, LOW); 12 delay(1 * 1000); 13 } 14</arduino.h></pre>
Sensor Data Variable TEXT OLED		

### 4.3 3. Projects:

### 4.3.1 Project 01: Lighting System



#### 1. Description

As an introductory project for smart farms, lighting up LED is one of the most basic KidsBlock (based on Scratch) practical projects. It is designed to let beginners understand the hardware and software programming of kidsIOT board (based on ESP32) and master basic circuit and programming knowledge.

In this project, you will learn the basic connections and settings of the kidsIOT board in the KidsBlock graphical programming environment, as well as control the digital pin to output level to control the state of LEDs, LED breathing lights, and button control of LEDs, and you can also apply them in your home or lounge.

### 2. Components





**White LED:** It is a device that can convert electrical energy into visible light. When current passes through the LED, it emits light.

#### Parameters:

Working voltage: DC 3.3V-5V Working current: (Max) 1.5mA@5V Maximum power: 0.07W



**Button Module:** It can output a digital signal 0 or 1. When the button is pressed, it will output a low level 0, otherwise it will output a high level 1. It is widely used in doorbells, desk lamps, air conditioner remote controls and fire alarms.

#### **Parameters:**

Working voltage: DC 3.3V-5V Working current: (Max) 1.1mA@5V Maximum power: 5.5mW Signal type: digital signal (0 or 1)



#### 3. Assembly Steps

### Step 1:Components Needed



#### Step 2: Process

Process 1





Process 2






Process 4



Process 5



Process 7



Complete



# 4. Wiring Diagram

Module	kidsIOT Mainboard
White LED Module	No.2 portcontrol pin is io2
Button Module	No.4 portcontrol pin is io27

Connect the kidsIOT mainboard to your computer via USB cable.



## 5. State of LED



## (1). Knowledge

1To keep the light on, the electricity is needed. When we say that there is electricity, we mean that there is current flowing through an electrical appliance like a light. Current comes to our home from the power station via wires. And the generator of a power station is the power supply, which enables to provide voltage and current. The battery we usually use is also the power supply. Wires can be used to conduct electricity, which connect a path for the current to flow. This path is called a circuit. If we want to make a lamp emit light, both a power supply and a complete circuit are needed.



## (2). Programming Steps

## Step 1Description of the Building Blocks



This block indicates that when the kidsIOT board is started, the code will be run.



Set **input** or **output** to the specified pin. **input** means input mode,**output** means output mode. Select **input-pullup** can set the input mode for the pin and make it become high level.



Set **high** or **low** to the specified pin. Select **high** means to set high level for the pin. If there is voltage and current, the LED will be on. Select **low** means to set low level for the pin. If there is no voltage and current, the LED will be off.



This is a delay block. The number 1 can be changed to whatever number of seconds it is delayed.



It will do one thing forever.

## Step 2Write the Program

Open the KidsBlock(based on Scratch)software to select the kidsIOT board and port(COMx).

KidsBlo	ock Desktop 2.0.1									-	
kids	block	∰ <b>-</b> Edit	kidslOT	🕌 USB-SE	RIAL CH340 (COM5)	Kids	File O	Download firmwa	re 🔆 Tutorials	Uploa	d 🕘 🔅
🔚 Cod	le 🦪 Costume	es 🌒 🌒 Sounds	3				<b>1</b>	lpload			
Events	Events							L // generated b 2 void setup() { 3 }	y KidsBlock		
Control	when Arduino beg	in an an an						void loop() {			_
	Control										
My	wait 1 seco	nds									
Blocks Pins	repeat 10	9 8 8 9 8 8 9 8 8									
						_					
			wh	en Arc	luino be	gin					
Drag	the instruc	ction bloc	k	_		in	the <b>Events</b>	module to th	e script are	a.	
ĕ	KidsBlock D	esktop 2.0.1									
k	idsbl	ock	⊕-	Edit	📒 kids	ют	😽 Unconne	ected Ki	File	Ø	٥
	Code	🖌 Cos	tumes	() Sou	nds				2	, Upload	
	Eve	ents								1 / 2 #	/ genera
EV	ents								2	3 4 v	oid setu
Co	ntrol	en Arduino	begin			wher	Arduino	begin		5} 6	
										7 v 8 }	oid loop
Ope	erator co	ntrol									
Ope	erator Co	ntrol								9	
Ope Vari	iable:	ntrol it 1 s	econds							9	
Ope Vari	iable:	ntrol it 1 s	econds							9	
Ope Vari	iable:	ntrol it 1 s	econds			•				9	

Drag the instruction block

in the "**Pin**" module to the

script area. Since the white LED module is connected to the No. 2 interface on the mainboard (The control pin is io2) and it is in output mode, so change "**input**" to "**output**".

kidsb	lock	⊕ -	Edit	iii kidsl	л 🎽	Unconnected		KidsBlo	ock Pro	F •	ile	
Code	🖌 Costu	mes	() Soun	ds								
ents N	ly Blocks											, R'
	Make a Blo	ock										2
ntrol P	ins											
erato	set pin 🛛 IO2 🗸	mode	input 🗢		when	Arduino begi	n v					
iable:		102 -		-	set pin		mode	outout				
		102 •	out nigr		Set pin			output				
icks	set pwm pin 🗌	lO2 <b>▼</b> I	use chann	el Cl								
ins	set dac pin 10	025 👻 o	out 0									
	fo	rever	-									
ag the b	fo	rever			• •	the " <b>Contro</b> l	" module	to the s	cript are:	a.		
ag the b	lock	rever			¢ in	the " <b>Contro</b>	l" module	to the s	cript area	a.		
ag the b KidsBlock	lock Desktop 2.0.1	rever	Edit	Kidsl	J in	the "Control	l" module	to the s	cript area	a.	File	
ag the b KidsBlock	lock Desktop 2.0.1	rever 	Edit () Sour	<b>kids</b>	р т	the "Control	l" module	to the s KidsBl	cript are: ock Pro	a.	File	
ag the b KidsBlock	Iock Desktop 2.0.1	rever 	Edit (1) Sour	<b>kids</b>	) in	the "Control	l" module	to the s	cript area	a.	File	
ag the b KidsBlock	lock Desktop 2.0.1 Lock Costu Costu Control wait 1 se	rever umes conds	Edit (1) Sour	tids	) in DT	the "Control Unconnected	l" module	to the s KidsBl	cript area	a.	File	
ag the b KidsBlock	Iock Desktop 2.0.1 Inch Costu Control wait 1 se	rever mes conds	Edit (1) Sour	Ids	or k wher set p	the "Control Unconnected Arduino bea	l" module	to the s KidsBl	cript area	a.	File	

My Blocks



KidsBlock Desktop 2.0.1				
kidsblock 🛛 🕀 -	Edit 🎦 kidslOT	Vinconnected	KidsBlock Pro	File
🛫 Code 🥒 Costumes	() Sounds			
Control		when Arduino begin		
Operator repeat 10		set pin IO2  mode forever	output 🝷	
Variables		set digital pin IO2 ▼	out high -	
Pins		wait 1 seconds		
f then	igital pin IO2 ▼ 1 seconds	out high -	forever	J
and change " <b>High</b> " to "Low".		into		

🥉 KidsBl	ock Desktop 2.0.1			
kids	block (	<b>€</b> Edit	iikidslOT	KidsBlock Pro File
Co	de 🦪 Costumes	s 🌒 Sounds	s	
Events	Control			
Control	wait 1 second	ds		when Arduino begin
Operator	repeat 10			set pin IO2 ▼ mode output ▼
e Variable				forever
My	forever			set digital pin IO2 ▼ out high ▼
Blocks	3	2		wait 1 seconds a a a a a
Pins	if then			set digital pin IO2 ▼ out Iow ▼
Serial				wait 1 seconds
Sensor	if then			

## (3). Test Result

🔔 Upload

Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, the white LED will be on and off for 1s.

We can also realize breathing light effect, flowing water light effect and police light effect via LEDs.

Level	Function
high	LED lights up
low	LED lights off

#### 6. LED Breathing LED



#### (1). Knowledge

The IO port on the kidsIOT mainboard outputs digital signals, which can only output high level and low level. For example, in the lighting up LEDs project, the digital output of the ESP32 is used, which has only two levels: high (3.3V) and low (0V).

Assuming that the high level of the kidsIOT board is 3.3V and the low level is 0V, then if you want to output a voltage between 0 and 3.3V, you need to use PWM (Pulse Width Modulation). PWM can output different voltage values, like a progress bar, which is analog output.

	20%				
	and the second	40%			
	an an an ann		60%		
a a a the a	Sec. Sec.			80%	
					100%

PWM uses digital control to generate square waves with different duty cycles (a signal that constantly switches between high level and low level) to control the analog output.



PWM has three elements: frequency (Hz), period (s) and duty cycle (%)

- **PWM frequency** (f): It refers to the number of PWM cycles in one second.
- **PWM period** (**T**): Period=1/frequency (T=1/f, where 1 is 1 second), for example: the frequency is 50Hz, which means that one period is 20ms, then one second is 50 PWM cycles.
- **PWM duty cycle:** It refers to the ratio of high level time to the entire cycle time within a pulse cycle. For example: the cycle time is 10ms, the pulse width time is 8ms, then the low level time is 2ms, and the total duty cycle is 8/(8+2)= 80%.



PWM can change the effective output voltage by changing the duty cycle in one cycle under the appropriate signal frequency. Among the levels output by the IO port at the specified time, the more high levels, the greater the PWM

value and the brighter the LED.



## (2). Programming Steps

## Step 1Description of the Building Blocks

The following are "Variable" command blocks.

Declare	Global 🔻	variable Type	int 🔻	Name	item	Assign	ed to	0
		int €						
		unsig	ned int		÷ .			
		long						
		unsig	ned long					
		float	a					
		boole	an		1			
		byte			1			
		char			1.			$\odot$
		uneia	nod char		×			(Q)

This block is used to create "Variable". You can declare "global" or "local", or set the type, name and value of the variable, item is the variable name.



Get variable item.



Set the value of variable item.



Set the variable item mode to increase item by 1 or decrease item by 1 every time the loop is executed.



Set the string variable item.

set pwr	m pin IO2 🔻	use channel	CH0 (LT	0) 🔻	out	0
	100					
	100					
	101					
	✓ IO2					
	103					
	104					
	105					
	100					
	106					
	107					
	108					
	INA	-				



This block is used to set the PWM. You need to set the corresponding pin via the channel (a total of 16 channels  $(0\sim15)$ ) and the output value, so that the PWM value can be output.



This is a conditional loop control statement that exit the loop when the number of loops is met. For example: 10 means that the loop is executed 10 times. The number 10 can be changed to other numbers.

#### Step 2Write the Program



Drag the instruction block

in the Events module to the script area.



🕀 🗸 Edit 📑 kidslOT 😽 Unconnected File 📄 kidsblock 🛫 Code 🖌 Costumes () Sounds Variable Type My Blocks variable Type int 
Name item Assigned to 0 e Global 🝷 Pins variable (item Serial item variable by 0 ( Sensor set pin 102 🗢 mode input 💌 Change item variable mode ++ • Data Name (item) Assigned to 0 Declare Global 
variable Type int item String variable by 0 Variable Туре TEXT 10 repeat forever 3 Drag blocks and in the "Control" module to the script area.

in the "Variable Type" module to the script area.

KidsBlock Desktop 2.0.1



se	t pwm pin IO2	• use channel	CH0 (LT0) -	o
ag the instruction block				
"Pin" module to the script area	a.			
KidsBlock Desktop 2.0.1				
idsblock @• Edit 3	📑 kidslOT 🛛 😽 Unconnected	KidsBlock Project	File	
🕿 Code 🛛 🖌 Costumes 🔹 🌒 Sounds				
Pins				
Ay set pin IO2 ▼ mode input ▼				
	when Arduino begin			
set digital pin 102 ♥ out nign ♥	set pin IO2 👻 mode	e input 🗢		
erial set pwm pin 102 🔹 use channel				
nSor set dac pin 1025 • out 0		anable Type Int  Name	Assigned to	0
ata	forever			
read digital pin 102 -	repeat 10			
IADIE read analog pin 102 -				
Tread touch pin 102 -	set pwm pin IO2	use channel CH0 (LT)	0) ▼ out 0	
				$\odot$
ED set servo pin 102 • use channel	<u>ح</u>			Q
variable	item			
ag the block	in the "Varia	able Type" module to the s	cript area.	

in

KidsBlock Desktop 2.0.1			
kidsblock 🗣 Edit 🖀	kidslOT 😽 Unconnected	KidsBlock Project	File
😂 Code 🖌 Costumes 🜗 Sounds			
Variable Type			
Blocks Declare Global - variable Type int	when Arduino begin		
Pins variable item	set pin IO2 ▼ mode input ▼		
Serial Set item variable by 0	Declare Global  variable Type	int - Name item	Assigned to 0
Sensor Change item variable mode ++ •	forever		
Data Set item String variable by 0	repeat 10		
Variable TEXT Type	set pwm pin IO2 ▼ use char	nnel CH0 (LT0) 🔻	out variable item
TEXT string hello	<b>3</b> 1 1 1		
	Jan San San		a a a a a(⊕) a
Change	item variable mode	++ 🕶	
Drag the block		in the "	Variable Type" module

to the script area"++" means that each time the loop is executed, **item** will be increased by **1**.

KidsBlock Desktop 2.0.1			
kidsblack 🕀 Edit 🖀 ki	dslOT 🦎 Unconnected	KidsBlock Project	File
😂 Code 🛹 Costumes 🜗 Sounds			
Variable Type			· · · · ·
Blocks Declare Global  variable Type int	when Arduino begin		а а а <u>с</u>
Pins variable item	set pin IO2 ▼ mode input ▼		
Serial Set (item) variable by 0	Declare Global 🔻 variable Type	int 🔻 Name item	Assigned to 0
Sensor Change item variable mode ++ •	forever		
Data Set item String variable by 0	repeat 10		
Variable TEXT Type	set pwm pin IO2 ▼ use char	nnel CH0 (LT0) 🔻 d	out variable item
TEXT string hello	Change item variable mode	++ •	· · · ·
OLED 123	5		y y y u⊕
OLED	Jan San San San San San San San San San S		A A A A



Drag the block in the "**Control**" module to the script area and set the delay to **0.01** second, the repeat **10** to **255**, for the corresponding PWM code block outputs 0~255. In this way, the LED light will slowly turn from dark to bright.

KidsBlock Desktop 2.0.1											
kidsblock 🤅	🕽 🕶 Edit 📲	kidslOT	🙀 Unconnecte	d		KidsB	ock Project		File		
🛫 Code 🚽 Costumes	; () Sounds										
Control		-									6
wait 1 second	ds	when A	rduino begir	1							•
	· · ·	set pin	102 🔻 r	node inp	ut 🔻						
		Declare	Global 🖣	variable	е Туре	int 🔻	Name	item	Assigned	to 0	
		forever									
My Blocks		repea	at 255								
Pins				102					ut cuestials	itam	
Serial if then		sei	pwm pin	102 V U	se chan					le llem	
ensor		Ch	ange item	variable	mode	++ •	J .				G
Data if then		wai	it 0.01 s	seconds							0
else											
-											
	repeat 255		1.1								
	set pwm p	in IO2	- use	channel	СНО	(LT0)	- ol	it va	riable	item	
										-	
	Change	item v	ariable m	ode ++	•						
	walt 0.0	1 seco	onds								
		و									
Copy the code block	D will slowly t	urn from l	bright to da	rk.							cha

🍯 KidsBl	ock Desktop 2.0.1												
kids	block	🕀 🔻 Edit	📳 ki	dslOT	😽 Unconnec	ted		KidsE	lock Projec		File		
Co	de 🦪 Costume	es 🌒 Sour	nds										
Events	Control			-									
Control	wait 1 seco	nds		when	Arduino beg	in							
Oporato	repeat 10			set pi	n IO2 🔻	mode	input 🔻	1					
Variable		۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲. ۲		Decla	Global	<ul> <li>varia</li> </ul>	able Type	int 🔻	Name	item	Assign	ed to 0	
My	forever			forev	er								
Blocks		مراجع الم		гер	eat 255								
Pins				s	et pwm pin	102 -	use cha	nnel	CHO (LTO	) 🔻 o	ut varia	able iter	m .
Serial	if then					n varia	ble mode						н н.
Sensor						Valla	ible mode						
Data	if ther			V	vait 0.01	seconds	а а						
Variable Type	else			гер	eat 255	<u> </u>				_			
TEXT	wait until			s	et pwm pin	IO2 🔻	use cha	nnel (	CHO (LTO	) 🔻 o	ut varia	able iter	m
OLED	repeat until			0	Change iter	n varia	ible mode	•					
WIFI		<b>9</b> a - a		v	vait 0.01	seconds	,						
	Operators					•							
						<b>ب</b>							. Q

Complete Program

Enterint ()       ()	💧 Kidsl	Block Desktop 2.0.1											-	
<pre>     Costumes</pre>	kid	lsblock	🌐 🕶 Edit	kidslOT	🙀 Unconnected				File	0		🔆 Tutorials		
Control <ul> <li>Events</li> <li>Events</li> <li>Vid seconds</li> <li>int item = 0;</li> <li>int item = 0;</li> <li>void setup() {                 <ul> <li>set pln 102 • mode input •</li> <li>Declare Global • variable Type int • Name (item Assigned to 0)</li> <li>gladwitte(0, 490, 8);</li> <li>ledcattach(0, 200, 100, 100, 100, 100, 100, 100, 10</li></ul></li></ul>	<b>2</b> 0	Code 🦪 Costum	ies 🌒 🌒 Sound	ts						<b>1</b> . U	pload			
Variable TEXT OLED WIFI Operators	Events Contro Operate Wanable My Blocks Serial Pins Serial Data Variablé Uata Data	S Control wait 1 second forever foreve	Image:	whe set p Dect	n Arduino begin in IO2 - mode are Global - v ver peat 255 set pwm pin IO2 Change (item) va wait 0.01 seco peat 255 set pwm pin IO2 Change (item) va wait 0.01 seco 	<ul> <li>input </li> <li>ariable Type</li> <li>use chan</li> <li>ariable mode</li> <li>ariable mode</li> <li>ariable mode</li> </ul>	Int Variation Nariation Nariatio Nariation Nariati	ne (item) 	Assigned to the second se	1 2 3 4 4 5 6 6 7 7 8 9 9 12 13 14 15 16 17 7 18 19 9 20 22 23 22 22 22 22 22 22 22 22 22 22 22	<pre>// generated by K: #include <arduino int item = 0; void setup() { ledcSetup(0, 49) ledcAttachPin(2, INPU } void loop() { for (int index ledcWrite(0, : item+; delay(0.01 * ; } for (int index ledcWrite(0, : item-; delay(0.01 * ; } }</arduino </pre>	<pre>idsBlock h&gt;  0, 8); 0); 0; 0; index &lt; 255 item); 10000; item); 10000; </pre>	; index++) { ; index++) {	

#### (3). Test Result



Click to upload the above complete code to the kidsIOT motherboard. After powering up via the USB cable, the LED will gradually brighten and then dim, like breathing.

#### 7. Read the value of the button module



## (1). Knowledge

The principle of the button module is based on the switch circuit.

When the button is pressed, the switch closes, allowing current to pass through the button to GND, then the digital input pin of the kidsIOT motherboard detects a low level signal.

When not pressed, the switch is in the off state, the pin is pulled high by the pull-up resistor, and the digital input pin detects a high-level signal.

#### (2). Programming Steps

#### Step 1Description of the Building Blocks

serial	0 🔻	begi	n bau	Idrate	115200 👻
					4800
					9600 19200
					38400 . 57600
					76800
				· 🗸	115200

The block is used to set serial baud rate(generally, the baud rate 9600 is taken as an example)

se	erial	0 🔻	print	Н	ello Ki	dsBlo	ock warp -	
	-					1	✓ warp	
							no-warp	
							HEX	

This block is used to set print mode for the serial port. **warp** means line feed printing, **no-warp** means no line feed printing, **HEX** means hexadecimal printing.



It is used to read the digital signal value of the specified pin0 or 1).

#### Step 2Write the Program



Drag the instruction block



module to the script area and take the baud rate 15200 as an example.

🥉 KidsBl	ock Desktop 2.0.1		
kids	black 🕀 🕈 Edit 📑 kidslOT 🦎 Und	onnected KidsBlo	File
Co	de 🥒 Costumes 🜗 Sounds		
	Serial		·
My Blocks	serial 0 - begin baudrate 115200		
Pins	serial 0 - print Hello KidsBlock when Arduino	begin	
Serial Sensor	serial     0 → available data length       serial     0 → read a byte	begin baudrate 115	200 🗸
Drag the	set pin IO2 - mode input	in the " <b>Pin</b> " modu	ale to the script area.

Since the button module is connected to No. 4 port ( the control pin is io27), so change pin IO2 to IO27.

🥉 KidsBlock Desktop 2.0.1

kidsblack	
Code Costumes 📣 Sounds	
Pins Pins Pins Set digital pin 102 • out high •	
Serial Set pwm pin 102 • use channel Ct Sensor Set dac pin 1025 • out 0	
Declare Global  variable Type int  Name item Assig	gned

in the "Variable Type" module to the script area, then change item to "Button".

kidsblock 🏾 🌐	• Edit	kidslOT		Riusi	юск ртојест		File			
Code Costumes	() Sour	nds								
My Variable Type										
ocks Declare Global -	variable Ty	pe int <del>-</del> Name	item Assigned to 0							J
Pins	n n									
erial variable item		when	Arduino begin							
Set item variable	by 0			445000 -						
NSOF Change item varia	ble mode	++ •		115200 •						
ata	ariable by	set pir	n IO27 - mode inp	ut 🝷		-	-	-	-	-
iable		Decla	re Global 👻 variable	Type int 🔻	Name	Button	Ass	igned	to	0
pe TEXT										
			<b>*</b>							
rag the block			in the "Con	<b>trol</b> " modul	e to the s	cript ar	ea.			
rag the block			in the "Con	<b>trol</b> " modul	e to the s	cript ar	ea.			
rag the block KidsBlock Desktop 2.0.1	• Edit	tidslOT	in the "Con	<b>trol</b> " modul KidsE	e to the s lock Project	cript ar	ea.			
rag the block KidsBlock Desktop 2.0.1	• Edit	<b>kidslOT</b>	in the "Con	<b>trol</b> " modul KidsE	e to the s lock Project	cript ar	rea.			
rag the block KidsBlock Desktop 2.0.1 MSD DOCK	<ul> <li>Edit</li> <li>(1) Sour</li> </ul>	<b>kidsiOT</b>	in the "Con	<b>trol</b> " modul KidsE	e to the s lock Project	cript ar	ea.			
rag the block KidsBlock Desktop 2.0.1 Code Costumes Ents Control Wait 1 seconds	<ul> <li>Edit</li> <li>(1) Sour</li> </ul>	ids	in the "Con	<b>trol</b> " modul KidsE	e to the s lock Project	cript ar	ea.		-	
rag the block KidsBlock Desktop 2.0.1 Alsolock (***) Code Costumes ents Control wait 1 seconds	<ul> <li>Edit</li> <li>(1) Sour</li> </ul>	tidsIOT	in the "Con	<b>trol</b> " modul KidsE	e to the s	cript ar	ea.		-	- -
ag the block KidsBlock Desktop 2.0.1 Code Costumes Code Costumes Main Control Wait 1 seconds ratol repeat 10	► Edit	ids when serial	in the "Con Unconnected Arduino begin begin baudrate	trol" modul KidsE	e to the s	cript ar	ea.			
ag the block GidsBlock Desktop 2.0.1 Code Costumes Code Costumes Market Control Wait 1 seconds rator repeat 10	• Edit	Ids kidslot when serial set pi	in the "Con in the "Con Unconnected Arduino begin 0 • begin baudrate in 1027 • mode inp	trol" modul Kidst 115200 -	e to the s	cript ar	ea.			
ag the block KidsBlock Desktop 2.0.1 Code Costumes Ints Control wait 1 seconds rator repeat 10 	<ul> <li>Edit</li> <li>(1) Source</li> </ul>	tidsIOT kidsIOT when serial set pi	in the "Con in the "Con	trol" modul Kidst 115200 -	e to the s	cript ar	ea.			
rag the block KidsBlock Desktop 2.0.1 Code Costumes ents Control wait 1 seconds rato repeat 10 forever	► Edit (1) Sour	tids tids when serial set pi Decla	in the "Con in th	trol" modul Kidst 115200 - Dut - Type int -	e to the s lock Project	Cript ar	rea.	signed		· · · ·
ag the block KidsBlock Desktop 2.0.1 A Code Costumes Code Costumes Mail Control Wait 1 seconds rato repeat 10 forever	► Edit	KidslOT Ids when serial set pi Decla	in the "Con in the "Con Unconnected Arduino begin I 0 • begin baudrate in IO27 • mode inp are Global • variable er	trol" modul KidsE 115200 - Dut - Type int -	e to the s lock Project	Cript ar	ea. File Ass	signed	- - - - - -	- - - - - - - - - - - - - - - - - - -
rag the block KidsBlock Desktop 2.0.1 Code Costumes ents Control wait 1 seconds ratol repeat 10 forever state forever rial if then	► Edit	kidsiOT Ids when serial set pi Decla	in the "Con in the "Con Unconnected Arduino begin 0 • begin baudrate in 1027 • mode inp are Global • variable er	trol" modul Kidst 115200 - Dut - Type int -	e to the s lock Project	Buttor	ea.	signed	- - - - - - - - - - - - - - - - - - -	- - - - - - - - - - - - - - - - - - -
rag the block KidsBlock Desktop 2.0.1 Code Costumes Control wait 1 seconds renats repeat 10 forever pins erial if then	<ul> <li>Edit</li> <li>(1) Sour</li> </ul>	tidsIOT tids when serial set pi Decla foreve	in the "Con in the "Con Unconnected Arduino begin 0 • begin baudrate in 1027 • mode inp are Global • variable er	trol" modul Kidst 115200 - Dut - Type int -	e to the s lock Project	Buttor	ea.	signed	l to	
rag the block KidsBlock Desktop 2.0.1 A code Costumes Prato Control Wait 1 seconds Prato forever if then	Edit	item	in the "Con in the "Con Unconnected Arduino begin I 0 • begin baudrate in IO27 • mode inp are Global • variable er	trol" modul Kidst 115200 - Dut - Type int -	e to the s lock Project	Buttor	ea.	signed		
rag the block KidsBlock Desktop 2.0.1	Edit	kidsioT Ids when serial set pi Decla forevu	in the "Con in the "Con Unconnected Arduino begin I 0 • begin baudrate in IO27 • mode inp are Global • variable er yariable by 0	trol" modul Kids 115200 - Dut - Type int -	e to the s	Buttor	ea.	signed	- - - - - - - - - - - - - - - - - - -	

area, then change item to "Button".

Kidsblock               Edit	
Code Costumes I Sounds	
Variable Type	
Blocks Declare Global - variable Type int when Arduino begin	
Pins variable (item) Social 0 ▼ begin baudrate 115200 ▼	
Sensor S	
Change item variable mode ++  Declare Global  variable Type int  Name Button Assigned to	0
Variable	
Set Button variable by 0	
• string hello	
Drag the block in the " <b>Pin</b> " module to the script area, the pin IO2 to IO27.	n change
Pins Events	
when Arduino begin	
Set pin       102 • mode input •       when Arduino begin         Set digital pin       102 • out high •       serial       0 • begin baudrate	
Set pin IO2 • mode input •   When Arduino begin Set digital pin IO2 • out high • Set pin IO2 • use channel Ct Set pin IO2 • mode input • <pinput p="" •<=""> Input • Input</pinput>	
Set pin       102 • mode       input ▼         Set digital pin       102 • out       high ▼         Set digital pin       102 • out       high ▼         Set pin       102 • out       high ▼         Set pin       102 • out       begin         Batteria       0 ▼       begin         Declare       Global ▼       variable         My       Name       Button	
Set pin 102 • mode   Control set digital pin   Operator set digital pin   Operator set pin   Set pin 102 • use channel   Control set pin   Set pin 102 • mode   Image: Set dac pin 102 • out   My Blocks   Fread digital pin 02 •    When Arduino begin Set pin <p< td=""><td></td></p<>	
Image: Set pin 102 → mode input ♥       when Arduino begin         Set digital pin 102 • out high ♥       serial 0 ♥ begin baudrate 115200 ♥         Operator       set pwm pin 102 • use channel Ct         Image: Set dac pin 1025 • out 0       set pin 1027 ♥ mode input ♥         My Blocks       read digital pin 102 •         Image: Tread digital pin 102 •       Image: Tread digital pin 102 •         Image: Tread digital pin 102 •       Set Button variable by read digital pin 102 •	
Set pin 102 • mode input •   Set digital pin 102 • out high •   Operator set digital pin   Operator set pin   Set pin 102 • use channel of   My Blocks   Fread digital pin 102 •   Fread digital pin 102 •   Serial 0 •   Serial <	

script area.

KidsBlock Desktop 2.0.1				
kidsblock - Edit	kidslOT	🙀 Unconnected	KidsBlock Pr	File
🛫 Code 🥒 Costumes 🔌	Sounds			
Operator serial 0 - begin baudrate	115200 when Ar	rduino begin		
Variable:	dsBlock set pin	IO27 • mode input •		· · · ·
Blocks serial 0 - available data	length Declare	Global 👻 variable Type	int - Name Button	Assigned to 0
Pins serial 0 • read a byte Sensor	Set	Button String variable by	read digital pin 1027	
Serial read hall sensor	serial	0 • print Hello KidsBl	ock warp -	
Drag the block "Button". KidsBlock Desktop 2.0.1 Kidsblock @- Edit ::	in the kidslOT 💥 Unconne	e "Variable Type" mod	lule to the script area, KidsBlock Project	then change item to
Code Costumes 📢 Sounds				
Variable Type Operator Declare Global - variable Type int Variable	when Arduino be	egin		
My Blocks Set (item variable by 0	serial 0 🔻 be	egin baudrate 115200 ·		
Pins Change item variable mode ++ •	set pin IO27 - Declare Globa	<pre>mode input ▼</pre>	Name Button A	Assigned to 0
Serial Set (item String variable by 0) Sensor TEXT	forever			
Data char a string hello	Set Button	String variable by	ead digital pin 1027 -	
Variable Type		print variable bullon	waip	

KidsBlock Desktop 2.0.1				
kidsblock - Edit	🖀 kidslOT 🛛 🙀 Unconnected	KidsBlock Project	File	
🛫 Code 🦪 Costumes 🔹 🌒 Sounds				
Control				
wait 1 seconds	when Arduino begin			
operator repeat 10	serial 0 🔹 begin baudrate 1	15200 👻		
ariable	set pin IO27 🕶 mode input			
My Blocks	Declare Global 🔻 variable Ty	pe int 🔻 Name 🖪	utton Assigne	d to 0
Pins	forever			
Serial if then	Set Button String variable	by <b>read digital pin</b>	1027 🗸	
Sensor	serial 0 - print variable	Button warp 🔻		
if then				

# Complete Program

KidsBlock Desktop 2.0.1				
kidsblock - Edit 🖀	kidslOT 🦎 Unconnected	KidsBlock Project	File	🖸 🔃 Download firmware 🔅 Tutor
😂 Code 🚽 Costumes 📢 Sounds				1 Upload
Events Control	when Arduino begin			<pre>1 // generated by KidsBlock 2 #include <arduino.h> 3 4 int Button = 0; 5</arduino.h></pre>
Operator Variable:	serial 0 ♥ begin baudrate set pin 1027 ♥ mode inp	115200 -		<pre>6 7 void setup() { 8 Serial.begin(115200); 9 pinMode(27, INPUT); 10 } 11</pre>
forever Blocks	Declare Global  variable	Type int ▼ Name But	ton Assigned to 0	12 void loop() { 13 Button=digitalRead(27):
Pins Serial if then	forever Set Button String variab	le by <b>read digital pin</b>	027 -	
Sensor if then	serial 0 - print variable	e Button warp 🔻		1
else				

## (3). Test Result



to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click  $\checkmark$  in the serial monitor and set the baud rate to 15200. When the button is not pressed, the serial monitor prints the digital signal 1, when pressed, the digital signal 0 is printed.

1	00
1	<b></b>
1	
1	
1	Buadrate 115200 V 2
0	
0	End of line LF & CR •
0	Hex form
0	Auto accellent
ō	
	Send 🔀

## 8. Self-locking button function



## (1). Knowledge

Self-locking button: It locks when pressed and will not automatically pop up. It will pop up only when pressed again, which is very similar to a switch. The switch can be turned on and off using self-locking buttons. If it is controlled by the kidsIOT mainboard, this operation can also be achieved via software.

## (2). Programming Steps

#### Step 1Description of the Building Blocks



It is a conditional statement code executing if-then function: If the logical judgment statement in **the statement below then** is executed.



It is a conditional statement: If the logical judgment statement in continuously. If not, the loop will be terminated.

is satisfied, the loop will be executed



It is a conditional statement code executing if-then-else function: If the logical judgment statement in is satisfied, the code statement below **then** is executed, otherwise, the code below **else** is executed.

# Step 2Write the Program

First define a Value to get the button status, and then define a Button. At the same time, select the serial baud rate to 15200 and the control button pin IO27 to "**input**" mode.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block 🕀 - Edit 🖀 ki	dslOT 🦎 Unconnected	KidsBlock Project	File	
Coc	de 🖌 Costumes 🌒 Sounds				
	Variable Type				
Operator	Declare Global ▼ variable Type int	when Arduino begin			•••••• •••••
Variable	variable item	serial 0 - begin baudrate 115	200 🗸		
My Blocks	Set item variable by 0	set pin 1027  mode input			
Pins	Change item variable mode ++ •	Declare Global - variable Type	int 🔻 Name Va	lue Assigned to	0
Serial	Set item String variable by 0	Declare Global  variable Type	int ▼ Name Bu	Itton Assigned to	0
Sensor	TEXT				

Assign the read button value to "Value".

🥉 KidsBl	ock Desktop 2.0.1											
kinlsblock - Edit 🖀 kidstOT 💥 Unconnected KidsBlock F				(Project	Project 📄 File							
Co	Code Costumes 🜗 Sounds											
Operator	Variable Type											
Variable	Declare Global  variable Type	int when Arduir	no begin									
	variable item	serial 0 🗸	begin baud	rate 1152	200 🗸							
My Blocks	Set item variable by 0	set pin IO	27 <b>v</b> mode	input 💌								
Pins	Change item variable mode ++	• Declare G	Blobal 💌 vari	able Type	int 📼	Name	Value	Assian	ed to			
Serial	Set (item) String variable by 0			abic Type		Name	value	Assign				
Sensor	техт	Declare G	Global 👻 vari	able Type	int 🔻	Name	Button	Assig	ned to	0		
	char a	forever										
	string hello	if	ead digital pin	1027 🗢	b then							
Variable Type	123	Set	Value variab	le by 1								
TEXT	OLED a a	else										
OLED	init oled width 128 height 64 l	2C Set	Value variab	le by 0						۲ 🕞 ۲		
WIFI	draw line x0: 0 y0: 0 , x1:	32										
=	draw rect x: 0 y: 0 width	32	٦							) (=)		


Determine whether the button is pressed. When pressed, change the value of "Button" and print it.

Description

if variable Value = 0 then
wait 0.01 seconds
if variable Value = 0 then
if variable Button = 1 then
Set Button variable by 0
else
Set Button variable by 1
repeat until read digital pin IO27 -
. <b>ع</b>
serial 0 - print variable Button warp -
و

The delay of 0.01 seconds is the button anti-shake function. After detecting that the button is closed, a delay program is executed, with a delay of 5ms 10ms (depending on the mechanical characteristics). After the jitter disappears, the button status is detected again. If the closed state level is still maintained, then a button is pressed.

When it is released, a delay of 5ms 10ms is also required. Only after the jitter disappears can the button processing program be transferred. When the button is pressed once, the button becomes 1, when pressed again, it becomes 0.

Complete Program



#### (3). Test Result



to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click  $\checkmark$  in the serial monitor and set the baud rate to 15200. When the button is pressed, the serial monitor prints the number 1, and when pressed again, the monitor prints the number 0, so as to achieve the self-locking function of buttons.

0 0 0 0 0 0	00 <b>逾</b>
0 0 0 0 0 0	Buadrate 115200 V 2 End of line LF & CR V
0 0 0 0	Hex form Auto scroll 🖉
	Send

9. Lighting control system



# (1). Programming Steps

#### **Step 1Flow Chart**



## Step 2Write the Program

Based on the code of the self-locking button above, add the relevant blocks for turning the LED on and off.



Complete Program



#### (2). Test Result



Click to upload the above complete code to the kidsIOT motherboard, then power up via the USB cable. When the button is pressed for the first time, the LED is turned on. When pressed for the second time, the LED is turned on again..., which is consistent with the LED switch in real life.



#### **10. Common Problems**

#### Q1: There are errors in kidsIOT board programming program

A: Please check whether the board type is correct.

Please check whether the USB port number is selected correctly.

#### Q2: The LED does not light up after burning the code

A: Please confirm whether the pins in the code are consistent with the actual wiring. If there is an error, please wire correctly according to the pins in the code.

#### Q3: Are the buttons insensitive? Sometimes it can be detected, sometimes not?

A: Modify the button delay time and set it to an appropriate delay.



# 4.3.2 Project 02: Light Controlled System



#### 1. Description

This light controlled system is composed of a photoresisitor, a LED and a kidsIOT mainboard, which can realize intelligent lighting control, saving energy and improving usage efficiency.

It empowers to automatically detect day and night as well as light intensity, making the entire system more intelligent and energy-saving.

When the ambient brightness is lower than the set value, the photoresisitor will detect a signal and automatically turn on the LED; when higher than the set value, it will send another signal to turn off the LED.

#### 2. Components



# 3. Assembly Steps

## Step 1Components Needed



## Step 2Process



 $\times 2$ 















Complete



(Note: Do not dismantle it, it will still be used in project 10.)

# 4. Wiring Diagram

Module	kidsIOT Mainboard
White LED Module	No.2 portcontrol pin is io2
Photoresistor	No.4 portcontrol pin is io39

Connect the kidsIOT mainboard to your computer via USB cable.



5. Read the value of the photoresistor



## (1). Programming Steps

## Step 1Description of the Building Blocks



It is used to read the analog signal value of the specified pin.

#### Step 2Write the Program

Set the baud rate to 15200.

KidsBlock Desktop 2.0.1														
kids	sble	ock	⊕-	Edit		kidslOT	۲	Uncor	nnected		Kidsl	Blo	File	9
Co	de	🖌 Cos	tumes	() Soun	ds									
	Ser	ial												
My Blocks	ser	ial 0 🗸	begin ba	udrate 11	5200									ر <b>تا :</b>
Pins	ser	ial 0 🗸	print H	ello KidsBl	ock	wher	n Ardu	ino b	egin					
Serial	sei	rial 0 🗸	availabl	e data leng	th					boude	eta (	4450	00 -	
Sensor	sei	rial 0 🗸	read a b	oyte		sena			egin i	Daudi		1152	00-	

Set the pin IO39 connected to the photoresistor ( control pin io39) to the "input" mode.



Define a "Photosensor" variable to store the value of the photoresistor.

KidsBlock Desktop 2.0.1										
kidsblock 🕀 Edit	kidslOT 🙀 Unconnected	KidsBlock Project	File							
🛫 Code 🛹 Costumes 🌒 Sounds										
My Variable Type										
Blocks Declare Global - variable Type	int - Name item Assigned to 0		· · · · · ·							
Pins variable item										
Serial Set (item) variable by 0	when Arduino begin									
Sensor Change item variable mode ++	serial 0 🔻 begin baudrate 115200 🕚	-								
Data Set (item) String variable by 0	set pin IO39 ▼ mode input ▼									
Variable Type TEXT	Declare Global 🕶 variable Type int	Name Photoser	asor Assigned to 0							

Store the read value of the photoresistor in the "Photosensor" variable.

KidsBlock Desktop 2.0.1												
kidst	olock @•	Edit	<b>kidslO</b> T	Vinconnected					<b>(</b> )	ile		
Code	Costumes	() Sounds										
Events E	Events											
Control	when Arduino begin		when /	Arduino begin								
Operator			serial	0 🔻 begin b	oaudrate 115	200 🔻						
Variable	Control		set pin	IO39 👻 m	ode input 👻							
Mu	wait 1 seconds		Declar	e Global 🔻	variable Type	int 🔫	Name	Photos	sensor	Assigne	ed to	0
Blocks	repeat 10		foreve	 -			)					
Pins												
Serial	forever		Set	Photosensor	variable by	read a	nalog pir	1 IO39				
Sensor	· · · · · · · · · · · · · · · · · · ·		seria	al 0 🔻 print	variable P	hotosens	sor	∕arp ▼				œ
Data			wait	0.5 second	ds							Q
<u></u> *	if then			ح ک								=

Complete Program

KidsBlock Desktop 2.0.1				
kidsblock @• Edit #	kidslOT 😽 Unconnected	File	o (	🗘 Download firmware 🔅 Tutor
🚝 Code 🚽 Costumes 🜗 Sounds			🔔 Uplo	ad
Events	<u> </u>		. 1	<pre>// generated by KidsBlock #include <arduino.h></arduino.h></pre>
Control when Arduino begin	when Arduino begin		. 4	int Photosensor = 0;
Operator	serial 0   begin baudrate 115200		6 7 8	<pre>void setup() {   Serial.begin(115200);</pre>
Variables vait 1 seconds	set pin IO39 ▼ mode input ▼		. 9 10 11	<pre>pinMode(39, INPUT); }</pre>
My	Declare Global 🔻 variable Type int 💌 Nam	e Photosensor Assigned to	) 12	<pre>void loop() {</pre>
Blocks repeat 10	forever			
Pins	Set Photosensor variable by read analog	pin IO39 🗸		
Serial forever	serial 0 - print variable Photosensor	warp 🗸 🔹 👘 👘		
Sensor	wait 0.5 seconds		2	
Data if then	e de la companya de l			

### (2). Test Result

Click Upload

to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click in the serial monitor and set the baud rate to 15200. Then the serial monitor will print the value read by the photoresistor. When the light detected by the photoresistor is brighter, the monitor prints a larger value, otherwise, the monitor prints a smaller value.

1623	00
2019	क्ति
2166	
2707	
2749	
2817	
2884	Buadrate 115200 V
2928	Buddiate Hozar
2975	
2997	End of line LF & CR V
3031	
3045	Hey form
3071	
3110	
3149	Auto scroll 🗹
3162	
	Send 💋

# 6. Light Controlled System



(1). Programming Steps

#### **Step 1Flow Chart**



#### Step 2Write the Program

Delete the "Wait 0.5 seconds" block in the complete program above, and then drag "**Set pin IO2 mode input**" block from the "**Pin**" module to the script area. Change "**input**" to "**output**".

KidsBlock Desktop 2.0.1		
kidsblock @• Edit 4	kidslOT 🙀 Unconnected	File
😂 Code 🔮 Costumes 🌒 Sounds		
Events	when Arduino begin	
Control when Arduino begin	serial 0  → begin baudrate 115200  →	
Variable: wait 1 seconds	set pin IO39 ▼ mode input ▼	
My Blocks	set pin     IO2 ▼ mode     output ▼       Declare     Global ▼ variable Type     int ▼	Name Photosensor Assigned to 0
Pins J	forever	
Sensor	Set Photosensor variable by read a	nalog pin 1039 •
Data if then	serial 0 - print variable Photosens	
		=

Determine the value read by the photoresistor. When the value is less than or equal to 700, the LED will turn on, otherwise, the LED will turn off.

KidsBlock Desktop 2.0.1		
kiusblock 🕀- Edit 🖀	idslOT 🙀 Unconnected	File
🔚 Code 🚽 Costumes 🌒 Sounds		
Events		
Control when Arduino begin	serial 0  begin baudrate 115200	
Control	set pin IO39 ▼ mode input ▼ A	
Variable: wait 1 seconds	set pin IO2 ▼ mode output ▼	
My Blocks	Declare Global ▼ variable Type int ▼ Name	Photosensor Assigned to 0
Pins J	forever	<u> </u>
Sensor	Set Photosensor variable by read analog pin	1039 -
Data	serial 0 - print variable Photosensor w	arp 🗸 🦂 kara kara kara kara kara kara kara kar
Variable	if not variable Photosensor > 700	then
TEXT if then	set digital pin IO2 ▼ out high ▼	
OLED else		
WIFI	set digital pin 102 • out tow •	
wait until	•	

Complete Program

🥉 KidsBl	ock Desktop 2.0.1									
kids	iblock @- Edit	📑 kidslOT	🙀 Unconnected		Ē	File		O		e 🔆 Tutorial
Co	de 🚽 Costumes 📢 Si	ounds						E	L Upload	
Events	Events	wher	Arduino begin						1 // generated by 2 #include <ardu 3</ardu 	y KidsBlock ino.h>
Control	when Arduino begin	seria	begin baudrate 1152	· · · ·					4 int Photosenso 5 6	r = 0;
Operator	Control	set p	in IO39 ▼ mode input ▼						<pre>7 void setup() { 8 Serial.begin 9 pinMode(39, 1 10 pinMode(2, 0)</pre>	(115200); INPUT); UTPUT):
Variable:	wait 1 seconds	set p	in IO2 - mode output -	n n n					11 } 12 13 void loop() { 14 Photosenson-	analogRead(30).
Pins	repeat 10	Deck	are Global 🔻 variable Type	int <ul> <li>Name</li> </ul>	Photosenso	Assign	ed to		15 Serial.print 16 if (!(Photos 17 digitalWri 18 }	<pre>ln(Photosensor); ensor &gt; 700)) { te(2, HIGH);</pre>
Serial	forever	Se	t Photosensor variable by	read analog pir	n IO39 🗸				19 else{ 20 digitalWri 21 } 22 }	te(2, LOW);
Sensor Data		se	rial 0 - print variable Pho	otosensor	varp 🔻				23	
Variable Type		if	not variable Photose	ensor > 700	then					
TEXT	if then	els	set digital pin IO2 ▼ out hig	gh 🔻						
OLED WIFI	else		set digital pin IO2  out Iov	N 🔹 k				a)		
<b>±</b>	wait until							Ð		

#### (2). Test Result

# \_\_\_\_\_

🔔 Upload

Click to upload the above complete code to the kidsIOT motherboard, then power up via the USB cable. When the light is strong during the day and the value of the photoresistor is greater than 700, the LED will turn off. After dark, when the value is less than or equal to 700, the LED will automatically turn on.



#### 7. Common Problems

#### Q1: The value detected by the photoresistor cannot be 0?

A: Because in actual operation, even if all the lights are turned off and the room is very dark, the value is only close to 0.

# Q2: After burning the sample code, why can't the LED light up even when the lights are turned off in the room?

A: You can set the value read by the photoresistor to be larger. The value in the sample code is 700, which can be adjusted to 1000 or even larger.

## 4.3.3 Project 03: Automatic Feeding System

#### 1. Description

The automatic feeding system is composed of a kidsIOT main board, an ultrasonic sensor and a servo. The ultrasonic sensor is used to detect the distance of pets in the feeding area. When the pet approaches the food bowl, the sensor detects that the distance is getting closer. After triggering the signal, it controls the servo to open the feed box and automatically feed the animals.

#### 2. Components





# 3. Assembly Steps

# Step 1Components Needed



## Step 2Process






























Process 16Initialize the servo angle



Wiring of servo

First write the following code in KidsBlock software and upload the code to the kidsIOT mainboard, then the servo will rotate  $190^{\circ}$ . (Note: If the servo can not rotate, you can press the RESET button on the kidsIOT board.)

when Arduino begin			
set pin IO33 👻 moo	de output 👻		
set pwm pin IO33 🗢	use channel	CH2 (LT1) 🔻	out 90
wait 0.5 seconds			
set pwm pin IO33 🗢	use channel	CH2 (LT1) 🔻	out 190
wait 0.5 seconds			





```
Process 20
```









Complete



# 4. Wiring Diagram

Module	kidsIOT Mainboard
Ultrasonic Adapter Board	No.9 portTrig-io18Echo-io19
Servo	G/V/io33 portBrown $\rightarrow$ GRed $\rightarrow$ VOrange $\rightarrow$ io33

Ultrasonic Sensor	Ultrasonic Adapter Board
Vcc	VCC
Trig	Trig
Echo	Echo
Gnd	GND

Connect the kidsIOT mainboard to your computer via USB cable.



5. Servo rotation



# (1). Programming Steps

## Step 1Description of the Building Block

set	pwm	pin	IO2 🔻	use cha	nnel CH0 (LT0) 🔻	out 0
					✓ CH0 (LT0)	
					CH1 (LT0)	
					CH2 (LT1)	
					CH3 (LT1)	
					CH4 (LT2)	
					CH5 (LT2)	
					CH6 (LT3)	
					CH7 (LT3)	
					CH8 (HT0)	
					сно литол	<b>•</b>

Set the servo's channel and output (rotation) angle to the specified PWM pin.

#### Step 2Write the Program

Set the pin IO33 (control pinio33) connected to the servo to "**Output**" mode.



KidsBlock Desktop 2.0.1			
kidsblock 🛛 🕀 🕶	Edit 🔡 kidslOT	😽 Unconnected 📄 File	
😂 Code 🔍 Costumes	() Sounds		
Events Events		· · · · · · · · · · · ·	
Control when Arduino begin	when	Arduino begin	
	set p	in IO33 ▼ mode output ▼	
Control	set p	wm pin IO33 👻 use channel CH2 (LT1) 👻 out 1	90
Variable: wait 1 seconds	wait	0.5 seconds	
My Blocks			

Initialize the control channel of the servo to CH2 (LT1) and the initial angle to 190°, with a delay of 0.5 seconds.

The servo rotates from  $190^\circ$  to  $120^\circ$  and then to  $60^\circ$  every 0.5 seconds.

🧕 KidsBlo	ock Desktop 2.0.1																
kids	block	•	Edit		kidslOT	<b>%</b> Ur	nconnected								File		
Coc	de 🦪 Costu	mes	🌒 Soun	ds													
events	Events																
Control	when Arduino b	egin			whe	n Arduin	o begin										
Operator					set p	oin IO3	33 🔻 m	ode	out	out 🗢				-			
	Control				set p	owm pin	IO33 •	- u	ise ch	annel	CH	12 (L1	[1) ▼	0	ut 🚺	90	
Variable	wait 1 see	conds			wait	0.5	seconds										
My Blocks	repeat 10				fore	/er											٦
Pins		و					i- 1022			chann					·	100	
Serial	forever					a pwin p			use	chann		582	(LII)		out	190	
Sensor		•			Wa	ait 0.5	secon	ds									
Data					se	t pwm p	in IO33	3 🔻	use	chann	el	CH2	(LT1)	•	out	120	
Variable	if th	en			wa	ait 0.5	secon	ds									
Туре					se	t pwm p	in IO33		use	chann	el	CH2	(LT1)	-	out	60	Ċ.
TEXT	if th	en			wa	ait 0.5	secon	ds									
	else																E
-																	-

Complete Program

Kitchblock     Edit     # kidslOT     K Unconnected     File     O     Item with the provided firming       # Code     Costumes     Item view     Item view     Item view     Item view	are 🔆 Tuto
Code Costumes Sounds	
Events     1     // generated fl       Events     2     #include <ardu< td="">       Octor     when Arduino begin     3       When Arduino begin     4     void setup() +       5     ledcstruck       6     ledcstruck</ardu<>	<pre>by KidsBlock uino.h&gt; { , 490, 8); in(33, 2):</pre>
Operator       Control       Set pim 1033 • mode output •       7       pinMode(33, 8)         Variable:       wait 1 seconds       9       delay(0.5 •	OUTPUT); , 190); 1000);
Wy         Wait         0.5         seconds         12         void loop() {           Blocks         repeat         10         forever         13         leddwite(2, 14         delay(0.5 *           Pins         Forever         16         delay(0.5 *         17         leddwite(2, 14	, 190); 1000); , 120); 1000); , 60);
Serial Sensor wait 0.5 seconds	1000);
Other     Set pwm pin     IO33 • use channel     CH2 (LT1) • out     120       Wait     0.5     seconds	
Type TEXT ff then else then then then then then then then then then	

## (2). Test Result

£

Upload

Click to upload the above complete code to the kidsIOT motherboard, then power up via the USB cable, then servo will rotate.

### 6. Read the Value of Ultrasonic Sensor



# (1). Programming Steps

### Step 1Add the Ultrasonic Sensor

Tap the "Sensor" module in the "Extension", then select "**Ultrasonic Sensor**" and click to return to the programming interface.



🥉 KidsBl	ock D	esktop	2.0.1														
kids	sbl	DC	K	¢	€ -	Edit		:	kidsk	от	1	Un 🖁	connec	ted	Ki	•	ile
Co	de		Cost	umes		<b>(</b> ) S	ounds	3									
Blocks	Ult	rasoi	nic														
Pins		D_Q	HC-S	R04	trig pin	102	2 🕶	ech	o pin	106	•	read d	listance	cm	•		0
Serial									•		•	+	•				
Sensor																	
Data																	
Variable Type																	
TEXT																	
OLED																	
WIFI	1																Ð
Da Ultrason	1															. (	<b>a</b>

# Step 2Description of the Building Block

D.(	Q) H	IC-SF	R04 ti	rig pin	10	2 💌	echo	pin	106 -	•	read	distance	cm 👻	
												√ cm		
												inc	n	

This block is used to measure distance to the specified pin, and the distance unit can be cm or inch.

# Step 3Write the Program

Set the baud rate to 15200.

🥉 KidsBl	lock Des	ktop 2.0.1												
kids	s <mark>bl</mark> o	ck	⊕-	Edit		kidslOT	۲	Uncor	inected		Kids	3lo	Fil	e
Co	de	🖌 Cos	tumes	() So	unds									
My Blocks	Seria seria	al al 0 -	begin ba	udrate	115200									
Pins	seria	al 0 🗸	print H	ello Kids	Block	when	Ardu	ino b	egin					
Serial Sensor	seri seri	al 0 ▼ al 0 ▼	availabl	e data ler byte	ngth	serial	0	• b	egin I	baudr	ate	1152	00 🔻	

Set the Trig pin of the ultrasonic sensor IO18 to "output" mode, and the Echo pin IO19 to "input" mode.

kids	block	🌐 🔻 Edit	📑 kidslOT	😽 Unconnected		File
Coc	le 🚽 Costume	es 🌒 🌒 Sou	nds			
Events	Events					
Control	when Arduino beg	in	when	Arduino begin		
Operator	Control		serial	0 - begin t	baudrate	115200 -
Variable:	wait 1 secor	nds	set pi	n IO18 🕈 m	ode outr	out 🗸
My Blocks	repeat 10		set pi	n IO19 🔻 m	ode inpu	it 🗢

🥉 KidsBlock Desktop 2.0.1

Set Trig to IO18 and Echo to IO19, and the serial port prints the distance value detected by the ultrasonic sensor at 0.1 second intervals.

KidsBlock Desktop 2.0.1			
kidsblock 🗣 Edit 😫	kidslOT 🙀 Unconnected	3.2_Ultrasonic-Sensor	File
😴 Code 🚽 Costumes 🌒 Sounds			
Events	· · · · · · · · · · · · ·		
when Arduino begin	when Arduino begin		
Control	serial 0 ▼ begin baudrate 115200 ▼		
Operator Control	set pin IO18 ▼ mode output ▼		
Variable: wait 1 seconds			
My	set pin IO19 ▼ mode input ▼		
Blocks repeat 10	forever		
Pins J	serial 0 - print D_Q HC-SR04 trig pin IO18	echo pin IO19	read distance cm 🔹 🛛 warp 💌
Serial forever	wait 0.1 seconds		
Sensor J			

#### **Complete** Program

KidsBlock Desktop 2.0.1				
kidsblock 🕀	দ Edit 🖀 kidslOT 💥 Unconnected		File 🕻	Download firmware 🔅 Tutorials
Code Costumes	() Sounds			1 Upload
Events	when Arduino begin			<pre>1 // generated by KidsBlock 2 #include <arduino.h> 3 #include <ultrasonic.h></ultrasonic.h></arduino.h></pre>
Control when Arduino begin	serial 0 ▼ begin baudrate 115200 ▼			<pre>4 5 Ultrasonic ultrasonic_18_19(18, 19); 6</pre>
Operator Control	set pin IO18 ♥ mode output ♥			<pre>7 void setup() { 8 Serial.begin(115200); 9 niMode(18 OUTDUT); </pre>
Variable: wait 1 seconds	set pin IO19 🔻 mode input 👻			10 pinMode(19, INPUT); 11 }
My Blocks repeat 10	forever			12 13 void loop() { 14 Serial.println(ultrasonic_18_19.rea
Pins	serial 0 v print (D.Q) HC-SR04 trig pin 10	18 ▼ ] echo pin [ IO19 ▼ ]	read distance cm  warp	
Serial forever				

#### (2). Test Result



to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click in the serial monitor and set the baud rate to 15200. Move your hand in front of the ultrasonic sensor. When you are close to it, the displayed distance value becomes smaller. When you move away from it, the value becomes larger.

10 14 12 10 8 7 5	00 <b>匝</b>
3 6 10 12 14 16	Buadrate 115200 <b>2</b> End of line LF & CR <b>7</b>
17 19 22 22 26	Hex form
	Send 🔀

7. Automatic Feeding System



## (1). Programming Steps

## **Step 1Flow Chart**



### Step 2Programming Steps

Set the baud rate to 15200, the Trig pin of the ultrasonic sensor IO18 to "**output**" mode, and the Echo pin IO19 to "**input**" mode.

KidsBlock Desktop 2.0.1									
kids	block 🛛 🕀 🕶	Edit	kidslOT 🍾 Unconnected 3.3_Intelligent-Feeding-Sy	File					
Cod	le 🥒 Costumes	() Sounds							
Events	Events		when Arduing begin						
Control	when Arduino begin		serial 0 = begin baudrate 115200 =						
Operator	Control								
Variable:	wait 1 seconds								
My			set pin IO19 ▼ mode input ▼						

Set the pin IO33 connected to the servo to "**Output**" mode, initialize the control channel of the servo to CH2 (LT1) and the initial angle to 190°, delay 0.5 seconds.

🥉 Kidsl	Block Desktop 2.0.1											0	
kid	sblock @•	Edit 📋	kidslOT 🕻	Unconnecte	d			3.3_Inte	lligent-F	eeding-S	y (		
🖉 🖉 C	ode 🧹 Costumes	() Sounds											
Events	Events	· ·											
Contro	when Arduino begin	· · ·	, when A	rduino beg	in ,								
Operato	DI Control		serial	0 🔻 beg	jin baudra	ate 11520	0 🔫						
Variable	wait 1 seconds		set pin	IO18 🔻	mode	output 🔻							
My Blocks	repeat 10		set pin	IO19 🔻	mode	input 👻							
Pins			set pin	IO33 <del>-</del>	mode	output 👻							
Serial	forever		set pwn	n pin 🛛 IO3	i3 <b>▼</b> us	e channel	CH2	2 (LT1)	• 0	ut 19	90		
Sensor	<b>J</b>		wait	0.5 seco	nds								

Define a "Distance" global variable to store the distance value detected by the ultrasonic sensor.

🍯 KidsBl	ock Desktop 2.0.1								
kids	block 🕀 🕶	Edit 📳	kidslOT 🎽	Unconnected		3.3_Intelligent-	Feeding-Sy	File	
Co	de 🚽 Costumes	() Sounds							
Events	Events	·							
Control	when Arduino begin		when Ardu						
Operator			serial 0	<ul> <li>begin baudrate</li> </ul>	115200 -				
	Control		set pin	O18 🔻 mode ou	itput 👻				
Variable	wait 1 seconds		set pin 10	O19 👻 mode in	put 👻				
My Blocks	repeat 10		set pin IC	O33 - mode ou	itput 👻				
Pins	•						4.00		
Serial	forever		set pwm pi		nannei CH2 (	LI1) 🔻 ol	190		
Concer			wait 0.5	seconds					
Data	if then		Declare	Global 🔻 variable	Type int 🔻	Name Dis	stance As	signed to	0

Set the Trig pin and Echo pin of the ultrasonic sensor, and print the read distance value of the ultrasonic sensor on the serial port.

KidsBlock Desktop 2.0.1				
kidsblock @• Edit #	kidslOT 🙀 Unconnected	3.3_Intelligent-Feeding-Sy	File	C.
🛫 Code 🚽 Costumes 🌒 Sounds				
Events	when Arduino begin			
Control when Arduino begin	serial 0 ▼ begin baudrate 115200 ▼			
Variable: wait 1 seconds	set pin IO18 ▼ mode output ▼			
My Blocks repeat 10	set pin 1019 • mode input •			
Pins J	set pwm pin IO33 ▼ use channel CH2 (LT1) ▼	out 190		
Serial forever	wait 0.5 seconds			
Data if then	Declare Global ▼ variable Type int ▼ Name	Distance Assigned to		· · · ·
Variable Type	Set Distance variable by D_Q HC-SR04 tr	ig pin IO18 🔻 echo	pin IO19 ▼ rea	ad distance cm 🕤
TEXT else	serial 0 • print variable Distance warp			
<u></u>	t			=

Determine the distance detected by the ultrasonic sensor. If 2cm < distance value < 7cm, the feed box will be opened; otherwise, it will be closed.

KidsBlock Desktop 2.0.1			
kidsblock 🗣 Edit 😩	sidsIOT 😽 Unconnected	3.3_Intelligent-Feeding-Sy	File
🛫 Code 🚽 Costumes 🌗 Sounds			
Events	when Arduino begin		
Control when Arduino begin	serial 0 ▼ begin baudrate 115200 ▼		
Operator Control	set pin IO18 V mode output V		
Variable: wait 1 seconds	set pin IO19 ▼ mode input ▼		
My	set pin IO33  mode output		
Blocks repeat 10	set pwm pin 1033 • use channel CH2 (LT1) • out 190		
Pins J	wait 0.5 seconds		
Serial forever	Declare Global  variable Type int  Name Distance Assig	ined to 0	
Sensor	, forever		
Data if then	Set Distance variable by D=Q HC-SR04 trig pin 1018 -	echo pin IO19 - read dis	stance cm 🔻
Variable Type	serial 0 • print variable Distance warp •		
TEXT if then	if variable Distance > 2 and variable Distance > 2	stance < 9 then	
	set pwm pin IO33    use channel CH2 (LT1)    out 90		
WIEI	wait 0.5 seconds		
wait until	set pwm pin IO33    use channel CH2 (LT1)    out 190		
repeat until	wait 0.5 seconds		
3			
	3		

Complete Program

🥉 KidsB	lock Desktop 2.0.1									
kide	sblock @	▼ Edit	kidslOT	🙀 Unconnected	3.3_Intelligent-Fe	. 📄 File	O			🔆 Tutorials
Co	ode 🦪 Costumes	() Sounds	3					💁 Upload		
Events Control Operato Variable My Blocks Pins Serial Sensor Data Variable Type	Events when Arduino begin Control wait 1 seconds repeat 10 forever		vitien Ardi serial 0 set pn 1 set pn 1 set pn 1 set pn 1 set prin set prin vat 0.0 Set 0 serial if vat set pi set	Lino begin baudrale 115200 • 1018 • mode output • 1019 • mode input • 1033 • use channel CH2 Seconds Clobal • variable Type int • Oldence variable Type int • Oldence variable Distance • 2 variable · Dis	RT11 • out 190       0         Rame Distance Assigned to 0       0         Rame Distance Assigned to 0       0         Rame Distance Assigned to 0       0         CL2 (L11) • out 00       0         Cl2 (L11) • out 90       0	1019 • read distance		1 // ( 2 #inu 3 #inu 4 5 int 6 9 voi 10 la 11 la 12 Sa 13 p: 14 p: 15 p: 16 la 17 da 18 } 19 20 voi	<pre>tenerated by Ki :lude <arduino. :lude <ultrasor Distance = 0; rasonic ultrasor i setup() { adcSetup(2, 496 adcAttachPin(33 erial.begin(111 inMode(13, 0UTF inMode(13, 0UTF inMode(13, 0UTF adcAttachPin(2, 196 elay(0.5 * 1006 d loop() {</ultrasor </arduino. </pre>	<pre>idsBlock .h&gt; iic.h&gt; &gt; onic_18_19(18, 19); 3, 8); 3, 2); 5200); DUT)</pre>

### (2). Test Result



to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click *in the serial monitor and set the baud rate to 15200. If an animal is detected within 2cm-9cm, the feed box will be opened to feed the animal.* 



### 8. Common Problems

#### Q1: Why doesn't the servo work?

A: It may be stuck. Before assembling the servo, use the code to adjust it to 80°.

### Q2: Why is the detection distance inaccurate when using the ultrasonic sensor?

A: Measurement should be started from the transmitting head of the ultrasonic sensor. This module is not a high-precision ultrasonic distance detection module and may exist errors.





# 4.3.4 Project 04: Anti-theft Alarm System

### 1. Description

The anti-theft alarm system is composed of a PIR motion sensor, a buzzer, a LED and a kidsIOT mainboard. When programming via the KidsBlock, you are able to judge whether someone is moving by reading the digital signal detected by the PIR motion sensor. If someone is moving, the buzzer will sound an alarm and the LED will flash to alert the user that someone has entered the area. Thus, a low-cost anti-theft alarm system can be realized, which is suitable for homes or offices.

# 2. Components



# **About Passive Buzzer and PIR Motion Sensor**

**Buzzer:** It is able to emit sounds of different frequencies and durations and is powered by DC voltage. What's more, it can be widely used in computers, alarms, electronic toys, automotive electronic equipment and so on to issue reminders or alarm sounds. There are active buzzers and passive buzzers.

**Passive buzzer:** Only by giving it "square wave signals" of a certain frequency can it make sound. Different square wave frequencies will produce different sounds (it can simulate tunes to achieve musical effects).



Note: The resonant frequency of the passive buzzer we provide is 2048Hz, and the sound is the loudest at this time.

#### **Parameters:**

Working voltage: DC 3.3V-5V Working current: (Max)30mA@5V Maximum power: 0.15mW Resonant frequency: 2048Hz Output sound pressure: (Min)80dB/10cm Control signal: square wave



**Active buzzer:** When connected to DC power supply, it will automatically emit sound (the frequency of the sound is fixed), and the sound is single.

**PIR Motion Sensor:** It is a sensor that detects infrared rays emitted by people or animals so as to output digital signals(1 or 0).

#### **Parameters:**

Working voltage: DC  $3.3V \sim 5V$ Working current: (Max)3.6mA@5VMaximum power: 18mWViewing angle: Y = 90°, X = 110° Detection distance:  $\leq 5m$ Control signal: digital signals(1 or 0).



Note: It should be away from the direct irradiation of sunlight, car headlights and incandescent lamps, or heaters and air conditioners so as to avoid the wrong alarms caused by the environmental temperature changes.

# 3. Assembly Steps

# Step 1Components Needed



# Step 2Process













Process 6Initialize the servo angle

Wiring of servo(it is the same as project 03)





First write the following code in KidsBlock software and upload the code to the kidsIOT mainboard, then the servo will rotate  $180^{\circ}$ . (Note: If the servo can not rotate, you can press the RESET button on the kidsIOT board.)

Process 7(Place the three Lego boxes on the same side, then assemble the four gears.)

















Complete 1




Process 11Share the LEGO board with project 03



Complete 2



# 4. Wiring Diagram

Module	kidsIOT Mainboard
PIR Motion Sensor	No.4 portcontrol pin is io27
Passive Buzzer	No.6 portcontrol pin is io23
Servo	G/V/io33 portBrown $\rightarrow$ GRed $\rightarrow$ VOrange $\rightarrow$ io33

Connect the kidsIOT mainboard to your computer via USB cable.



5. Passive buzzer makes sound



# Method 1

# (1). Knowledge

The passive buzzer is driven by square waves. Let's simulate the square waves below. The high and low levels of the pin can simulate a square wave: keeping the high level for 1000us and the low level for 1000us can make the buzzer sound.



Changing the time of high and low level can change the sound volume of the buzzer. You can try changing it to 1500us, 2000us, 3000us...

## (2). Write the Program

Initialize the buzzer's pin IO23 and "Output" mode.

🥉 KidsBl	ock Desktop 2.0.1	
kids	sblock - Edit 🖀	kidslOT 🤾 Unconnected 4.1 🎦 File
Co	de 🧹 Costumes 📢 Sounds	
My Blocks	Pins set pin 102 • mode input •	when Arduino begin
Pins	set digital pin 102 ▼ out high ▼	set pin IO23 ▼ mode output ▼

Set the buzzer pin **IO23** to "**High**" and "**Low**". Here we take the delay of 0.001 second (1000 microseconds) as an example to make the buzzer emit sound.

🥉 KidsBlo	ock Desktop 2.0.1							
kids	block 🛛 🕀 -	Edit 🔡 I	<b>cidsIOT</b>	Unconnec	ted	4.1_P	File	C
Cod	le 🥒 Costumes	() Sounds						
Events	Events							
Control	when Arduino begin		when A	rduino begi	n			
			set pin	IO23 🔻	mode	outpu	ut 🔻	
Operator	Control		forever					
Variables	wait 1 seconds		l cot di	aitol pip	022 -	out	biab .	
My	repeat 10		Set u		023 •	Jui	night -	
			wait	0.001 s	seconds			
Pins	3		set di	gital pin	023 👻	out	low -	
Serial	forever		wait	0.001	econds			
Sensor	3				¢			Ð
							. (	



Note: The conversion relationship between seconds, milliseconds and microseconds is: 1 second = 1000 milliseconds = 1000000 microseconds.

By f=1/T, changing high and low levels in 1000us, we can know that the frequency of such a square wave is 1000Hz (that is, the number of high and low level changes per second is 1000 times).

Complete Program

🍯 KidsBl	ock Desktop 2.0.1								
kids	block ⊕•	Edit	kidslOT	<b>V</b> Unconnected	4.1_P	File	0	Download firmware	. 🔆 Ти —
San Coo	de 🦪 Costumes	() Sounds					2.	Upload	dsBl
events	Events							<pre>1 // generated by Kid 2 #include <arduino.h 3</arduino.h </pre>	IsBlock
Control	when Arduino begin		when Ar	duino begin				<pre>4 void setup() { 5     pinMode(23, OUTPU 6     }</pre>	UT); JT);
Dperator	Control		set pin	lO23 <b>▼</b> r	node out	put 🔻		7 8 void loop() { 9 digitalWrite(23.	HIG (00);
ariable:	wait 1 seconds		set dia	ital nin IO	23 <b>–</b> out	bigh -		<pre>delay(0.001 * 100 11 digitalWrite(23, 12 delay(0.001 * 100</pre>	LOW); LOW); LOW);
My Blocks	repeat 10		wait	0.001 se	conds	, ingir •		12 1	
Pins	,		set.dic	ital pin 10	23 <b>-</b> out	low 💌			
Serial	forever		wait	0.001 se	conds				
Sensor	و			ر ا					
						6			

# (3). Test Result



Click to upload the above complete code to the kidsIOT motherboard. After powering up via the USB cable, the passive buzzer will make sound.

## Method 2

# (1). Knowledge

Use the "passive buzzer" code block to drive. The "passive buzzer" code block can generate a fixed-frequency PWM signal to drive the passive buzzer to sound. The sounding time length (beat) and sounding frequency can be controlled via parameters.

Å	Tone	e PIN#	IO23 🔻	freque	ncy NOTE_C3	duration	131
					NOTE_A4	<b>^</b>	
					NOTE_B4 NOTE C5		
					NOTE_D5		
					NOTE_E5 NOTE_F5		
					NOTE_G5 NOTE A5		
					NOTE_B5	<b>•</b>	

# (2). Add "passive buzzer"

Tap the "Actuator" module in the "Extension", then select "**esp32 Passive buzzer**" and click to the programming interface.







🥉 KidsBl	ock Deskt	top 2.0.1				14 - 1/3101				
kids	kidsblock 🕀 🕈 Edit 🖀 KidslOT 🦎 🕻									
Se Co	de	🧹 Costumes	s 📢 1) S	Sounds						
Pins	Passi	ve buzzer								
Serial	ĥ	Tone PIN#	IO33 -	frequency	NOTE_C3 -	duration 131				
Sensor	ß	Tone PIN#	IO33 🕶	play music	Birthday 👻	- 1 - 1 - 1				
Data	Å	noTone	033 🗸							
Variable Type										
TEXT										
OLED										
WIFI										
Passive buzzer										

(3). Description of the Building Block

) I	Ton	e PIN#	IO23 🔻	freque	ncy NOTE_C3 -	duration	131
					NOTE_A4	<b>^</b>	
					NOTE_B4 NOTE_C5		
					NOTE_D5 NOTE_E5		
					NOTE_F5 NOTE G5		· ·
					NOTE_A5		
					NOTE_B5	<b>V</b>	

Set the frequency and beat of the passive buzzer to the specified pin.

Å	Tone	e PIN#	IO33	•	play r	nusio	Birthday 🔻
							Pirthdau
							Tetris
							Ode to Joy
							Christmas
							Super mario
							Star war tone
						_	

sive buzzer to play specific music to the specified pin.



Set the passive buzzer to make no sound to the specified pin.

#### (4). Write the Program

Initialize the buzzer's pin IO23 and "Output" mode.



Set the sound pin, frequency and beat can be selected by yourself.

🍯 KidsBlo	ock Desktop 2.0.1									
kids	block 🌐	▼ Edit	<b>kidslO</b> T	Vinconnected		4.2_Pas	sive-Tone	Fi Fi	le	
Cod	de 🥒 Costumes	() Sound	s							
Events	Events			1 - 1 - 1						
Control	when Arduino begin		when	Arduino begin						
			set pin	IO23 🔻 n	node output	<b>-</b>				
	Control		foreve							
Variable	wait 1 seconds	н н		Tone PIN#	IO23 🔻 fi	requency	NOTE_C	3 🔻 dura	ation 131	
My Blocks	repeat 10	н н. С		و			IOTE C3			
Pins		ы н. С				· •	IOTE_D3			
Serial		ы н. Т				· •	IOTE_E3			
	torever	м м. С					IOTE_F3			
Sensor	و					· •	IOTE_A3			
Data	if then	н н. С				. N	IOTE_B3			
Variable Type		1 - F				· N	IOTE_D4			
							INTE EA	Ť		

Produce different tones.

KidsBlock Desktop 2.0.1											
kidsblock 🗣 Edit 🗄	kidslOT 😽 Unconnected	4.2_Passive-Tone	File								
😂 Code 🔮 Costumes 🌒 Sounds											
Events											
Control when Arduino begin	when Arduino begin										
	set pin IO23 ▼ mode output ▼										
Control	forever										
Variable: wait 1 seconds	Tone PIN# IO23 - frequ	uency NOTE_C3 -	duration 131								
My Blocks	wait 1 seconds										
Pins											
Serial	Tone PIN# IO23 ▼ frequ	uency NOTE_D3 •	duration 131								
forever	wait 1 seconds										
Sensor	Tone PIN# IO23 - frequ	uency NOTE_E3 🔻	duration 131								
Data if then	wait 1 seconds										
Variable Type	۲ (۲) (۲) (۲) (۲) (۲) (۲) (۲) (۲) (۲) (۲										
Data Variable Type	Tone PIN# IO23 ▼ freque wait 1 seconds	Jency NOTE_E3 ▼	duration 131								

# Complete Program

1	ð	KidsBlock Desktop 2.0.1	
	•	radio lo che o controp cion	

k	ids	block @	🕽 🔻 Edit	kidslOT	X Unconnected		4.2_Passiv	ve-Tone	File		0	🕆 Download firmware 🔅	π
	🕿 Cod	le 🖌 Costumes	() Sound	ds							🔔 Upl	oad	
Eve	ents	Events		when	Arduino begin	· ·	 				1 2 3 4	<pre>// generated by KidsBloc #include <arduino.h> #include <esp32tone.h></esp32tone.h></arduino.h></pre>	ck
Ope		Control		set pin foreve	IO23 🔻 mod	le outpu	ut 🕶 🛛				6 7 8 9	<pre>void setup() {     pinMode(23, OUTPUT);     pinMode(23, OUTPUT); }</pre>	
Varia N Blo	able: 1y icks	wait 1 seconds	<b>S</b>	wait	Tone PIN#	IO23 🔻	frequency	NOTE_C	3 🔻 dura	ation 131	11 12 13 14 15	<pre>void loop() {   tone(23, 131, 131, 0)   delay(1 * 1000);   tone(23, 147, 131, 0)</pre>	;
Pi Se	ns rial	forever		wait	Tone PIN#	1023 👻	frequency	NOTE_D3	3 🔻 dura	ation 131	16 17 18 19 20	<pre>delay(1 * 1000); tone(23, 165, 131, 0); delay(1 * 1000); }</pre>	;
Ser	nsor ata	if then			Tone PIN#	IO23 👻	frequency	NOTE_ES	B 👻 dura	ition 131			
Vari Ty	able pe			wait	1 seconds								

## (5). Test Result



Click to upload the above complete code to the kidsIOT motherboard. After powering up via the USB cable, the passive buzzer will make sounds with different tones.

#### 6. Passive buzzer plays music



#### (1). Write the Program

The buzzer pin is IO23, and then select a piece of music (we take Birthday as an example here).

🥉 KidsBl	ock Deskt	op 2.0.1								
kids	bloc	:k ∉	<b>} -</b> Edit		kidslOT	🙀 Unconnecte	d	4.3_Passive-M	lu 📄 File	
Co	de	🖌 Costumes	<b>()</b> St	ounds						
Pins	Passiv	ve buzzer								
Serial	Å	Tone PIN#	I033 <b>•</b>	frequenc						2 <b>س</b> ر .
Sensor	Å	Tone PIN#	1033 🗸	play mus	when A	Arduino begin				
Data	Ř	noTone K	033 🔹		<b>A</b>	Tone PIN#	IO23 🔻	play music	Birthday 👻	
Variablo								, , Bi	rthday	
Туре								· · · Oo	teris de to Joy	
TEXT								сі	hristmas	
								Su	ıper mario	
								St	ar war tone	
WIFI										$\odot$
Passive buzzer										Q

Complete Program

KidsBlock Desktop 2.0.1				
kidsblock 🗣 Edit 🕄	kidslOT 🦎 Unconnected	4.3_Passive File	o O	: Download firmware 🔅 Tu
😂 Code 🚽 Costumes 🌒 Sounds			🛃 Uploa	d
Pins Passive buzzer			1 2	// generated by KidsBlock #include <arduino.h></arduino.h>
Serial Tone PIN# 1033 • frequence			. 4	<pre>#include <esp32tone.h> #include<musicesp32.h></musicesp32.h></esp32tone.h></pre>
● ji Tone PIN# 1033 ▼ play mus	when Arduino begin		6	music Music(23):
	Tone PIN# IO23	play music Birthday	× 8 9	<pre>void setup() {</pre>
Data			10 11	<pre>pinMode(23, OUTPUT);</pre>
Variable			12	Music.birthday();

# (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, the passive buzzer will play a "Happy Birthday" music.

# 7. Read the value of PIR Motion Sensor



#### Step 1Write the Program

Set the baud rate to 15200.

🥉 KidsBl	KidsBlock Desktop 2.0.1													
kids	sblo	ock	⊕-	Edit		<b>kidslOT</b>	۲	Uncor	nected		Kids	Blo	File	
Co	de	🦨 Cos	tumes	() Soun	ds									
My Blocks	Seri	ial al 0 -	begin ba	udrate 11	5200									
Pins	seri	ial 0 🗸	print H	lello KidsBl	ock	when	Ardı	uino b	eain					
Serial	ser	ial 0 🗸	availabl	e data leng	th	serial		• h	egin l	baudr	ate	1152	00 🔻	
Sensor	ser	ial 0 💌	read a b	oyte					-gill i					

Set the pin IO27 connected to the PIR motion sensor to "input" mode.

kids	block @• Edit 🖀	kidslOT 😽 Unconnected	4.4
Coc	de 🥜 Costumes 🌒 Sounds		
Events	Events		
Control	when Arduino begin	when Arduino begin	
Operator		serial 0 ▼ begin baudrate 115200 ▼	
	Control	set pin IO27 - mode input -	
Variable	wait 1 seconds		

Define a "PIR\_motion\_sensor" global variable to store the value of the sensor.

🍯 KidsB	lock Desktop 2.0.1							
kid	sblock 🕀 -	Edit 🔡 kid	ISIOT 😽 Unconnected	4.4_P	IR-Motion-Senso	r 📄 🖻 F	ile	
Co	ode 🖌 Costumes	ຟາ) Sounds						
Events	Events							
Control	when Arduino begin	· · •	when Arduino begin					
Oporato			serial 0 🔻 begin baudrate 115200					
	Control		set pin IO27 ▼ mode input ▼					
Variable	wait 1 seconds		Declare Global 💌 variable Type ji	nt ▼ Nam	e PIR mot	ion senso	Assigne	ed to 0
My Blocks	repeat 10							

Store the read value of the sensor in the "PIR\_motion\_sensor" variable and print it on the serial port.

KidsBlock Desktop 2.0.1

👵 KidsBl	lock Desktop 2.0.1			
kids	sblock @• Edit 3	kidslOT 😽 Unconnected	4.4_PIR-Motion-Sensor	File
Co	ode 🥒 Costumes 📢 Sounds			
Events Control	Events when Arduino begin	when Arduino begin	· · · · · ·	
Operato	Control	serial 0   begin baudrate 115200 set pin 1027   mode input		
My Blocks Pins	repeat 10	Declare Global  variable Type int forever	Name PIR_motion_	_sensor Assigned to 0
Serial Sensor	forever	Set PIR_motion_sensor variable by serial 0   print variable PIR_m	otion_sensor warp -	
Data	if then	wait 0.1 seconds		

#### **Complete Program**

KidsBlo	ock Desktop 2.0.1						
kids	block @	) <del>-</del> Edit	🖀 kidslOT 🛛 🙀 Unconnected	4.4_PIR-Motion-Sensor	File	Ø	🔅 Download firmware 🔅 Tutorials
Cod	le 🥒 Costumes	() Sounds	S			<b>1</b>	lpload
Events	Events						1 // generated by KidsBlock 2 #include <arduino.h></arduino.h>
Control	when Arduino begin	1 - 1	when Arduino begin	<u>.</u>			<pre>int PIR_motion_sensor = 0;</pre>
			serial 0  begin baudrate 115200	•			6 7 void setup() {
Operator	Control		set pin IO27 ▼ mode input ▼	• • • • •			<pre>Serial.begin(115200); pinMode(27, INPUT); </pre>
Variable	wait 1 seconds	5		t - Name PIR motion	sensor Assigned to 0		2 void loop() {
My Blocks					_school Assigned to to	1	<pre>B PIR_motion_sensor=digitalRead(27); Serial.println(PIR_motion_sensor);</pre>
Pins			forever		<u> </u>	1	6 } 7
	, ,		Set PIR_motion_sensor variable	by read digital pin IO:	27 🗸		
	forever		serial 0 - print variable PIR_	motion_sensor warp -			
Sensor	و		wait 0.1 seconds				
Data	if then	· · · ·					
•		-					

#### **Step 2Test Result**

Click

🔔 Upload

to upload the above complete code to the kidsIOT motherboard. After powering up via the USB cable, click  $\swarrow$  in the serial monitor and set the baud rate to 15200.

When the sensor detects movement of a person or animal, the serial monitor window prints 1, and the red LED on the sensor will be off; otherwise, the monitor prints 0, and the red LED on the sensor will be on.

Note: The sensor does not have penetrating capabilities. When detecting human movement, please do not block it.

Buadrate 115200 V 2 End of line LF & CR V Hex form Auto scroll V 1		00 1
Buadrate 115200 V 2 End of line LF & CR V Hex form Auto scroll V 1	0	
End of line LF & CR    Hex form  Auto scroll	- 1 1	Buadrate 115200 V 2
Hex form Auto scroll I	1 1	End of line LF & CR V
1 Auto scroll 🗹	1	Hex form
	1	Auto scroll 2

# 8. Anti-theft Alarm System

Now, we will use a passive sensor, a servo and a PIR motion sensor to make an anti-theft alarm system.	
	/

(1). Programming Steps

#### **Step 1Flow Chart**



#### Step 2Write the Program

KidsBlock Desktop 2.0.1

Set the baud rate to 15200, the IO27 pin of PIR motion sensor to "input" mode.

 Unconnected ∰ kidslOT kidsblock Edit 🔚 Code Costumes Sounds Events Events Control serial 0 🔻 begin baudrate 115200 • Operator Control 1027 🗢 mode input 👻 set pin Variable 1

Set the pin IO33 connected to the servo to "Output" mode, initialize the control channel of the servo to CH2 (LT1)

and the initial angle to  $90^\circ$ , delay 0.5 seconds.

KidsBlock Desktop 2.0.1

ī.

kids	block 🛛 🕀 🕶	Edit \! 📋	kidslOT 🉀 Unconnected	4.5_Burglar-Alarm-Sy
Coc	de 🦪 Costumes	() Sounds		
Events	Events		when Arduino begin	
Control	when Arduino begin		serial 0   begin baudrate 115200	
Operator	Control		set pin IO27 ▼ mode input ▼	
Variables	wait 1 seconds		set pin IO33  mode output	
My Blocks			set pwm pin IO33 ▼ use channel CH2	(LT1) • out 90
Pins	repeat 10		wait 0.5 seconds	

Define a "PIR\_motion\_sensor" global variable to store the value of the PIR motion sensor.

KidsBlock Desktop 2.0.1				
kidsblock 🕀• Edit	🖀 kidslOT 🦞 Unconnected	4.5_Burglar-Alarm-Sy	File	Ľ
🛫 Code 🕜 Costumes 🌒 Sounds				
Events	when Arduino begin			
Control when Arduino begin	serial 0   begin baudrate 115200			
Operator Control	set pin IO27 ▼ mode input ▼			
Variable: wait 1 seconds	set pin IO33  mode output			
My Blocks	set pwm pin IO33 ▼ use channel CH2	e (LT1) ▼ out 90		
Pine Pine	wait 0.5 seconds			
	Declare Global  variable Type int	Name PIR_motion_s	ensor Assigned to 0	

Store the read value of the sensor in the "PIR\_motion\_sensor" variable.

🥉 KidsBlo	ock Desktop 2.0.1			
kids	block 🕀 Edit	👫 kidslOT 🛛 🙀 Unconnected	4.5_Burglar-Alarm-Sy	File
Cod	le 🖌 Costumes 📢 So	unds		
Events	Events	when Arduino begin		
Control	when Arduino begin	serial 0 ▼ begin baudrate 115200 ▼		
Operato	Control	set pin 1027 ▼ mode input ▼ .		
Variables	wait 1 seconds	set pin 1033 ▼ mode output ▼	H2 (I T1) 🗸 out 90	
Blocks	repeat 10	wait 0.5 seconds		
Pins		Declare Global   variable Type int	Name PIR_motion_s	ensor Assigned to 0
Serial	forever	forever		
Data		Set PIR_motion_sensor variable b	y read digital pin 1027	

Judge whether the sensor detects that a person or animal is moving. When someone or an animal is moving, the buzzer sounds, the servo rotates to close the door, and the serial monitor prints "Someone"; otherwise, the buzzer does not sound, the servo rotates to open the door, and the monitor prints "No one".

SkidsBlock Desktop 2.0.1									
kids	s <mark>block 🛛 🏵 -</mark>	Edit 🔡 ki	ādslOT 😽 Unconnected						
	do di Casturas	A Court							
- C0	Costumes	<ul> <li>Sounds</li> </ul>							
Events	Events								
Control	when Arduino begin		when Arduino begin						
Operator			set pin 1027 • mode input •						
	Control		set pin 1033 ▼ mode output ▼						
Variable	wait 1 seconds		set pwm pin 1033 • use channel CH2 (LT1) • out 90 wait 0.5 seconds						
My Blocks	repeat 10		Declare Global   variable Type int  Name PIR_motion_sensor Assigned to 0						
Pins			forever						
Sorial	5		if variable PIR_motion_sensor = 1 then						
	forever		serial 0 - print Someone warp						
Sensor	3		Tone PIN# 1023 • frequency NOTE_C3 • duration 131						
Data	if then		Tone PIN# 1023 • frequency NOTE_A4 • duration 131						
Variable Type			Tone PIN# 1023 • frequency NOTE_C5 • duration 131						
TEVT	if then		set pwm pin 1033 • use channel CH2 (LT1) • out 180						
			wait 0.5 seconds						
OLED			else serial 0 • print No one warp •						
WIFI			ji noTone 1023 ▼						
Passive	Wait until		set pwm pin 1033 • use channel CH2 (LT1) • out 90 wait 0.5 seconds						
DULLOI	repeat until								

Complete Program

KidsBlock Desktop 2.0.1								
kidsblock - Edit	👫 kidslOT 🛛 🙀 Unconnected	4.5_Burglar 🎦 File	<u>o</u> (	Download firmware				
Code Costumes (1) Sounds								
Events	when Arduino begin serial 0 • begin baudrate 115200 •			<pre>// generated by KidsBl #include <arduino.h> #include <esp32tone.h></esp32tone.h></arduino.h></pre>				
Operator Control	set pin 1027 • mode input • set pin 1033 • mode output • set pwm pin 1033 • use channel CH2	(LT1) • out 90	6 7 8 9	<pre>int PIR_motion_sensor void setup() {</pre>				
Variable: wait 1 seconds	wait 0.5 seconds Declare Global + variable Type int +	Name (PIR_motion_sensor) Assigned to 0	10 11 12 13 14	<pre>ledcSetup(2, 490, 8) ledcAttachPin(33, 2) pinMode(23, OUTPUT); Serial begin(115200)</pre>				
Pins	Set PIR_motion_sensor variable by r variable PIR_motion_sensor	read digital pin 1027 •	14 15 16 17 18	<pre>pinMode(27, INPUT); pinMode(33, OUTPUT); ledcWrite(2, 90); delay(0.5 * 1000);</pre>				
Serial Forever	Serial 0 - print Someone warp Tone PIN# 1023 - frequenc Tone PIN# 1023 - frequenc	v         NOTE_C3 •         duration         131           y         NOTE_C4 •         duration         131	19 20 21 22 23	<pre>} void loop() {     PIR_motion_sensor=di     if (PIR_motion_senso</pre>				
Data	Tone PIN# 1023 - frequence	y NOTE_A4 - duration 131	24 25 26	Serial.println("So tone(23, 131, 131, tone(23, 262, 131,				
TEXT If then	sel pwm pin 1033 • use channel wait 0.5 seconds	CH2 (LT1) - out 180						
OLED else	else serial 0 + print No one warp +							
Passive wait until	set pwm pin 1033 - use channel wait 0.5 seconds	CH2 (LT1) + out 90						
repeat until								

## (2). Test Result

# Click Upload

to upload the above complete code to the kidsIOT mainboard. After powering up via the USB

cable, click *in the serial monitor and set the baud rate to 15200.* 

When the sensor detects that someone or an animal is moving, the buzzer sounds, the servo rotates to close the door, and the serial monitor prints "Someone"; otherwise, the buzzer does not sound, the servo rotates to open the door, and the monitor prints "No one".

No one No one No one No one No one No one No one	00 10
No one No one No one Someone Someone Someone Someone Someone Someone Someone	Buadrate 115200 V 2 End of line LF & CR V Hex form Auto scroll V
	Send 2



## 9. Common Problems

#### Q1: The tone of the passive buzzer is not accurate to the actual tone?

A: The tones simulated by ordinary passive buzzers cannot meet the requirements of professional tones. If you need accurate tones, you need to use a more professional passive buzzer.

#### Q2: Does the PIR motion sensor make false alarms?

A: Non-professional PIR motion sensor.

The requirements for the sensor to avoid false alarms are as follows: Within the detection range, avoid objects that are caused by the wind, such as curtains, clothing, flowers, etc. Avoid interference from strong light within the detection range, such as sunlight, car lights, spotlights, lighting and other light sources.

# 4.3.5 Project 05Rainwater Control System

Note: Sprinkling water on other sensors will cause a short circuit and device failure. Sprinkling water on batteries will cause heating and explosion. Please be careful when using the device, especially when used by young children, it must be under the supervision of parents. To ensure safe operation of the device, please follow relevant usage guidelines and safety regulations.



#### 1. Description

This project explains how to use a steam sensor, a passive buzzer and a kidsIOT board to make a rain detection system. When the sensor detects rain, it sends a signal to the kidsIOT mainboard to trigger various actions.

For example, the buzzer can be used to sound an alarm to alert the user that it is raining. This system is able to monitor rainfall in gardening, agriculture, and detect leaks in roofs or buildings. The sensor can be easily connected to the kidsIOT motherboard to form a simple rain detection system.

# 2. Components



# 3. Assembly Steps

# Step 1Components Needed

	e va va va va va	મ્લે અને અને અને અને અને અને અને અને અને અન	
$\times 3$	$\times 2$		$\times 2$
	ł		
$\times 1$	$\times 17$	$\times 2$	$\times 6$
		Ì	

 $\times 1$ 

Note: The color of the building blocks is subject to the actual object.

HOOD

 $\times 3$ 

 $\times 6$ 

# Step 2Process

 $\times 1$ 

























Process 7






Complete



# 4. Wiring Diagram

Module	kidsIOT Mainboard
Steam Sensor	No.6 portcontrol pin is io36
Passive Buzzer	No.8 portcontrol pin is io5

Connect the kidsIOT mainboard to your computer via USB cable.



5. Read the analog value of the steam sensor



# Step 1Write the Program

Set the baud rate to 15200.

🥉 KidsBl	KidsBlock Desktop 2.0.1													
kids	s <mark>bl</mark> o	ck	<b>-</b>	Edit		kidslOT	۲	Uncon	nected		Kids	Blo	Fik	e
Co	de	🖌 Cos	tumes	() Sou	nds									
My	Seria													
Blocks	seria		begin ba	udrate 1	15200									
Pins	seria	l 0 -	print (H	lello KidsE	llock	when	Ardu	ino be	egin					_
Serial	seri	al 0 🗸	availabl	e data len	gth	seria	0	▼ b	egin l	baudr	ate	1152(	00 🗢	
Sensor	seri	al 0 🔻	read a l	oyte										

Set the pin IO36 connected to the steam sensor to "input" mode.

KidsBlock Desktop 2.0.1	
kidsblock 🕀 🕶 Edit 📰	kidsIOT 🙀 Unconnected 5.
😂 Code 🖌 Costumes 🌒 Sounds	
Events Events	when Arduino begin
Control when Arduino begin	serial 0 → begin baudrate 115200 →
Operator Control	set pin IO36 ▼ mode input ▼

Define a "Steam\_sensor" global variable to store the analog value of the steam sensor.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block 🕀 Edit 🏭	kidslOT 🙀 Unconnected	5.1_Steam-Sensor	File	C
Coc	de 🗸 Costumes 🌗 Sounds				
Events	Events	when Arduino begin			
Control	when Arduino begin	serial 0   begin baudrate 115200 ●			
Operator	Control	set pin IO36 ▼ mode input ▼			
Variables	wait 1 seconds	Declare Global 🗸 variable Type int	Name Steam_se	ensor Assigned to 0	

Store the read analog value of the sensor in the "Steam\_sensor" variable and print it on the serial port.

🍯 KidsBlo	ock Desktop 2.0.1								
kids	block - Edit	kidslOT	🙀 Unconnected		5.1_Steam	Sensor	File		D
Coc	ie 🚽 Costumes 📢 S	ounds							
Events	Events	when A	Arduino begin						
Control	when Arduino begin	serial	0 👻 begin baudr	ate 115200 👻					
Operator	Control	set pin	IO36 ▼ mode	input 💌					
Variable	wait 1 seconds	Declar	e Global 🔻 varia	able Type int •	Name	Steam_se	nsor Assi	gned to	0
Blocks	repeat 10	Set	Steam_sensor	variable by rea	ad analog p	in IO36			
Pins	9	seria	I 0 🔻 print va	riable Steam_	sensor	warp 🔻			-
Sensor	forever	wait	0.1 seconds						-

Complete Program

🥉 KidsBl	ock Desktop 2.0.1							
kide	block 🕀	Edit 🚦	kidslOT	😽 Unconnected	5.1_Steam-Sensor	File	0 (	🛱 Download firmware 🔆 Tutorials
Co	de 🚽 Costumes	() Sounds					🛓 Uple	oad
Events	Events		when Ar	duino begin			1 2 3	<pre>// generated by KidsBlock #include <arduino.h> int Steep second 0:</arduino.h></pre>
Control	when Arduino begin		serial	0 ▼ begin baudrate 115200			4 5 6	<pre>int Steam_sensor = 0;</pre>
Operato	Control		set pin	IO36 ▼ mode input ▼			8	<pre>Serial.begin(115200); pinMode(36, INPUT);</pre>
Variable	wait 1 seconds		Declare	Global  variable Type int	Name Steam_se	Assigned to 0	10 11 12 13	<pre>&gt; void loop() {     Steam sensor_analogRead(36);</pre>
My Blocks	repeat 10		Set	Steam_sensor variable by	read analog pin IO36		13 14 15 16	<pre>steam_sensor analogkead(s0); Serial.println(Steam_sensor) delay(0.1 * 1000); }</pre>
Pins	,		serial	0  variable Steam	n_sensor warp ▼	· · · ·	17	
Serial Sensor	forever		wait	0.1 seconds				

#### **Step 2Test Result**



to upload the above complete code to the kidsIOT motherboard. After powering up via the

USB cable, click  $\swarrow$  in the serial monitor and set the baud rate to 15200.

Then the serial monitor will print the value(range0~4095) read by the steam sensor. Touch the detection area on the sensor with a moistened finger, the larger the area, the greater the value!

1623	00
2019	<u>A</u>
2166	
2410	
2707	
2749	
2817	
2884	Buadrate 115200 V 2
2928	
2975	End of line LE & CD
3031	
3045	
3071	Hex form
3110	
3149	Auto scroll
3162	
	Send 💋

#### 6. Rainwater Control System



#### (1). Programming Steps

#### **Step 1Flow Chart**



## Step 2Add "passive buzzer"



### Step 3Write the Program

Set the baud rate to 15200, the IO36 pin of steam sensor to "input" mode.

KidsBlock Desktop 2.0.1										
kidsblock 🕀	Edit 🔡 kidslOT	🙀 Unconnected	5.2_R 📔 File							
Code Costumes	() Sounds									
Events	when	Arduino begin								
Control when Arduino begin	serial	0 👻 begin bau	udrate 115200 👻							
Variables wait 1 seconds	set pir	n IO36 ▼ mode	e input 🗸							

Define a "Steam\_sensor" global variable to store the analog value of the steam sensor.

🍯 KidsBlo	ock Desktop 2.0.1									
kids	block (	🕀 🗕 🖶	🕌 kidslOT	🙀 Unconnected		2_Rainwate	r-Detection	File		
Coc	de 🥒 Costumes	s 🌒 Sou	nds							
Events	Events									
Control	when Arduino begi	•	when A	Arduino begin						
Operator	Control		serial	0  begin baudrate	115200 -					
Variables	wait 1 second	ds	set pin	IO36 ▼ mode inpu						
My Blocks			Declare	e Global ▼ variable⊺	Гуре int ▼	Name	Steam_ser	nsor As	signed to	0

Receive analog value of the sensor and print it on the serial port.

🍯 KidsBl	KidsBlock Desktop 2.0.1										
kids	block 🕀	🗕 Edit	kidslOT 🙀 Unconnected 5.2_Rainwater-Detection	<b>File</b>							
Coo	de 🦪 Costumes	() Sounds									
Events	Events		when Arduino begin								
Control	when Arduino begin		serial 0 🔹 begin baudrate 115200 🔹								
Operator	Control		set pin IO36 ▼ mode input ▼								
Variable	wait 1 seconds		Declare Global  variable Type int  Variable Steam_sensor Assigned to	0							
My Blocks	repeat 10		Set Steam_sensor variable by read analog pin 1036 -								
Pins			serial 0 ▼ print variable Steam_sensor warp ▼ P = P = P								
Serial	forever										

Determine the received analog value of the sensor. When 800analog value<2000, the buzzer will sound alarm 1.

🍯 KidsBl	ock Desktop 2.0.1			
kids	iblack @- Edit \$	kidslOT 🙀 Unconnected	5.2_Rainwater-Detection	File
<b>5</b> Co	de 🖌 Costumes 📢 Sounds			
Events	Events	when Arduino begin		
Control	when Arduino begin	serial 0 ▼ begin baudrate 115200 ▼		
Operato	Control	set pin IO36 ▼ mode input ▼		
Variable	wait 1 seconds	Declare Global ▼ variable Type int ▼ Name S	Steam_sensor Assigned to	
My Blocks	repeat 10	Set Steam sensor variable by read analog bin	IO36 -	
Pins	,	serial 0 • print variable Steam_sensor wa	arp 🗸 🔹 k	
Serial	forever	if not variable Steam_sensor < 80	0 and variable	Steam_sensor < 2000 then
Sensor	<b>a</b>	Tone PIN# IO5 ▼ frequency NOTE_C	C3 - duration 131	
Data	if then	Tone PIN# IO5   frequency NOTE_D	03 - duration 131	
Variable Type		Tone PIN# IO5 ▼ frequency NOTE_E	duration 131	
TEXT	if then	<b>.</b>		

Determine the received analog value of the sensor. When 2000analog value<4000, the buzzer will sound alarm 2.

kids	sblock @	🕽 🕶 Edit	kidslOT	😽 Unconnected			2_Rainwater-Dete		File			
Co	de 🥒 Costumes	() Sound	5									
events	Events		when Ard	uino begin								
Control	when Arduino begin		serial 0	begin baudrate	115200 🗢							
Operator			set pin	IO36 - mode inpu	ut 🔻							
Variables	Control		Declare	Global 👻 variable 1	Type int 🔻 N	lame Stear	m_sensor As	signed to	0			
My	wait 1 second		forever									
Blocks	repeat 10	- N - N	Set	Steam_sensor varia	ble by read ar	nalog pin IC	036 -					
Pins			serial	0 - print variable	Steam_sens	or warp	•					
Serial	forever		if 📢	not variable	Steam_sensor	< 800	and va	riable	steam_sen	sor <	2000	then
Sensor	3			Tone PIN# IO5	frequency	NOTE_C3 -	duration 1	31				
Data	if then	н н		Tone PIN# IO5	frequency	NOTE_D3 🗸	duration 1	81				
Variable Type				Tone PIN# IO5 •	frequency	NOTE_E3 🗸	duration 1	31				
TEXT	if then											
OLED	else		if 🔇	not variable	Steam_sensor	< 2000	and	ariable	Steam_se	nsor <	4000	then
WIFI				Tone PIN# IO5 •	frequency	NOTE_E4 🔻	duration 1	81				
Passive	wait until			Tone PIN# IO5 •	frequency	NOTE_F4 🔻	duration 13	81				
buzzer	repeat until	- n n		Tone PIN# IO5 ◄	frequency	NOTE_G4 🔻	duration 1	31				۲. <sub>©</sub>
				٠								(Q)

Determine the received analog value of the sensor. When analog value4000, the buzzer will sound alarm 3.

🥉 KidsBl	ock Desktop 2.0.1							
kids	s <mark>block</mark> — Ed	it 📑 kidslOT	😽 Unconnected	5.2_Rainv	vater-Detection	🏲 File		
Co	de 🥒 Costumes 📢	Sounds						
Events	Events	when Ardu	uino begin					
Control	when Arduino begin	serial 0	✓ begin baudrate 115200					
Operator	Control	Set pin I Declare	O36 ▼     mode     input ▼       Global ▼     variable Type     int	Name Steam_senso	r Assigned to	0		
Variables	wait 1 seconds	forever		and applica pin				
My Blocks	repeat 10	serial	0 • print variable Steam_	_sensor warp •				 
Pins	<b>,</b>	1 I I I I I I I I I I I I I I I I I I I	not variable Steam_se	nsor < 800 and	variable St	eam_sensor	< 2000	then
Serial Sensor	forever	, , ,	Tone PIN#     IO5 ▼     frequer       Tone PIN#     IO5 ▼     frequer	ncy NOTE_C3   durat	on 131			
Data	if then	Ĩ	Tone PIN# IO5 ▼ frequer	ncy NOTE_E3 👻 durati	on 131		н н -	
Variable Type		i i i	not variable Steam_se	nsor < 2000 and	variable S	iteam_sensor	< 4000	then
TEXT	if then	Ę.	Tone PIN#     IO5 ▼     frequer       Tone PIN#     IO5 ▼     frequer	ncy NOTE_E4 - durati	on 131			
OLED	else		Tone PIN# IO5 → frequer	ncy NOTE_G4 👻 durat	ion 131		 	 
WIFI	wait until	i i i	not variable Steam_sen	sor < 4000 then	ан — ан			
Passive buzzer	repeat until		Tone PIN# 105 ▼ frequer	ncy NOTE_G5 → durat	ion 131			
	9		Tone PIN# IO5	ncy NOTE_B5 - durati	on 131			
=	Operators		ۍ ا					• (=)

Complete Program

Article Prod       Cold       Educit       Machine Login       Image: Cold State Display	💧 KidsBl	ock Desktop 2.0.1			1					
Image: Control       40 Bounds         Image: Control       Image: Control         Image: Control       Image: Control<	kids	block	∰ <del>▼</del> Edit	kidslOT	Vinconnected	5.2_Rainw	rater-Detection 🛅 Fi	le	0 (	Download firmware 🔅
Events	📰 Co	de 🚽 Cos	tumes 🌒 🌒 Soun	ds					🛓 Uplo	ad
	Events Control Operator Jariables Pins Serial Sensor Data Sensor Data Variable Type TEXT OLED WIFI Passive buzzer	Events when Arduino Control wait 1 s repeat 10 forever	econds J then then J J	when A serial set pin Declare forever Set a i i i i i	rduino begin 0 • begin baudrate 115200 1036 • mode input • 9 Global • variable Type in Steam_sensor variable by 1 0 • print variable Steam_se 1 0 • print variable Steam_se 1 0 • print variable Steam_se 1 Tone PIN# 105 • frequ 1 Tone PIN# 105 • frequ	Name Steam_sensor read analog pin 1036 • n_sensor warp • ensor < 800 and ency NOTE_C3 • duratie ency NOTE_D3 • duratie ency NOTE_E4 • duratie ency NOTE_F4 • duratie ency NOTE_F5	Assigned to       0         Assigned to       0         Assigned to       0         Variable       Steam_sense         Image: transmission of trans	$ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0$	1 2 3 4 5 6 7 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 9 30 31 2 2	<pre>// generated by KidsBloc #include <arduino.h> finclude <esp32tone.h> int Steam_sensor = 0; void setup() { pinMode(5, OUTPUT); Serial.begin(115200); pinMode(36, INPUT); } void loop() { Steam_sensor-analogRes Serial.println(Steam_sensor &lt;2 tone(5, 131, 131, 0) tone(5, 131, 131, 0) tone(5, 134, 131, 0) tone(5, 134, 131, 0) tone(5, 349, 131, 0) tone(5, 349, 131, 0) tone(5, 349, 131, 0) tone(5, 880, 131, 0) tone(5, 880, 131, 0) tone(5, 988, 131, 0) tone</esp32tone.h></arduino.h></pre>

### (2). Test Result

Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, when the rain value detected by the sensor is larger, the buzzer alarm sound will be louder.



#### 7. Common Problems

#### Q1: Is the steam sensor waterproof?

A: The detection area can be exposed to water, but when detecting water, please be careful not to use too much water.

#### Q2: After the sensor detects water, the alarm still sounds after a long time?

A: The passive buzzer keeps alarming because there are still water stains in the detection area of the sensor. Just clean it.

## 4.3.6 Project 06Temperature and Humidity Control System



#### 1. Description

This project introduces how to use a kidsIOT mainboard, a temperature and humidity sensor, a fan and an OLED display to build an intelligent temperature and humidity control system.

The system can measure ambient temperature and humidity and control fans to cool down and dehumidify based on demand. When the temperature or humidity exceeds the set threshold, it will automatically turn on the fan to reduce the temperature or humidity in the environment below the set value to protect the animals and plants on the farm.

What's more, it enables to adjust the ambient temperature and humidity and display them on the OLED display.

# 2. Components

		Aumdity Emperature		4 ( 1 / 0 / 0 / 1 / 1 / 1 / 1 / 1 / 1 / 1 /
kidsIOT Main- board×1	Motor×1	Temperature and Humid- ity Sensor×1	Wire×2	Battery Holder×1
			AA Battery	
Fan×1	USB Cable×1	Temperature and Hu- midity System LEGO Pieces×1	AA BatteryNot provide×6	

### About motor, temperature and humidity sensor and OLED display

**Motor:** It is a module that can control the rotation of the motor via the voltage direction of the output signal ends IN+ and IN-, and it can control the output PWM signal to adjust the speed of the motor. It is suitable for applications where fan speed and rotation direction need to be adjusted such as computer cooling and industrial production.

#### **Parameters:**

Working voltage: DC 3.3V-5V Working current: (Max)200mA@5V Maximum power: 2W Speed: (2500±10%) rpm @3.3V; (16000±10%) rpm @5V

**Temperature and Humidity Sensor:** It is a digital output temperature and humidity sensor that uses special analog signal acquisition, conversion technology and temperature and humidity sensing technology to ensure its good stability. It contains a high-precision resistive temperature and humidity sensor and a resistive thermal sensor, and is connected to an 8-bit high-performance microcontroller.

#### **Parameters:**

Working voltage: DC 3.3V-5V Working current: (Max)50mA@5V Maximum power: 0.25W Temperature measurement range: -25°C-+60°C Temperature accuracy: ±2°C Humidity measurement range: 20-90%RH Humidity accuracy: ±5%RH

Aumoday temperature

OLED: It is a display featuring clear picture quality, small size and high brightness.

#### Parameters:

Working voltage: DC 3.3V-5V Maximum power: 0.06W Viewing angle: >160° Resolution: 128\*64 Communication mode: IIC Driver chip: SSD1306



# 3. Assembly Steps

## Step 1Components Needed



# Step 2Process



















Complete



## 4. Wiring Diagram

Module	kidsIOT Mainboard
Temperature and Humidity Sensor	No.6 portcontrol pin is io23
Motor	No.9 portIN+control pin is io18IN-control pin is io19

Connect the kidsIOT mainboard to your computer via USB cable, connect the external power supply and turn the DIP switch on the mainboard to ON end.



# 5. OLED Display



# (1). Programming Steps

Step 1Description of the Building Block

nit oled	width	128	heig	ht 6	4	I2C add	lress	0x78 (0x3c) 🔻
		-					v 0	x78 (0x3c)
							0	x7a (0x3d)

This block is used to initialize the OLED's width, height and an I2C address.

draw	/ line >	x0:	0	y0:	0,	x1:	32	y1:	16	color	white 🔻	
			-			-		-		/ wh	ite	
										bla	ick	
										inv	erse	

This is a command block for drawing a straight line from the initial position  $x0:0\ y0:0$  to the final position  $x1:32\ y1:16$ . The number in the block can be changed.

draw rect x	0	<b>y</b> : 0	width 32	height 16	color white -
					✓ white
					black
					inverse

This is a command block that draws a recta with a width of 32 and a height of 16 from the initial position x:0 y:0, the numbers can be changed.

fill rec	ct x:	0	y:	w	idth	32	heigh	t 16	color	white 🔻
									🗸 wt	nite
									bla	ack
									inv	/erse

This is a command block that draws a rectangle with a width of 32 and a height of 16 from the initial position x:0 y:0, the numbers can be changed.

draw circle	<b>x</b> : 16	<b>y</b> : 16	6 radius	8	color	white 🔻
					√ wh	ite
					bla	ck
					inv	erse

This is the command block that draws a circle with a radius of 8 starting from the initial position x:16 y:16.



This is the command block that fills a circle with a radius of 8 starting at an initial position of x:16 y:16.

draw	round	rect x	) y:	0	width	32	heig	ght 10	6 rad	dius 🧹	4 color	white 🔻
											v wt	ite
											bla	ick
											inv	verse

This is the command block that draws a round rectangle with width 32, height 16, and radius 4 starting from an initial position of x:16 y:16.

fill roun	d rect x:	0	<b>y</b> :	0	width	32	hei	ght	16	radius	4	color white 💌
												√ white
												black
												inverse

This is the command block that fills a round rectangle with width 32, height 16 and radius 4 starting from initial position x:16 y:16.

draw	triangle	<b>x0</b> :	0	y0:	0	, x1:	16	y1:	0	, x2:	8	<b>y</b> 2:	16	color	white 🔻	
														🗸 wł	iite	
														bla	ick	
														inv	/erse	

This is the command block to draw a triangle from three positions x0:0 y0:0, x1:16 y1:0 and x2:8 y2:16.

fill tria	angle	x0:	0	y0:	0	, x1: (	16	y1:	0	, x2: (	8	<b>y2</b> :	16	color	white 👻
										-				√ wh	iite
														bla	ick
														inv	verse

This is the command block that fills the triangle between the three positions x0:0 y0:0, x1:16 y1:0 and x2:8 y2:16.



This is a command block for setting text size and color and background color.



This is the command block that sets the cursor position.

print	Hell	o Kids	Block warp -	
			✓ warp	
			no-warp	

This is a command block for setting the way of printing strings on the OLED screen. "**warp**" means newline printing, "**no-warp**" means no newline printing.

oled display i	mage ∩_∩ ▼
-	<ul> <li>✓ ∩_∩</li> </ul>
	U_U
	U.
	†.
	1
	←
	$\rightarrow$
	•

This is the command block to set the OLED display pattern.



This is the command block to clear the OLED screen.



This is the command block to refresh the OLED screen and display the next content.



This is a command block that sets strings to start scrolling in a certain direction.



This is the command block to set stop scrolling.

### Step 2Write the Program

Initialize the OLED with width 128, height 64 and I2C address 0x78 (0x3c).

🧕 KidsBl	lock Desktop 2.0.1	
kids	Solock 🕀 - Edit 📑 kidslOT 😽 Unconnected 6.	1_OLED
Co	de 🖌 Costumes 🌗 Sounds	
Events	Events	
Control	when Arduino begin	
Operator	init oled width 128 height 64 I2C address	s 0x78 (0x3c) ▼
	Control	

Set the text size displayed on the OLED to 6x8, the text color to white and the background color to black.

🥉 KidsBl	ock Desktop 2.0.1										
kids	block @	🕽 🗕 Edit	<b>iii</b> kidslOT	🕌 Unce	onnected			6.1	_OLED		
Co	de 🛹 Costumes	() Sounds									
Events	Events										
Control	when Arduino begin		whei	n Arduino	begin						
Operator	Control		init o	led width	128	height	64 I2C	address	0x78 (0	x3c) 🔻	
Variable:	wait 1 second	s	set to	ext size	6x8 🔻	color	white 🔻	backgr	ound colo	black	•
My											

### OLED displays straight lines.

KidsBlock Desktop 2.0.1

kids	block @•	Edit 🔡 I	kidslOT 🛛 🦎 Unc	connected		6.1_OLED	File
Code	e 🖌 Costumes	() Sounds					
Events	Events						
Control	when Arduino begin		when Arduind	begin .			
Operator	Control		init oled width	128 height	64 I2C addr	ess 0x78 (0x30	<b>;)                                    </b>
Variable	wait 1 seconds		set text size	6x8 - color	white  bac	kground color	black 🔻
My Blocks	repeat 10		clear oled				
Serial	¢		draw line x	0: 0 y0: 0	), <b>x1</b> : 128	y1: 64 colo	r white 👻
Sensor	<b>,</b>		refresh ole	d see a			

The OLED displays the straight line and delays 1 second.

🥉 KidsBlo	ock Desktop 2.0.1												
kids	block 🕀	• Edit 📳	kidslOT	🕌 Unconne	cted			6.1	_OLED			File	e
Coc	de 🛹 Costumes	(I) Sounds											
Events	Events		init	oled width	28 heig	ht 64	12C	address	0x78	(0x3c)	•		
Control	when Arduino begin		set	text size 6x	8 🔻 colo	or wh	iite 🔻	backgr	ound co	olor b	ack 🔻		
Operator			fore	ever									
Variables	Control		c	lear oled									
My	Wait The Seconds		d	raw line x0:	0 <b>y0</b> :	0	, x1:	128 y	1: 64	color	white	-	
Blocks	repeat 10		· re	efresh oled				,					
Pins				vait 1 se	conds								
Serial	forever												
Sensor													

The OLED displays rectangle and delays 1 second.

🥉 KidsBl	ock Desktop 2.0.1					
kids	block 🕀	Edit	kidsl01	🙀 Unconnected	6.1_OLED	File
Co	de 🚽 Costumes	() Sounds				
events	Events			draw line x0: 0 y0: 0 , x1: (	128 y1: 64 color	white -
Control	when Arduino begin		' - I	refresh oled		
Operator				wait 1 seconds		
Variable:	Control			clear oled		n n n n
Mv	Walt T Seconds			draw rect x: 0 y: 0 width	28 height 64 cold	or white 💌
Blocks	repeat 10			refresh oled		
Pins				wait 1 seconds		
Serial	forever			<b>J</b>		· · · ·
Sensor						



KidsBlo	ock Desktop 2.0.1					
kids	block 🕀	• Edit	📑 kids	iIOT 🙀 Unconnected	6.1_OLED	File
Coc	ie 🖌 Costumes	() Sour	nds			
Events	Events			draw rect x: 0 y: 0 width	128 height 64	color white 👻
Control	when Arduino begin			refresh oled		
Operato	Control			wait 1 seconds		
Variables	wait 1 seconds			refresh oled		
My Blocks				fill rect x: 0 y: 0 width	128 height 64 col	or white 🔻
Pins	repeat 10			refresh oled		
Serial	forever			wait 1 seconds		
Sansor				<u>Ú</u> ran a s		

The OLED displays circle and delays 1 second.

🍯 KidsBlo	ock Desktop 2	2.0.1					
kids	block	⊕-	Edit	iii kids	sIOT 😽 Unconnected	6.1_OLED	File
Cod	le 🖌	Costumes	() Sour	nds			
Events	Events				wait 1 seconds		
Control	when Ard	uino begin		, I .	refresh oled		
Operator					fill rect x: 0 y: 0 width	128 height 64 color w	/hite 🔻
Variables	Control				refresh oled		
My	wait 1	seconds			wait 1 seconds		
Blocks	repeat 1	10			clear oled		
Pins		٦			draw circle x: 64 v: 32 radi	us 32 color white 🗸	
Serial	forever				rafrach alad		
Sensor		٦					
Data	a de la compañía de la	then			wait 1 seconds		$\mathbf{x}_{i} = \mathbf{x}_{i} = \mathbf{x}_{i}$
•		ulen					

The OLED displays fill circle and delays 1 second.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block <b>@</b> •	Edit 📑 kidslO	T 🦞 Unconnected	6.1_OLED	File
Cod	le 🥜 Costumes	() Sounds			
Events	Events		draw circle x: 64 y: 32 radiu	s 32 color white -	
Control	when Arduino begin		refresh oled		
Operator	Control		wait 1 seconds		
Variables	wait 1 seconds	1 - 1 - 1	clear oled	a a a a a	
My Blocks		1 - 1 - 1	fill circle x: 64 y: 32 radius	32 color white ▼	
Dinc	repeat 10	1 - 1 - 1	refresh oled	a a a a a a a	
	<i>y</i>		wait 1 seconds		
Serial	forever				

The OLED displays round rectangle and delays 1 second.

kids	block 🛛 🕀 -	Edit	kidslOT	🙀 Unconnected				6.1_0	LED			File				
Coc	de 🥜 Costumes	() Sour	nds													
Events	Events			clear oled			,	,								
Control	when Arduino begin		r r f	ill circle x: 64 y:	32 1	radius	32	color	white <							
Operator	Control		·	efresh oled												
Variable:	wait 1 seconds			vait <u>1</u> seconds						,						
My				clear oled												
Blocks	repeat 10			Iraw round rect x:	0 y:	0	width	128	height	64	radius	16	color	white	e 🔻	
Pins	3		· · ·	efresh oled												
Serial	forever			vait 1 seconds												-
Sensor	3			و												

The OLED displays fill round rectangle and delays 1 second.

🍯 KidsBlo	ock Desktop 2.0.1												
kids	block (	🕀 🕶 Edit	kidslOT	🙀 Unconnected			6.1_OLED		6	File			
Coc	ie 🥒 Costume	s 🌒 🌒 Sou	inds										
Events	Events		d	raw round rect x:	0 <b>y</b> : 0	width	128 heiç	ght 64	radius	16	olor	white 🖣	2
Control	when Arduino begi	in	re	efresh oled									
)perator	Control		n n	ait 1 seconds	a series a								
/ariable:	wait 1 secon	ıds	c	ear oled									
My Blocks			fi	I round rect x: 0	<b>y</b> : 0	width 1	28 height	64 r	adius (	6 col	or whi	ite 🔻	
Pins	repeat 10		re	efresh oled									
Serial	forever		w w	rait 1 seconds									
				<u>ر</u>									

The OLED displays triangle and delays 1 second.

🍯 KidsBl	KidsBlock Desktop 2.0.1												
kids	block 🕀	Edit	kidslOT	🙀 Unconnected		6.1_OLED	File						
Co	de 🚽 Costumes	() Sounds											
events	Events	н н	v	vait 1 seconds									
Control	when Arduino begin		, c	lear oled									
Operator	Control		f	ill round rect x: 0 y:	: 0 width 12	height 64	radius 16 color	white 👻					
Variables	wait 1 seconds		. r	efresh oled									
My			· •	vait 1 seconds									
Blocks	repeat 10		d	lear oled									
Pins	•		d	Iraw triangle x0: 0	y0: 0 , x1: 1	128 <b>y1</b> : 0	, <b>x2</b> : 64 <b>y2</b> : 64	color white 🔻					
Serial	forever		ſ	efresh oled									
Sensor	<del>ر</del> ا			vait 1 seconds				, , , ,					
	if then			<u> </u>				<u>, , (@</u> )					

The OLED displays fill triangle and delays 1 second.

🍯 KidsBl	ock Desktop 2.0.1												
kids	block 🕀-	Edit	idslOT	🙀 Unconnected			6.1_OLE			File			Ľ
Co	de 🦪 Costumes	() Sound	Is										
Events	Events		• • • •	ait 1 seconds	n - 1								
Control	when Arduino begin		ci d	ear oled									
Operator	Control		d	aw triangle x0:	<b>y0</b> : 0	, x1:	128 <b>y1</b> :	0,	<b>x2</b> : 64	y2:	64 co	lor wh	ite 🔻
Variable:	wait 1 seconds		re	fresh oled									
My			<b>.</b> .	ait 1 seconds									
Diocks	repeat 10		· · C	ear oled									
Coriol	3		i i	triangle x0: 0	<b>y0</b> : 0	, <b>x1</b> : 12	8 y1:	0 , <b>x2</b> :	64	<b>y2</b> : 64	4 color	white	
Senal	forever		re	fresh oled									
Data	3		, , w	ait 1 seconds									
	if then		, ,										

The OLED displays smile face and delays 1 second.

🥚 KidsBlo	ock Desktop 2.0.1																	
kids	block 🕀	- Edit	kidslOT	🙀 Unconnected					6.1_OLE					File				
Coc	de 🖌 Costumes	() Sound	ds															
Events	Events		, , w	ait 1 second	ls ,													
Control	when Arduino begin		d c	ear oled														
Operator	Control		fil	triangle x0: 0	y0:	0,	x1:	128	y1:	0	, <b>x2</b> :	64	<b>y</b> 2:	64	colo	r w	hite	-
Variable:	wait 1 seconds		re	fresh oled														
My			w	ait 1 second	ls													
Diocks	repeat 10		<b></b>	ear oled														
Coriol	<i>J</i>		ol	ed display image	<u></u>	•												
Sonsor	forever		re	fresh oled														
Data	5		w	ait 1 second	ls													
	if then			3														•



KidsBlock Desktop 2.0.1											
kids	block <b>O</b> -	Edit 😬	kidslOT	Vinconnected	6.1_OLED	File					
Coo	de 🖌 Costumes	() Sounds									
Events	Events			oled display image ∩_∩ ▼							
Control	when Arduino begin			refresh oled							
Operator	Control			wait 1 seconds							
Variable:	wait 1 seconds			clear oled							
My				oled display image U_U -							
Blocks	repeat 10			refresh oled							
Pins	<b>J</b>			wait 1 seconds							
Serial	forever			<b>3</b>							

The OLED displays cry face and delays 1 second.

🥉 KidsBl	ock Desktop 2.0.1					
kids	<mark>iblock</mark> ⊕•	Edit 📳 k	üdslOT 🦎 Unconnected	6.1_OLED	File	
Co	de 🚽 Costumes	<b>∢</b> ) Sounds				
Events	Events		oled display image U	_U 👻		
Control	when Arduino begin		refresh oled			
Operator			wait 1 seconds			
• Variables	vait 1 seconds		clear oled	i i i		
My			oled display image T	T •		
Blocks	repeat 10		refresh oled			
Pins	5		wait 1 seconds			
Serial	forever			A A		
Sonsor						

The OLED displays " $\uparrow$  " and delays 1 second.

KidsBlock Desktop 2.0.1 kidsblock ⊕ - Edit kidslOT 😽 Unconnected File 🔚 Code 🥒 Costumes () Sounds Events oled display image T\_T ▼ Events refresh oled Control 1 Operator Control clear oled Variables wait 1 seconds oled display image 1 🔻 My Blocks repeat 10 refresh oled Pins 1 Serial Sensor

The OLED displays " $\downarrow$ " and delays 1 second.

KidsBlock Desktop 2.0.1

kids	block	•	Edit	iii kidslOT	🙀 Unconnected	6.1_OLED	File
Co	de 🛹 Cost	umes	() Sounds	S			
Events	Events				oled display image 🕴 🔻		
Control	when Arduino	begin			refresh oled		
Operator	Control			<mark>.</mark>	wait 1 seconds		
Variable	wait 1 se	econds			clear oled		
My				· •	oled display image 斗 🔻		
DIOCKS	repeat 10				refresh oled		
Pins		٦			wait 1 seconds		
Serial	forever						
The OLED displays " $\leftarrow$ " and delays 1 second.



The OLED displays " $\rightarrow$ " and delays 1 second.

KidsBlock Desktop 2.0.1			
kidsblock @• Edit 🖀	kidslOT 🦎 Unconnected	6.1_OLED	File
😂 Code 🖌 Costumes 🔹 🌗 Sounds			
Events	wait 1 seconds		
Control when Arduino begin	clear oled		
Operator	oled display image 🛛 🔶 🔻		
Variables wait 1 seconds	refresh oled		
My	wait 1 seconds		
Blocks repeat 10	clear oled		
Pins	oled display image $\rightarrow \bullet$		
Serial forever	refresh oled		
Sensor	wait 1 seconds		
Data if then	J		

The OLED displays "" and delays 1 second.



Set the cursor position of the OLED to display the "Hello, KidsBlock" string at x:0 y:30 with a delay of 1 second.



**Complete** Program

0: 0 ,x1: 128 y1: 64 color white + ) y: 0 width 128 height 64 color white • dth 128 height 64 color white • 0 y: 32 radius 32 color white • 32 ius 32 0 y: 0 width 128 height 64 radius 16 color white • 0 dth 128 height 64 radius 16 color white • 0: 0 , x1: 128 y1: 0 , x2: 64 y2: 64 color white • 0 y0: 0 ,x1: 128 y1: 0 ,x2: 64 y2: 64 color white •

## (2). Test Result



Click to upload the above complete code to the kidsIOT motherboard. After powering up via the external power supply, the OLED display on the kidsIOT board displays various patterns and English letters.

### 6. Fan rotates



### (1). Programming Steps

### Step 1Add "DC Motor"

Tap Tap, click the "Actuator" module in the "Extension", then select "**DC Motor for esp32**" and click **Back** 

to return to the programming interface.





🥉 KidsBl	ock Deskt	op 2.0.1											
kids	bloc	:k (	🌐 <del>-</del> Edi	it	i kidslO	Г	😽 Unc	onnected			6.1_OL	ED	
Co	Code Costumes () Sounds												
My Blocks	DC Mo	otor				-							
<b>P</b> ins	R	Motor INA	# IO2 •	State	HIGH 🝷	INB#	106 🕶	State	IIGH 🔻				
Serial	R	Motor INA	# IO2 <del>•</del>	State	HIGH 👻	INB#	IO6 🔻	channle	CH0 (LT0)	•	analogWrite	4096	
Sensor		· · ·		, ,									1
Data													
Variable Type													
TEXT													
OLED													
WIFI													
DC Motor													

# Step 2Description of the Building Block

R	Moto	r INA#	102	•	State	HIGH 🔻	INB#	IO6 🔻	State	HIGH 🔻
					<ul> <li>I</li> </ul>	HIGH				
					Ľ	LOW				

Set the high and low level states of the motor INA pin and INB pin.

R	Motor IN	IA#	102 👻	State	HIGH •	INB	# IO6 🔻	channle CH0 (LT0)	analogWrite	e 4096
								🗸 СНО (LTO)	<b>A</b>	
								CH1 (LT0) CH2 (LT1)		
								CH3 (LT1)		
								CH4 (LT2) CH5 (LT2)		
								CH6 (LT3)		
								СН7 (LT3) СН8 (НТ0)		. (0
								CH0 (HTN)	<b>•</b>	· Q

Set the high and low level status of the motor INA pin and the analog output value of the INB pin in certain channels. If the INA pin is in a high-level state, the smaller the INB analog output value, the faster the fan rotates; and if the INA pin is in a low-level state, the larger the INB analog output value, the faster the fan rotates.

## Step 3Write the Program

The pin INA of the motor module is IO18, the level is low, the INB pin is IO19, the channel is CH0 (LT0), and the analog output value is 0, then the motor does not rotate.

🥉 KidsBl	ock Desktop 2.0.1												
kids	block 🛛 🕀 🕶	Edit 👫	kidslOT	Vinconnected				6.2	_Motor		File		
Cor	de 🦪 Costumes	() Sounds											
events	Events												
Control	when Arduino begin	· ·	when /	Arduino begin									
Operator			R	Motor INA#	IO18 🔻	State	LOW -	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	0
	Control												

Set the motor pin INA to low level and the analog output value of the INB pin to different values, then the motor rotates clockwise at different speeds.

🧯 KidsBlo	ock Desktop 2.0.1													
kids	block	<b></b>	Edit 🚦	<b>kidslO</b> T	8	Unconnected				6.2_M			File	
Cod	de 🦪 Costu	imes (	() Sounds											
Events	Events			foi	rever									
Control	when Arduino I	begin		1.	R	Motor INA#	IO18 🔻	State	LOW 🗸	INB#	IO19 🔻	channle	CH0 (LT0) 👻	analogWrite 100
Operator					wait	2 seconds								
Variables	Control	conds			R	Motor INA#	IO18 🔻	State	LOW 🗸	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite 150
My					wait	2 seconds								
Blocks	repeat 10				R	Motor INA#	IO18 💌	State	LOW 🗸	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite 200
Pins		3			wait	2 seconds								
Serial	forever				R	Motor INA#	IO18 🔻	State	LOW 🗸	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite 150
Sensor		•			wait	2 seconds	· ·							
Data	if t	nen			R	Motor INA#	IO18 🔻	State	LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite 100
Variable Type					wait	2 seconds								
TEXT	if 🚺 t	nen				£								

Set the motor to stop rotating for 3 seconds.

KidsBlo	ck Desktop 2.0.1	<u> </u>				N.I									
kids	block	Eq.		Kidsi	01						6.2_N	lotor			
🚝 Coo	le 🦪 Costur	nes 🌒 📢 🗤	Sounds												
ents	Events				R	Motor INA#	IO18 🔻	State	LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	
ntrol	when Arduino b	egin			foreve	r									
erator	Control				S	B Motor INA	ŧ IO18 ·	<ul> <li>State</li> </ul>	e LOW •	• INB#	IO19 🔻	channle	CH0 (LT0)	analogWrite	100
iable	wait 1 sec	onds			wait	2 secon	ls ,								
/ly bcks					S	6 Motor INA	ŧ IO18 ·	<ul> <li>State</li> </ul>	e LOW •	INB#	IO19 🔻	channle	CH0 (LT0)	analogWrite	150
ins	repeat 10				wait	2 secon	ls								
erial	forever	<i>3</i>			S	B Motor INA	ŧ IO18 ·	<ul> <li>State</li> </ul>	e LOW •	INB#	IO19 🗣	channle	CH0 (LT0)	analogWrite	200
nsor					wait	2 secon	ls								
ata					S	B Motor INA	ŧ IO18 ·	<ul> <li>State</li> </ul>	e LOW •	INB#	IO19 -	channle	CH0 (LT0)	analogWrite	150
) iable		en			wait	2 secon	ls								
/pe	if the	20			S	Motor INA	ŧ IO18 ·	State	E LOW	INB#	IO19 -	channle	CH0 (LT0)	analogWrite	100
=XT	else				wait	2 secon	ls								
					S	Motor INA	ŧ IO18 ·	<ul> <li>State</li> </ul>	EOW	INB#	IO19 -	channle	CH0 (LT0)	analogWrite	0
	wait until				wait	3 secon	ls ,								
)C						<u> </u>									

Set the motor pin INA to high level and the analog output value of the INB pin to different values, then the motor

rotates anticlockwise at different speeds.

KidsBlo	ck Desktop 2.0.1													
kids	block	<b>.</b>	Edit	kidslOT	🙀 Unconnected					6.2_Moto		File		
Sa Coo	de 🚽 Cost	tumes	() Sounds	Б										
_ Events	Events				Sc Motor INA#	IO18 🔻	State	LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	0
Control	when Arduino	begin			wait 3 seconds									
)perato	Control			·   •	Se Motor INA#	IO18 🔫	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	205
/ariable:	wait 1 s	econds		· •	wait 2 seconds	· ·								
My					Se Motor INA#	IO18 🔻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	155
Dine	repeat 10				wait 2 seconds									
Corial		٦		· •	Sc Motor INA#	IO18 🔻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	105
	forever			· •	wait 2 seconds	н н. С								
Data		<i>J</i>			S Motor INA#	IO18 <del>-</del>	State	HIGH 🔻	INB#	IO19 <del>-</del>	channle	CH0 (LT0) 🔻	analogWrite	55
	if the second se	then			wait 2 seconds									
Туре					S Motor INA#	1018 <del>-</del>	State	HIGH 🔻	INB#	IO19 <del>-</del>	channle	CH0 (LT0) 🔻	analogWrite	105
IEXT	if <b>C</b>	then		· .	wait 2 seconds									
DLED	else			· •	S Motor INA#	IO18 👻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	155
WIFI	wait until				wait 2 seconds									
DC Motor	repeat until				Se Motor INA#	IO18 👻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	205
					wait 2 seconds									
					<b>†</b>									

Set the motor to stop rotating for 3 seconds.

	K Desktop 2.0.1													
kids	block 🕀	- Edit 🚦	kidslO	т 🎽	Unconnected					6.2_Moto		File		
Coc	e 🖋 Costumes	() Sounds												
Events	Events			R	Motor INA#	IO18 👻	State	HIGH -	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	105
Control	when Arduino begin			wait	2 seconds									
Operator	Control			R	Motor INA#	IO18 🔻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	155
variable:	wait 1 seconds			wait	2 seconds									
My Blocks	report 10			R	Motor INA#	IO18 🔻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	205
Pins				wait	2 seconds									
Serial	forever			R	Motor INA#	IO18 🔻	State	HIGH 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	255
Sensor	<b>,</b>			wait	3 seconds	· ·								
Data					٦									

Complete Program

when A	rduino begin											
R	Motor INA#	IO18 •	S	ate	LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0)	analogW	rite 0	
Æ	Motor INA	# IO18	-	State	LOW	• INB#	# IO19 •	channle	e CH0 (LT0	) 🔹 analog	Write 100	
wait	2 secon	ds										
K	Motor INA	# IO18	•	State	LOW	• INB#	# IO19 •		e CH0 (LT0)	) 🔻 analog	Write 150	
	2 secon	ds .				-		• •				
Æ	Motor INA	# IO18	•	State	LOW	• INB#	# IO19 •	channle	e CH0 (LT0)	) 🔹 analog	Write 200	
wait	2 secon	ds										
K	Motor INA	# IO18	•	State	LOW	• INB#	# IO19 •	channle	e CH0 (LT0	) 🔻 analog	Write 150	
wait	2 secon	ds						• •				
Æ	Motor INA	# IO18	-	State	LOW	• INB#	# IO19 •		e CH0 (LTO	) 🔻 analog	Write 100	
wait	2 secon	ds										
Æ	Motor INA	# IO18	•	State	LOW	• INB#	# IO19 •	- channle	e CH0 (LTO	) 🔻 analog	Write 0	
wait	3 secon	ds ,										
K	Motor INA	# IO18	-	State	HIGH	<ul> <li>INB</li> </ul>	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 205	
wait	2 secon	ds										
K	Motor INA	# IO18	•	State	HIGH	<ul> <li>INB</li> </ul>	# IO19	<ul> <li>channl</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 155	
wait	2 secon	ds										
Æ	Motor INA	# IO18	•	State	HIGH	<ul> <li>INB</li> </ul>	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 105	
wait	2 secon	ds										
K	Motor INA:	# IO18	•	State	HIGH	• INB	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 55	
wait	2 secon	ds										
K	Motor INA	# IO18	•	State	HIGH	<ul> <li>INB</li> </ul>	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔹 analog	gWrite 105	
wait	2 secon	ds .							· ·			
K	Motor INA	# IO18	-	State	HIGH	INB	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 155	
wait	2 secon	ds										
K	Motor INA:	# IO18	•	State	HIGH	<ul> <li>INB</li> </ul>	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	gWrite 205	
wait	2 secon	ds										
Å	Motor INA:	# 1018	•	State	HIGH	✓ INB	# IO19	<ul> <li>channi</li> </ul>	e CH0 (LT0	) 🔻 analog	Write 255	
	3 secon	ds										

## (2). Test Result



Click to upload the above complete code to the kidsIOT motherboard. After powering up via the external power supply, the motor rotates clockwise at different speeds and stops for 3 seconds, and then rotates counterclockwise at different speeds and stops for 3 seconds.

#### 7. Read data from the temperature and humidity sensor



## (1). Programming Steps

### Step 1Add "temperature and humidity sensor"

Tap Tap, click the "Sensor" module in the "Extension", then select "DHT sensor for esp32" and click Back



to return to the programming interface.

KidsBlock Desktop 2.0.1		-
← Back	Choose an Extension	
Q Search All Shie	ld Actuator Sensor Display	Communication Other
DS1307 real time clock module	Encoder Encoder module	e e
Version Author 1.0.0 keyes	Version Author 1.0.0 keyes	DHT sensor for ESP32 DHT Temperature and humidity sensor module for ESP32
Help	Help	Version Author 1.0.0 keyes Help
		Not loaded



Step 2Description of the Building Block



Initialize the pin and mode of the temperature and humidity sensor (dht11, dht21 or dht22).



Read the temperature and humidity from the temperature and humidity sensor.

## Step 3Write the Program

Set the baud rate to 15200.

🥉 KidsBl	ock Desktop 2.0.1							_					
kids	block	⊕-	Edit	8	kidslOT	1	Uncor	nected		Kidsl	Blo	Fik	e
Coc	de 🦪 Costu	mes	() Sound	s									
	Serial												
My Blocks	serial 0 🔹	begin ba	udrate 11	5200									د د
Pins	serial 0 🗸	print H	ello KidsBlo	ck	wher	n Ardı	uino b	egin					
Serial	serial 0 🗸	availabl	e data lengti								4450	00 -	
Sensor	serial 0 •	read a b	yte		seria			egin i	baudr	ate	1152	00-▼	

Initialize pin IO23 of the temperature and humidity sensor, and select the dht11 mode.



The serial port prints the read temperature and humidity in the current environment every 1 second.



### Complete Program

🥉 KidsBloc	k Desktop 2.0.1							
kids	b <mark>lock</mark>	🗕 Edit 📲	kidslOT 🏼 🙀 Unconnected	6.3_DHT11	File	0	Download firmware	🔆 Tutorials 🛛 Upl
Sa Code	Costumes	() Sounds				🕹 Upl	oad	
Events Control	Events when Arduino begin		when Arduino begin	e 115200 -		1 2 3 4 5 6	<pre>// generated by Kiu #include <arduino.l #include="" <dht.h=""> DHT dht(23, 11);</arduino.l></pre>	dsBlock h>
Operator Variables	Control wait 1 seconds		DHT pin IO23 -	mode dht11 -		7 8 9 10 11	<pre>void setup() {     dht.begin();     Serial.begin(115) }</pre>	200);
My Blocks Pins	repeat 10		serial 0   print Temp serial 0   print ∭	perature: no-warp ▼	Jre ▼ Warp ▼	12 13 14 15 16 17	<pre>void loop() {    Serial.print("Ten    Serial.println(d)    Serial.println("Hun    Serial.print("Hun    Serial.print("Serial.print("Hun    Serial.print("Hun    Serial.print(</pre>	<pre>mperature:"); ht.readTemperature()); midity:");</pre>
Serial Sensor	forever J		serial 0   print Hum serial 0   print II	idity: no-warp •	▼ warp ▼	18	Serial.println(d	ht.readHumidity());
Data Variable	if then		wait 1 seconds					

## (2). Test Result



to upload the above complete code to the kidsIOT mainboard. After powering up via the USB

cable, click  $\swarrow$  in the serial monitor and set the baud rate to 15200. Then the serial port prints the temperature and humidity in the current environment.



## 8. Temperature and Humidity Control System



## (1). Programming Steps

## **Step 1Flow Chart**



## Step 2Write the Program

Initialize pin IO23 of the temperature and humidity sensor, and select the dht11 mode.



KidsBlock Desktop 2.0.1		
kidsblock 🗣 Edit 🕄	kidslOT 😽 Unconnected	6.4_Temperature an File
🛫 Code 🛹 Costumes 🔹 🌒 Sounds		
Events	when Arduino begin	
Control when Arduino begin	DHT pin IO23 - mode dht11	
Operator	init oled width 128 height 64 I2C ac	ddress 0x78 (0x3c) 🕶
Variable: wait 1 seconds	set text size 12x16 ▼ color white ▼	background color black

Initialize the width, height, I2C address, text size and color as well as background color of the OLED display.

Set the strings "Temper:" and "Humid:" to be displayed on the OLED.

🥚 KidsBlo	ock Desktop 2.0.1								
kids	block	<b>⊕</b> - E	dit 📋	kidslOT 🎽	Unconnected		6.4_Ten	perature an	. 📄 File
Coc	de 🚽 Costu	imes 📢	) Sounds						
Events	Events			when Arduin	o begin				
Control	when Arduino b	begin		DHT	pin IO23 🗸 n	node dht11	-	· ·	 
Operato	Control			init oled widt	h 128 height	64 I2C ad	ldress	0x78 (0x3c)	•
Variables	wait 1 se	conds		set text size	12x16 ▼ colo	white 🔻	backgr	ound color	black 🔻
My Blocks	repeat 10			clear oled					
Pins		ر ح		print Temp	er: warp 🔹				
Serial	forever			set cursor x:	0 y: 30				
Sensor		ر م		print Humi	d: warp 👻				
Data	if th	ien		refresh oled					

Define variables "temperature" and "humidity".

🥉 KidsBlo	ock Desktop 2	.0.1											
kids	block	<b>.</b>	Edit	kidslOT	😽 Uncon	nected		6.4_Te	nperature a	n ) (	<b>File</b>		
Coc	de 🚽 (	Costumes	() Soun	ds									
Events	Events			when A	rduino begin	- a a							
Control	when Ardu	ino begin		ti t	DHT pin	023 <b>-</b> ma	ode dht11	•					
Operator	Control			init oled	width 128	height	64 <b>I2C ac</b>	ldress	0x78 (0x3	ic) 🔻			
Variable:	wait 1	seconds		set text	size 12x10	5 🔻 color	white 🔻	backg	round colo	r blac	:k ▼		
My				clear ol	ed a s								
Blocks	repeat 1	0		set curs	or x: 0	y: 0							
Pins		٦		print	Temper:	warp 👻							
Serial	forever			set curs	or x: 0	<b>y:</b> 30							
Sensor		و		print	Humid: w	arp 🔻							
Data				refresh	oled								
Variable	if and the second secon	then		Declare	Global 💌	variable T	ype int 🖣	Nam	e tempe	rature	Assign	ed to	0
Туре				Declare	Global 🔻	variable T	ype int •	Nam	e humid	ity As	signed t	• 0	

Store the read temperature data into the "temperature" variable. The read humidity data is stored in the "humidity" variable.

👵 KidsBl	ock Desktop 2.0.1							
kids	block	🌐 🕶 Edit	kidslOT	😽 Unconnect	ed	6.4_Temperat	ure an	File
Co	de 🚽 Costu	mes 🌒 📢 S	Sounds					
events	Events							
Control	when Arduino b	egin	when A	Arduino begin		· · · ·		
				DHT pin IO23 💌	mode dht11 -	]		
Operator	Control		init ole	d width 128 heigt	nt 64 I2C addres	ss 0x78 (0x3c) 🕶		
Variable	wait 1 se	conds	set tex	tsize 12x16 <del>-</del> c	olor white 🔻 ba	ckground color bla	ck 🔻	
My			clear o	led				
Blocks	repeat 10		set cur	sor x: 0 y: 0				
Pins			print	Temper: warp 🔻				
Coriol			set cur	sor x: 0 y: 30				
Senar	forever		print	Humid: warp 🔻				
Sensor		J	refresh	oled				
Data			Declar	e Global 👻 varial	ole Type 🛛 int 💌 N	Name temperature	Assigned	to 0
•	if th	en	Declar	e Global 💌 varial	ble Type 🚺 💌 N	Name humidity A	ssigned to	0
Variable Type			forever				<u> </u>	
	if th	en	Set	temperature vari	able by 🚺 C	OHT get temperatu	ne 🔻	
			Set	humidity variable	e by 🥼 DHT	get humidity 🕶		
OLED	else			5				

Display temperature data and humidity data at the corresponding position on the OLED.



Determine the temperature and humidity value in the environment detected by the temperature and humidity sensor. When the temperature is greater than 29°C, or the humidity is greater than 80%RH, the fan will be turned on.



**Complete Program** 

when Arduino begin	
III DHT pin IO23 → mode dht11 →	
init oled width 128 height 64 12C address 0x78 (0x3c) -	
set text size 12x16   color white   background color black	
clear oled	
set cursor x: 0 y: 0 and a set and a set a set a set a	
print Temper: warp -	
set cursor x: 0 y: 30	
print Humid: warp - I	
refresh oled	
Declare Global  variable Type int  Name temperature Assigned to	0
Declare Global   variable Type int   Name humidity Assigned to 0	
forever	
Declare       Global        Variable Type       int        Name       humidity       Assigned to       0         forever       Set       temperature       variable by       Image: DHT get       temperature	
forever Set temperature variable by DHT get temperature Set humidity variable by DHT get humidity =	
Declare       Global        Variable Type       int        Name       humidity       Assigned to       0         forever       Set       temperature       variable by       Image: DHT get       temperature       Image: DHT get       Image: DHT	
Declare Global • variable Type int • Name humidity Assigned to 0   forever   Set temperature variable by Image: Comparison of the temperature Image: Comparison of temperature Image: Comparison of temperature   Set humidity variable by Image: Comparison of temperature Image: Comparison of temperature   Set humidity variable by Image: Comparison of temperature Image: Comparison of temperature   Set humidity variable by Image: Comparison of temperature Image: Comparison of temperature   Set humidity variable by Image: Comparison of temperature Image: Comparison of temperature   Set cursor x: 90 y: 0	
Declare Global < variable Type int < Name Numidity Assigned to 0   forever   Set temperature variable by Image: DHT get temperature n   Set humidity variable by Image: DHT get temperature n   set cursor x: 90 y: 0   print variable temperature warp <	
Declare Global      variable     forever     Set   temperature   variable   by     DHT get   temperature     Set   humidity   variable   by     DHT get   temperature     output     temperature     variable   by     temperature     temperature     variable   temperature     warp     refresh     output     temperature     warp     temperature     warp	
Declare Global < variable Type int < Name Numidity Assigned to 0   forever   Set temperature variable by Image: DHT get temperature Image: DHT get I	
Declare Global • variable type int • Name humidity Assigned to 0   forever   Set temperature variable by • • DHT get temperature • <th><math display="block"> \begin{bmatrix} 1 \\ 1 \\ 2 \\ 2 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3</math></th>	$ \begin{bmatrix} 1 \\ 1 \\ 2 \\ 2 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3$
Declare Global • variable lype int • Name humidity Assigned to 0   forever    Set temperature variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get temperature •   Set humidity variable temperature warp •   refresh oled Image: DHT get temperature variable   f variable temperature >   Image: Set Variable temperature >   f variable temperature >   Image: Set Varia	x       x       x       x         x
Declare Global • variable lype int • Name humidity Assigned to 0   forever   Set temperature variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get temperature •   Set humidity variable by Image: DHT get humidity •   set cursor x: 90 y: 0   print variable temperature warp •   refresh oled Image: DHT get temperature   Image: Set Motor INA# IO18 •   State LOW • INB# IO19 •   channle else	1       1       1         1       1       1         1       1       1         2       1       1         2       1       1         2       1       1         3       1       1         3       1       1         4       1       1         5       1       1         6       1       1         80       1       1         610       LTO       Total and
Declare Global • Variable Type Int • Name humidity Assigned to 0   forever   Set temperature variable by I DHT get temperature • 1   Set humidity variable by I DHT get humidity • 1   Set humidity variable by I DHT get humidity •   Set humidity variable by I DHT get humidity •   set cursor x: 90 y: 0 assistance   print variable temperature warp • variable   if variable temperature 29 or   Variable temperature 29 or variable   if variable temperature 29 or   Set Motor INA# IO18 • State LOW •   INB# IO19 • channle	1       1
Declare Global • Variable Type Int • Name numidity Assigned to 0   forever   Set temperature variable by Image: DHT get temperature • numidity	1       1       1       1         1       1       1       1         2       1       1       1         3       1       1       1         4       1       1       1         5       1       1       1         6       1       1       1         80       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1       1       1       1         1

## (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the external power, the OLED displays the temperature and humidity in the current environment.

When the temperature reaches 29°C or the humidity reaches 80%RH, the motor will turn on to dissipate heat or dehumidify (the fan simulates heat dissipation and dehumidification, and the heat dissipation and dehumidification effect is average); otherwise, the motor will turn off to achieve automatic farm temperature and humidity control effect.



### 9. Common Problems

#### Q1: Is the temperature and humidity sensor waterproof?

A: The sensor detects the temperature and humidity in the air, which is not waterproof. Please do not put the module into water.

### Q2: Does the rotation of the motor cause the kidsIOT mainboard to reset?

A: When the motor rotates, it requires a larger current than other sensors, which will cause voltage and current fluctuations in the circuit. Especially when the motor rotates forward and reverse, the voltage and current fluctuations are too large, causing the voltage and current of the kidsIOT mainboard to be too low, thus causing a reset.

# 4.3.7 Project 07Soil Moisture Detection System

Note: Do not allow water to overflow from sinks and soil troughs when using the device. Sprinkling water on other sensors will cause a short circuit and device failure. Sprinkling water on batteries will cause heating and explosion. Please be careful when using the device, especially when used by young children, it must be under the supervision of parents. To ensure safe operation of the device, please follow relevant usage guidelines and safety regulations.



### 1. Description

This project introduces how to use a kidsIOT mainboard, a soil moisture sensor, a passive buzzer and an OLED display to make an intelligent soil moisture detection system. The system can display the value of the soil moisture sensor in real time through the OLED display. When the soil moisture is lower than the set value, the buzzer will sound an alarm to remind you that it is time to water the land.

## 2. Components



# **About Soil Moisture Sensor**

**Soil Moisture Sensor:** It can be used to detect soil moisture and make automatic watering systems, flower pot soil moisture monitoring and automatic irrigation. It adopts a fork-shaped design for easy insertion into the soil. When it is inserted into the soil, the output voltage increases as the soil moisture temperature increases, and when the soil is short of water, the output value of becomes smaller, otherwise it will increase.



# 3. Assembly Steps

## Step 1Components Needed



## Step 2Process



























(wate pipe+water pump) $\times 1$ 








Process 11





Process 12





Process 13



Process 14





Complete



# 4. Wiring Diagram

Module	GPIO Shield	kidsIOT Mainboard
Soil Moisture Sensor	$G \rightarrow GV \rightarrow VS \rightarrow S3$	No.4 portcontrol pin of S3 is io27
Passive Buzzer		No.8 portcontrol pin is io5

Connect the kidsIOT mainboard to your computer via USB cable.



5. Read the value of soil moisture sensor



## (1). Write the Program

Set the baud rate to 15200.

🦲 KidsBl	lock De	sktop 2.0.1	I											
kids	sbl	ock	- (	Edit		kidslOT	1	Uncor	nnected		Kidsl	Blo	Fil	e
Co	de	J Cos	stumes	() Soun	ds									
	Ser	ial												
My Blocks	ser	ial 0 🗸	begin ba	udrate 11	15200									
Pins	ser	ial 0 🗸	print (	lello KidsBl	ock	wher	n Ardı	uino b	egin					
Serial	se	rial 0 🗸	availabl	e data leng	th	seria		<b>•</b> 1	egin l	baudr	ate	1152	00 🔻	
Sensor	se	rial 0 🗸	read a l	oyte					-ogiii i	Seraran		1102		

Set the pin IO27 connected to the soil moisture sensor to "input" mode.

KidsBlock Desktop 2.0.1

kids	block 🛛 🏶 •	Edit 🔛	kidslOT	😽 Unconnected		7.1_Soil-M
Coc	de 🥒 Costumes	() Sounds				
Events	Events					
Control	when Arduino begin		when Arc	duino begin		,
Operator	Control		serial (	) ▼ begin	baudrate	115200 -
Variable:	wait 1 seconds		set pin	lO27 ▼ n	node inp	ut 🔻

Define a "Soil\_humidity\_sensor" global variable to store the analog value of the soil moisture sensor.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block 🛛 🕀 -	Edit 🔡	kidslOT 🦎 Unconnected	7.1_Soil-Moisture-Sensor	File
Coc	de 🥒 Costumes	() Sounds			
Events	Events	· · ·	when Arduino begin		
Control	when Arduino begin		serial 0 ▼ begin baudrate 115200 ▼		
Operato	Control		set pin IO27 ▼ mode input ▼		
Variable	wait 1 seconds		Declare Global  variable Type int	Name Soil_humidity_s	sensor Assigned to 0

Store the read analog value of the soil moisture sensor in the "Soil\_humidity\_sensor" variable and print it on the serial port.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block 🕀 -	Edit 📑	idsIOT 🙀 Unconnected	7.1_Soil-Moisture-Sensor	File
Coc	le 🥒 Costumes	() Sounds			
Events	Events		when Arthring bogin		
Control	when Arduino begin		serial 0 - begin baudrate 115200 -		
Operator	Control		set pin IO27 ▼ mode input ▼		
Variable:	wait 1 seconds		Declare Global ▼ variable Type int ▼	Name Soil_humidity_s	ensor Assigned to 0
My Blocks			forever		
Pins	repeat 10		Set Soil_humidity_sensor variable by	read analog pin 1027	
Serial	forever		wait 0.5 seconds	waip •	
Sensor			J		

**Complete Program** 

KidsBlock Desktop 2.0.1			
kidsblock	Edit 🎬 kidslOT 💥 Unconnected	7.1_Soil-Moisture-Sensor File	🗿 🤃 Download firmware 🔅 1
🚝 Code 🚽 Costumes	an Sounds		1 Upload
Events	when Arduino begin		1 // generated by KidsBlocl 2 #include <arduino.h> 3 4 int Soil_humidity_sensor</arduino.h>
Control when Arduino begin	serial 0 ♥ begin baudrate 115200 ♥ set pin 1027 ♥ mode input ♥		5 6 7 void setup() { 8 Serial.begin(115200);
Variable: Wait 1 seconds	Declare Global  variable Type int  forever	Name Soil_humidity_sensor Assigned to	<pre>9 pinMode(2/, INPUT); 10 } 11  12 void loop() { 13 Soil_humidity_sensor=a)</pre>
Blocks Pins	Set Soil_humidity_sensor variable by serial 0 • print variable Soil_humid	read analog pin 1027 ▼ ity_sensor warp ▼	<pre>14 Serial.println(Soil_hu 15 delay(0.5 * 1000); 16 } 17</pre>
Serial forever	wait 0.5 seconds		

### (2). Test Result

Click Upload to upload the above complete code to the kidsIOT board. After powering up via the USB cable, click in the serial monitor and set the baud rate to 15200.

Insert the soil moisture sensor into the soil (or touch the sensor with wet hands), then the serial monitor will print the analog value of the soil moisture sensor (range:  $0 \sim 4095$ ). The greater the soil moisture (or the wetter the hand), the greater the value!

1623 1819 2019	00
2166	
2410	
2707	
2817	
2884	Bundards 115200 -
2928	Buadrate 115200 • 2
2975	
2997	End of line LF & CR V
3031	
3045	Hex form
3110	
3149	Auto scroll
3162	
	Send 🔀

6. Soil Moisture Detection System



(1). Programming Steps

# Step 1Flow Chart



## Step 2Add "passive buzzer"



### Step 3Write the Program

Set the pin IO27 connected to the soil moisture sensor to "input" mode.

🥉 KidsBl	ock Desktop 2.0.1				
kids	block	🕀 🕶 Edit	🥌 kidslOT	🙀 Unconnecte	ed 7.
Co	de 🚽 Costun	nes 🌒 S	ounds		
events	Events				
Control	when Arduino be	egin	whe	n Arduino begin	
Operato	Control		set p	oin IO27 ▼	mode input 🔻

Initialize the width, height, I2C address, text size and color as well as background color of the OLED display.

KidsBlock Desktop 2.0.1

kids	block 🕀-	Edit 📑 kids	OT 🎽 Unconn	ected		7.2_Soil-I	Humidity-Test	ti (
Coc	de 🦪 Costumes	() Sounds						
Events	Events							
Control	when Arduino begin		hen Arduino begin					
Operator	Control	i i	et pin IO27 - r nit oled width 128	node input -	I2C addr	ress 0x7	78 (0x3c) ▼	
Variable:	wait 1 seconds	s	et text size 6x8	color white	▼ bac	ckground (	color blac	

Define a "Soil\_humidity\_sensor" global variable to store the analog value of the soil moisture sensor.

🥉 KidsBlo	ock Desktop 2.0.1				
kids	block @•	Edit 📳	kidslOT 🦎 Unconnected	7.2_Soil-Humidity-Testi	File
Coc	de 🥜 Costumes	() Sounds			
Events	Events				
Control	when Arduino begin		when Arduino begin		
			set pin IO27 ▼ mode input ▼		
	Control		init oled width 128 height 64 120	C address 0x78 (0x3c) 🔻	
Variable	wait 1 seconds		set text size 6x8 ▼ color white ▼	background color black -	
My Blocks	repeat 10		Declare Global 🔻 variable Type i	nt ▼ Name Soil_humidity_s	sensor Assigned to 0

Store the read analog value of the soil moisture sensor in the "Soil\_humidity\_sensor" variable and display it on the OLED.

🥉 KidsBl	ock Desktop 2.0.1									
kids	block	🕀 🕶 Edi	t 🔡 kids	IOT 🦎 Unconne	ected	7.2_\$	oil-Humidity-Testi.	. 🏲 File		
Co	de 🦪 Costu	umes 🌒 📢	Sounds							
Events	Events									
Control	when Arduino I	begin		when Arduino begin						
Operator				set pin IO27 ▼	mode input 🔻					
Variable	Control			init oled width 128	B height 64	I2C address	0x78 (0x3c) ▼			
My	wait 1 se	conds		set text size 6x8	<ul> <li>color white</li> </ul>	backgrou	und color black			
Blocks	repeat 10			forever			le Soli_numian	y_sensor	Assigned	
Pins		٦		Set Soil humi	dity sensor var	iable by rea	d analog pin	27 -		
Serial	forever			clear oled			· · · ·			
Sensor		ر م		set cursor x: 0	y: 30					
Data	if t	hen		print Soil_Humi	idity: warp -					
Variable				set cursor x: 90	y: 30 .					
	if t	hen		print variable	Soil_humidity_se	ensor war	p 🗸 🔹 🗤			
	else			refresh oled			· · ·			
OLED				wait 0.3 seco	onds					
WIFI				و						$\odot$

Determine the analog value of the sensor. If it is less than 500, the buzzer will sound an alarm; otherwise, the buzzer will not sound.



Complete Program



#### (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, insert the soil moisture sensor into the soil (or touch the sensor with wet hands), when the sensor detects that the moisture of the soil (or the moisture of your hands) is less than the set threshold, the buzzer will sound an alarm.



### 7. Common Problems

#### Q: Is the sensor waterproof?

A: The detection area of the soil moisture sensor is waterproof. Exceeding the detection area will cause a short circuit.

# 4.3.8 Project 08Water Level Detection System

Note: Do not allow water to overflow from sinks and soil troughs when using the device. Sprinkling water on other sensors will cause a short circuit and device failure. Sprinkling water on batteries will cause heating and explosion. Please be careful when using the device, especially when used by young children, it must be under the supervision of parents. To ensure safe operation of the device, please follow relevant usage guidelines and safety regulations.



### 1. Description

This project introduces how to use a kidsIOT mainboard, a water level sensor, a passive buzzer and an OLED display to make an intelligent automatic water level detection system. It is able to monitor water level changes, detect problems in time and take measures to avoid disasters, and is widely used in water conservancy projects, urban drainage as well as environment monitoring.

### 2. Components



Water Level Sensor: It can determine the water level by measuring the water volume, and convert the water volume into analog signal. The output analog value can be directly read by development boards to achieve the effect of water level alarm.



## 3. Assembly Steps

It shares the same structural shape with **Project 07**. If the assembly parts of **Project 07** have finished, there is no need to assemble it again.

### 4. Wiring Diagram

Module	GPIO Shield	kidsIOT Mainboard
Water Level Sensor	$G \rightarrow GV \rightarrow VS \rightarrow S4$	No.4 portcontrol pin of S4 is io39
Passive Buzzer		No.8 portcontrol pin is io5

Connect the kidsIOT mainboard to your computer via USB cable.



5. Read the value of water level sensor



## (1). Write the Program

Set the baud rate to 15200.

🥉 KidsBl	lock Des	ktop 2.0.1	l											
kids	s <mark>bl</mark> a	ck	<b>.</b>	• Edit		kidslOT	8	Uncor	nected		Kids	Blo	File	
Co	de	🦨 Cos	stumes	() So	unds									
M	Seria	al												
Blocks	seria	al 0 🗸	begin b	audrate	115200									ر م
Pins	seria	al 0 🗸	print (	Hello Kids	Block	whe	n Ardı	uino b	egin					
Serial	seri	al 0 🔻	availab	le data ler	ngth	corio				boud	rata	1150	00 -	
Sensor	seri	al 0 🗸	read a	byte		Sella			egin	Jauui	ale	1152	00-•	

Set the pin IO39 connected to the water level sensor to "input" mode.

🥉 KidsBlo	ock Desktop 2.0.1	
kids	block 🌐 - Edit 🖀 kidslOT 🦎 Unconnected	8.1_
Coc	le ✔ Costumes 📢) Sounds	
Events	Events when Arduing begin	
Control	when Arduino begin serial 0 - begin baudrate 115200 -	
Operator	Control set pin IO39 ▼ mode input ▼	
Variables	wait 1 seconds	

Define a "Water\_level\_sensor" global variable to store the analog value of the water level sensor.

KidsBlock Desktop 2.0.1			
kidsblock + Edit #	kidslOT 🙀 Unconnected	8.1_Water-Level-Sensor	File
Code Costumes 🜗 Sounds			
Events			· · · · · · · ·
Control when Arduino begin	when Arduino begin		
	serial 0 ▼ begin baudrate 115200 ▼	]	
Control	set pin IO39 ▼ mode input ▼		
Variable: wait 1 seconds	Declare Global 🔻 variable Type int 🔻	Name Water_level	sensor Assigned to 0
My Blocks			

Store the read analog value of the water level sensor in the "Water\_level\_sensor" variable and print it on the serial port.

🍯 KidsBlo	ock Desktop 2.0.1														
kids	block 🕀	• Edit 📳	kidslOT	😽 Unconnected			8.1_Wa	iter-Leve	el-Senso		File				
Cod	le 🖌 Costumes	() Sounds													
Events	Events	· · ·													<b>D</b>
Control	when Arduino begin		when A	rduino begin											
		· ·	serial	0 🔻 begin	baudrate 1	15200 🔻									
Operator	Control		set pin	1039 🔻 m	node input	-									
Variables	wait 1 seconds		Declare		variable Tv	ne int 💌	Nam		/ater I	evel	sensoi	Δοσ	ianed	to	
My Blocks									rator_i	0101_	oonool		Ignou		
Pins	repeat 10		forever		- 10 - 10 - 10 - 10 - 10 - 10 - 10 - 10										
	3		Set	Water_level	_sensor va	ariable by	read	analo	g pin	1039	•				
Serial	forever		serial	0 👻 print	t variable	Water_le	vel_ser	nsor	wa	тр 💌					
Sensor	٦		wait	0.5 secon	ds										
Data	if then			¢.											
												_			

Complete Program

KidsBlock	Desktop 2.0.1													
kidsb	olock 🕀	Edit 📳	kidslOT	😽 Unconnected			8.1_Water-Le	vel-Sensor	File			C	0	🗘 Download firmw
Code	J Costumes	() Sounds											📩 Uplo	ad
Events E Control Operator	when Arduino begin		when Ard serial	luino begin	baudrate 115	; ;200 ₹							1 2 3 4 5 6 7 8 9	<pre>// generated #include <ard int="" pinmode(39,<="" pre="" serial.begi="" setup()="" void="" water_lev=""></ard></pre>
Variable: My Blocks	wait 1 seconds repeat 10		Declare forever	Global 👻	variable Type	int 💌	Name	Water_leve	el_senso	Assig	ned to	0 0 0	10 11 12 13 14 15 16	<pre>} void loop() {   Water_level   Serial.prin   delay(0.5 * }</pre>
Pins Serial Sensor Data	forever J		Set serial wait	Water_level 0   print 0.5 secon •	_sensor varia t variable V	able by ( Vater_lev	read anal	log pin IC	39 <b>-</b>			  	17	

### (2). Test Result

Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, click in the serial monitor and set the baud rate to 15200.

Insert the water level sensor in the water (or touch the sensor with wet hands), then the serial monitor will print the analog value of the sensor (range: 0~4095). The greater the water level, the greater the value!

2019 2166 2410 2707 2749	<b>D</b>
2817 2884 2928 2975 2997 3031 3045 3071	Buadrate 115200 V 2 End of line LF & CR V Hex form
3110 3149 3162	Auto scroll 🗹

# 6. Water Level Detection System



(1). Programming Steps

# **Step 1Flow Chart**



# Step 2Add "passive buzzer"



## Step 3Write the Program

Set the pin IO39 connected to the water level sensor to "input" mode.

KidsBlock Desktop 2.0.1											
kids	block 🛛 🕀 -	Edit 🔡 I	kidslOT 🎽 Unconnected	8.2							
Coc	de 🥜 Costumes	() Sounds									
events	Events										
Control	when Arduino begin		when Arduino begin								
Operator	Control		set pin IO39 👻 mode input 💌								
Variable	wait 1 seconds										

Initialize the width, height, I2C address, text size and color as well as background color of the OLED display.

👵 KidsBlo	ock Desktop 2.0.1													
kids	block	<b>.</b>	Edit	<b>kidslO</b> T	😽 Unc	connected				8.2_W	ater-Leve	el-Testing	J-S	F F
Cod	ie 🥜 Costu	umes	() Sounds											
Events	Events													
Control	when Arduino I	begin		wher	n Arduinc	begin								
Operator				set p	in IO3	9 🔻 ma	ode in	put 🔻						
	Control			init o	led width	128	height	64	I2C addi	ess	0x78 (	(0x3c)	•	- 1
My	wait 1 se	conds		set te	ext size	6x8 🔻	color	white	- bad	kgrou	ind col	or bl	ack •	
Blocks	40													

Define a "Water\_level\_sensor" global variable to store the analog value of the water level sensor.

👵 KidsBl	ock Desktop 2.0.1				
kids	block 🛛 🕀 -	Edit 🚦	kidslOT 🙀 Unconnected	8.2_Water-Level-Testing-S	File
Coo	de 🥒 Costumes	() Sounds			
Events	Events				· · · · · · ·
Control	when Arduino begin		when Arduino begin		
Operator			set pin IO39 ▼ mode input ▼		
	Control		init oled width 128 height 64 I2C	address 0x78 (0x3c) 🔻	
Variable	wait 1 seconds		set text size 6x8 ▼ color white ▼	background color blad	
My Blocks	repeat 10				
Pins			Declare Global  variable Type in	t  Name Water_leve	Sensor Assigned to 0

Store the read analog value of the sensor in the "Water\_level\_sensor" variable and display it on the OLED.

🥉 KidsBl	ock Desktop 2.0.	1													
kids	block		Edit	👫 kid	sIOT	🕌 Unconne	cted		8.2_Wate	r-Level-Te	sting	<b>•</b>	ile		
Co	de 🚽 Cos	stumes	📣 Sour	nds											
Events	Events														
Control	when Arduin	o begin			set pin	IO39 🔻 r	node input	•							
Operator	Control				init oleo	d width 128	height 64	I2C addre	ess 0x78	3 (0x3c)	•				
Variable	wait 1	seconds			set text	tsize 6x8 •	color wh	ite 🔻 bac	kground co	olor bla	ick 🔻				
My					Declare	e Global 💌	variable Typ	oe int <del>▼</del>	Name	Vater_lev	/el_senso	or As	signed	l to	0
Blocks	repeat 10				forever										
Pins		٦			Set	Water leve	sensor va	riable by	read analo	a nin	039 🗸				
Serial	forever				cloar					g pin []		•			
Sensor					Cieai	lieu									
Data					set c	ursor x: 0	y: 30								
	if	then			print	Soil_Level:	warp 🗸								
Variable Type					set c	ursor X: 80	<b>y</b> : 30								
TEXT	if 🔶	then			print	variable	Water_level_s	ensor v	varp 🔻						
	else				refre	sh oled									
					wait	0.3 seco	nds								
WIFI						و ر									

Determine the analog value of the sensor. If it is greater than 2000, the buzzer will sound an alarm; otherwise, the buzzer will not sound.



**Complete** Program



### (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, insert the water level sensor in the water (or touch the sensor with wet hands), when the sensor detects that the water level (or the moisture of your hands) is greater than the set threshold, the buzzer will sound an alarm.



### 7. Common Problems

#### Q: Is the sensor waterproof?

A: The detection area of the sensor is waterproof. Exceeding the detection area will cause a short circuit.

# 4.3.9 Project 09Automatic Irrigation System

Note: Do not allow water to overflow from sinks and soil troughs when using the device. Sprinkling water on other sensors will cause a short circuit and device failure. Sprinkling water on batteries will cause heating and explosion. Please be careful when using the device, especially when used by young children, it must be under the supervision of parents. To ensure safe operation of the device, please follow relevant usage guidelines and safety regulations.



#### 1. Description

This project introduces how to use a kidsIOT mainboard, a soil moisture sensor, a water level sensor, a passive buzzer, an OLED display, a relay and a water pump to build an automatic irrigation system.

We will read the analog values of the soil moisture sensor and water level sensor by writing code to control the relay and the water pump.

When the soil is too dry, the relay will be turned on to control the water pump to irrigate the plants, when too low, the water pump will not be started and the buzzer will alarm.

At the same time, the OLED display will display the dryness of the soil and the water level, thus realizing automated plant watering and water level control, improving production efficiency while reducing the time and energy costs of manual operations.

### 2. Components



### **About Relay Module**

**Relay Module:** It is an "automatic switch" module that uses small current to control large current, which is usually used in automated control circuits. Its three green binding posts NO, COM and NC are used for external circuits.

When the relay module is not connected to the control signal, COM and NC are connected and COM and NO are disconnected. During control, when the signal terminal is high level, the relay is closed, COM and NC are disconnected, and COM and NO are connected. When it is low level, the relay is disconnected, COM and NC are connected and COM and NO are disconnected.

### Parameters:

Working voltage: DC 3.3V-5V Working current: 125mA@3.3V, 75mA@5V Access capacity: 250VAC/3A, 30VDC/3A



### 3. Assembly Steps

It shares the same structural shape with **Project 07**. If the assembly parts of **Project 07** have finished, there is no need to assemble it again.

#### 4. Wiring Diagram

Module	GPIO Shield	kidsIOT Mainboard
Soil Moisture Sensor	$G {\rightarrow} GV {\rightarrow} VS {\rightarrow} S3$	No.4 portcontrol pin of S3 is io27
Water Level Sensor	$G \rightarrow GV \rightarrow VS \rightarrow S4$	No.4 portcontrol pin of S4 is io39
Passive Buzzer		No.8 portcontrol pin is io5
Relay Module		No.2 portcontrol pin is io2

Connect the kidsIOT mainboard to your computer via USB cable, connect the external power supply and turn the DIP switch on the mainboard to ON end.



5. Pumping System



## (1). Write the Program

Set the pin IO2 connected to the relay module to "**input**" mode.

🥉 KidsBlo	ck Desktop 2.0.1								
kids	block	⊕-	Edit		<b>kidslOT</b>	😽 Unconnect	ed	9.1 (	File
Code	e 🖌 Costu	mes	() Sour	ids					
Events	Events								
Control	when Arduino b	egin			when	Arduino begii	י 		
Operator	Control				set pi	n 102 🔻 r	node	output	

Set the relay module to close for 0.5 seconds and then disconnect, controlling the water pump to pump water for 0.5 seconds.

KidsBlock Desktop 2.0.1 kidsblock ⊕-💥 Unconnected File kidslOT Edit 🔚 Code 🖌 Costumes () Sounds Events Events Control output 🔻 102 🗸 mode set pin Operator Control set digital pin high 🔻 102 🔻 out Variable 1 0.5 wait seconds My Blocks repeat 10 set digital pin 102 🔻 out low 🔻 Pins

Complete Program

4.3. 3. Projects:

Nasblock Desktop 2.0.1				
kidsblock - Edit	📲 kidslOT 🛛 🦎 Unconnected	9.1_Water File	0	Download firmware
🛫 Code 🚽 Costumes 🌒 Sounds			🔔 Upload	
Events Events				<pre>// generated by KidsBlock #include <arduino.h></arduino.h></pre>
Control when Arduino begin	when Arduino begin	· · ·	- 5	<pre>void setup() {     pinMode(2, OUTPUT);</pre>
Operator	set pin IO2 ▼ mode	output 🔻	6 . 7 . 8	<pre>digitalWrite(2, HIGH); delay(0.5 * 1000); digitalWrite(2, LOW);</pre>
Variable	set digital pin IO2 💌	out high 👻	9 10	
wait 1 seconds	wait 0.5 seconds		11 V 12 ] 13	/old loop() {
Blocks repeat 10	set digital pin IO2 🔻	out low -		
Dinc				

### (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, the relay module will control the water pump to pump water.

#### 6. Automatic Irrigation System

Now we will use a kidsIOT mainboard, a soil moisture sensor, a water level sensor, a passive buzzer, an OLED display, a relay and a water pump to make an automatic irrigation system.
# (1). Programming Steps

## **Step 1Flow Chart**



### Step 2Add "passive buzzer"



#### Step 3Write the Program

Initialize pin IO27 of the soil moisture sensor and pin IO39 of the water level sensor to "input" mode.

🥉 KidsBlock D	esktop 2.0.1						
kidsbl	lock @-	Edit 🚦	kidslOT	Unconnecte	d		
Code	J Costumes	() Sounds					
Events Ev	rents	н н					
Control	hen Arduino begin		when A	Arduino begir			
Operator			set pin	IO27 <b>•</b>	mode	input •	
Variable	ontrol		set pin	IO39 🔻	mode	input •	
vanable.	ait <u>1</u> seconds						

Initialize pin IO2 of the relay module and pin IO5 of the passive buzzer to "Output" mode.



Initialize the width, height, I2C address, text size and color as well as background color of the OLED display.

KidsBlock Desktop 2.0.1		
kidsblock 🕀 Edit 🖀	kidslOT 😽 Unconnected	9.2_Auto-irrigation
🔚 Code 🖌 Costumes 🌒 Sounds		
Events Events		
Control when Arduino begin	when Arduino begin	
Dperato	set pin IO27 ▼ mode input ▼	
/ariables wait 1 seconds	set pin IO39 ▼ mode input ▼	
My	set pin IO2 ▼ mode output ▼	
Blocks repeat 10	set pin IO5 ▼ mode output ▼	
Pins	init oled width 128 height 64 I2C address 0x78 (0x	(3c) 👻
Serial forever	set text size 6x8 ▼ color white ▼ background color	black 🔻
Sensor		

Define a "Soil\_humidity\_sensor" global variable to store the analog value of the soil moisture sensor and a "Wa-ter\_level\_sensor" global variable to store the analog value of the water level sensor.



Assign sensor data to variables.



Display the sensor data at the corresponding position on the OLED display.



Judgment: When the water level is less than 1000, or the soil moisture is less than 500, the buzzer will sound an alarm.

if	$\overline{\langle}$	variable S	Soil_humi	dity_sensor	< 500	then
	, in the second	Tone PIN#	IO5 🔻	frequency	NOTE_C3 -	duration 131
	, I	Tone PIN#	IO5 🔻	frequency	NOTE_D3 -	duration 131
	1	Tone PIN#	IO5 🔻	frequency	NOTE_E3 -	duration 131
if		variable	Vater_lev	el_sensor	< 1000 t	hen
		Tone PIN#	IO5 🝷	frequency	NOTE_G5 🝷	duration 131
	F	Tone PIN#	IO5 🔻	frequency	NOTE_A5 🔻	duration 131
	, I	Tone PIN#	IO5 🔻	frequency	NOTE_B5 -	duration 131
		•				
		5				

When the soil moisture is less than 500 and the water level in the sink is greater than 1000, the relay will drive the water pump for automatic irrigation.

if variable Water_level_sensor	) >	1000	an	d	variable	Soil	_humidi	ty_ser	nsor	< 500	then
set digital pin IO2 ▼ out high ▼											
wait 0.5 seconds											
set digital pin IO2 ▼ out low ▼											
wait 0.7 seconds											
🗲 🖌 🕹 🕹											

Complete Program

when Arduino begin
set pin 1027 ▼ mode input ▼
set pin 1039 • mode input •
set pin IO2  mode output
set pin IO5 • mode output •
init oled width 128 height 64 I2C address 0x78 (0x3c) -
set text size 6x8 ▼ color white ▼ background color black ▼
Declare Global  variable Type int  Name Soil_humidity_sensor Assigned to 0
Declare Global  variable Type int  Name Water_level_sensor Assigned to 0
forever
Set Soil_humidity_sensor variable by read analog pin IO27 •
Set Water_level_sensor variable by read analog pin 1039 -
clear oled
set cursor x: 0 y: 20
print Soil_humidity: warp -
set cursor x: 90 y: 20
print variable Soil_humidity_sensor warp •
set cursor x: 0 y: 40
print Water_level: warp -
set cursor x: 80 y: 40
print variable Water_level_sensor warp -
refresh oled
wait 0.2 seconds
if variable Soil_humidity_sensor < 500 then
Tone PIN# IO5 • frequency NOTE_C3 • duration 131
Tone PIN# IO5 • frequency NOTE_D3 • duration 131
Tone PIN# IO5 • frequency NOTE_E3 • duration 131
i uniche Materiane anne 1000 there
if variable Water_level_sensor > 1000 and variable Soil_humidity_sensor < 500
set digital pin 102 • out high •
wait 0.5 seconds
set digital pin 102 v out 1ow v
wait 0.7 seconds

### (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard. After powering up via the USB cable, the OLED displays current soil moisture and sink water level information.

When the soil moisture is lower than the set threshold, then the soil is too dry, the buzzer will sound an alarm. At this time, the pumping system will automatically irrigate the soil. When the water level in the sink is lower than the set threshold, the pumping system will not work and the buzzer will sound an alarm to indicate that there is insufficient water in the sink.



#### 7. Common Problems

#### Q1: Is the sensor waterproof?

A: The relay is not waterproof.

#### Q2: Does the rotation of the water pump cause the kidsIOT board to reset?

A: When the water pump rotates, it requires a larger current than other sensors, which will cause voltage and current fluctuations in the circuit. Excessive voltage and current fluctuations will cause the voltage and current of the kidsIOT mainboard to be too low, causing the kidsIOT mainboard to reset.

# 4.3.10 Project 10 WiFi Web Page Controls Smart Farm System

Note: Do not allow water to overflow from sinks and soil troughs when using the device. Sprinkling water on other sensors will cause a short circuit and device failure. Sprinkling water on batteries will cause heating and explosion. Please be careful when using the device, especially when used by young children, it must be under the supervision of parents. To ensure safe operation of the device, please follow relevant usage guidelines and safety regulations.



#### 1. Description

In today's era of rapid technological development, unified control of intelligent devices via mobile phones has gradually gained people's favor. This method uses a microcontroller to establish a connection between a mobile phone and an intelligent device through a WiFi module and the Internet network to achieve remote control of the intelligent device.

In this project, we will focus on the WiFi infrastructure of ESP32 and control the smart farm system via WiFi web page.

## 2. Components



# 3. Assembly Steps

Just put the structural shapes assembled in Project 02, Project 06 and Project 07 together .



# 4. Wiring Diagram

Module	GPIO Shield	kidsIOT Mainboard
Water Level Sensor	$G \rightarrow GV \rightarrow VS \rightarrow S4$	No.4 portcontrol pin of S4 is io39
Soil Moisture Sensor		G/V/io33 portG $\rightarrow$ GV $\rightarrow$ VS $\rightarrow$ io33
Photoresistor		No.6 portcontrol pin is io36
Temperature and Humidity Sensor		No.8 portcontrol pin is io5
White LED Module		No.7 portcontrol pin is io16
Passive Buzzer		No.2 portcontrol pin is io2
Motor		No.9 portIN+control pin is io18IN-control pin is io19
Relay Module		No.3 portcontrol pin is io26

Connect the kidsIOT mainboard to your computer via USB cable, connect the external power supply and turn the DIP switch on the mainboard to ON end.



5. WiFi Web Page Display



The ESP32 chip on the kidsIOT board comes with Wi-Fi (2.4GHz) and Bluetooth (4.2) functions. It can easily connect to the Wi-Fi network and communicate with other devices on the network. You can use ESP32 to display web pages in the browser.

# (1). Knowledge



The Wi-Fi library supports configuring and monitoring ESP32 Wi-Fi networking functions. Supported configurations:

- station mode (STA mode or Wi-Fi client mode), at this time the ESP32 is connected to the access point (AP).
- AP mode (Soft-AP mode or Access Point mode), at this time the base station is connected to the ESP32.
- station/AP coexistence mode (ESP32 is both an access point and a base station connected to another access point).
- Various security modes for the above mentioned modes (WPA, WPA2, WPA3, etc.).
- Scan access points (including active scanning and passive scanning).
- Monitor IEEE802.11 Wi-Fi packets using promiscuous mode.

For details about wifi, tap ithttps://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp\_wifi.html

Espressif official websitehttps://www.espressif.com.cn/en/home

# (2). Programming Steps

## Step 1Add the "Web Page Editing PRO" library



to return to the programming interface.



KidsBlock Desktop 2.0.1



👵 KidsBlo	ck Desktop 2.0.1	
kids	block 🕀 - Edit 🖀 kidsIOT 🦎 Unconnected	9.2_Auto-irrigation
Co	de 🖌 Costumes 📢 Sounds	
_ Events	Web Editor PRO	
Control	Update card label temperature card unit °C card type temperature • card ID 1 • value 20	
Operator	Set state card label system status card icon success - card ID 1 - value normal	
Variable:	Update chart header Temperature curve card type BAR   CHART card ID 1   data origin mylistx data	a origin Y mylisty
My Blocks	Update card label Turn on the light in the bedroom card type button - card ID 1 - value 0	
Pins	Get card value card label Turn on the light in the bedroom card type button - card ID 1 - return value	
Serial		
Sensor	Get joystick value labe joystick card type bothway  card ID 1  return x,y	
o Data		
Variable	Set webpage title is keyes DIY robot	
	Set username keyestudio password 123456	
TEXT	Add page bedroom page ID 1	
	Add statistics label author value keyes DIY robot ID 1	
Passive	Set card page type temperature  card ID 1  page ID 1	
Web Editor		

# Step 2Description of the Building Block



Enter the WiFi name and password to connect to the WiFi hotspot.



Read the WiFi IP address.

) U	pdate	card	label	tem	perati	ure	card (	unit (	°C	card ty	уре	temperature 👻	card ID	1 🕶	value	20
											1	temperature				
												humidity				
												air pressure				
												energy				
												progress bar				
												tags				

Set up a card on the web page, and its card label, card unit and card type correspond one to one.

Get card value card label	Turn on the light in the bedroom	card type button 👻 card ID	1 - return value
		✓ button	
		TEXT INPUT CARD	

Add a button card to the web page.

#### Step 3Write the Program

Change the WiFi name and password in the code to your own WiFi name and password, and connect to the WiFi hotspot.



Display the WiFi IP address on the OLED.

🧯 KidsBlock 🛛	Desktop 2.0.1									
kidsbl	lock 🗣	Edit 📒 kia	islOT 🛛 🙀 Un	connected				10.1_Wi	Fi_Html_Te	st
Code	Costumes	) Sounds								
Events Ev	vents		when Arduino	begin						
Control	hen Arduino begin		🛜 wifi co	onnect ssid:	ChinaNet	_2.4G	password:	China	aNet@23	33
Operator Co	ontrol		init oled width	128 heig	ght 64 I	2C addre	ess 0x78	(0x3c)	-	
	ait 1 seconds		set text size	12x16 🔻	color whi	te 🔻 b	ackground	color	black 🔻	
Blocks	epeat 10		set cursor x:	0 y: 0						
Serial	ۍ		clear oled							
Sensor	Jever J		print 🛜	WiFi read i	p warp	•				
Data	then		refresh oled							

Set a card on the web page, the card label is "Temperature", the card type is "Temperature", the card unit is "°C", the card ID is 1, and the temperature value is 30.

👵 KidsBlo	ock Desktop 2.0.1												
kids	<mark>block</mark> @	🕶 Edit 📲	kidslOT 🛛 🙀 Unc	onnected			10.1_WiFi	_Html_Test		File			
Cod	le 🦪 Costumes	() Sounds											
Events	Events		when Arduino	begin									
Control	when Arduino begin		🛜 wifi co	nnect ssid: Chi	naNet_2.4G	password	: Chinal	let@233					
Operator	Control		init oled width	128 height	64 I2C add	Iress 0x7	8 (0x3c) 🗸						
Variable:	wait 1 seconds		set text size	12x16 🔻 color	white 🔻	background	d color b	lack 🔻					
Blocks	repeat 10		set cursor x:	0 y: 0									
Serial	January Ja		clear oled										
Sensor	J		print 🛜	WiFi read ip	warp 🔻								
Data	if then		refresh oled								,		
Variable				card label Tem	perature ca			pe tempe	erature			aiue	30



Set up a card on the web page. The card label is "Air humidity", the card type is "Humidity", the card unit is "%RH", the card ID is 2, and the temperature value is 60.



Complete Program

KidsBlock Desktop 2.0.1						
kidsblock 🕀 Edit 🖀	kidsIOT 🦎 Unconnected		File		Ō	🔅 Download firmware 🔅 Tutorials 🛛
🖙 Code 🚽 Costumes 🌒 Sounds					🛓 Up	load
Events	when Arduino begin					// generated by KidsBlock #include <arduino.h> #include <wifi.h></wifi.h></arduino.h>
Control when Arduino begin	wifi connect ssid: ChinaNet_2.4G	password: ChinaNet@233			5	<pre>#include <wire.h> #include <adafruit_gfx.h> #include <adafruit_ssd1306.h> #include <webpro.h></webpro.h></adafruit_ssd1306.h></adafruit_gfx.h></wire.h></pre>
Control Variable: wait 1 seconds	init oled width 128 height 64 I2C ad	ldress 0x78 (0x3c) ▼			9 10 11	<pre>const char* ssid = "ChinaNet_2.4G"; const char* password = "ChinaNet@233";</pre>
My Blocks repeat 10	set text size 12x16 • color White • set cursor x: 0 y: 0	background color black -			13 14 15 16	Adafruit_SSD1306 oled(128, 64, &Wire); AsyncWebServer server(80); ESPDash dashboard(&server, 1);
Pins Serial Former	clear oled					
Sensor	print WiFi read ip warp -					
Data Variable	Update card label Temperature	card unit ° card type temper	ature 🔻 card II	D 1 - value 30		
Type if then	Update card label Air humidity ca	ard unit %RH card type humi	dity 🔻 card ID	2 • value 60		

# (3). Test Result

🏦 Upload

Click to upload the above complete code to the kidsIOT mainboard and power on. Once connected to WiFi, the OLED on the board will display the corresponding IP address (Here we take the IP address: 192.168.0.48 as an example ).



Note: When the PC, mobile phone and kidsIOT mainboard are connected to the same network, this webpage can be opened on the PC and mobile phone at the same time. Here is the IP address displayed on the OLED on your own kidsIOT board

You can enter "http://[IP address displayed on the OLED display]" in the browser to view the web page. In this way, you will create a simple web page that displays a fixed temperature information and a fixed humidity information:

PC					
单 🛛 keyes	× +		$\sim$	_	
$\leftarrow \rightarrow $ C	ධ 🔿 👌 192.168.0.48	器 ☆ へ	lii\ <b>t</b>	6 4	മ ≡
					^
Logo	Temperature <b>30</b> °C	Air humidity 60 %RH			
•					~

Mobile phone





### 6. WiFi Web Page Controls Smart Farm System



#### (1). Programming Steps

### **Step 1Flow Chart**



## Step 2Add "Passive Buzzer", "DC Motor", "Temperature and Humidity Sensor" and "Web Page Editing PRO" libraries

Tap , click "Actuator" and find "esp32 Passive buzzer" and "DC Motor for esp32". Click "Sensor" and find "DHT sensor for ESP32".

Click the "communication", then select "Web Page Editing PRO" and click to return to the programming interface.



Help

Help

Help



🕨 KidsBlo	ck Desktop 2.0.1		
kids	black 🕀 - Edit 🖀 kidslOT	😽 Unconnected	KidsBlock Project
Se Co	ie 🖌 Costumes 🜗 Sounds		
ariables	DC Motor		
My	S Motor INA# IO2   State HIGH   INB#	IO6 ▼ State HIGH ▼	
Blocks	S Motor INA# IO2  → State HIGH  → INB#	IO6 ▼ channle CH0 (LT0) ▼ analogWrite 4096	
Pins	Passive buzzer	· · · · · · · · · · · · · · · · · · ·	
Serial	Tone PIN# 1033 - frequency NOTE_C3 -	duration 131	
Sensor	🎁 Tone PIN# 1033 🔹 play music Birthday 🔹		
Data	noTone IO33 -		
'ariable Type	Temperature and humidity		
TEXT	DHT pin IO26 - mode dht11 -		
OLED	DHT get temperature -		
WIFI	Web Editor PRO		
DC	Update card label temperature card unit °C	card type temperature  card ID 1  value 20	
Motor	Set state card label system status card icon	success   card ID 1  value normal	· · · · · · · · ·
assive	Update chart header Temperature curve card	type BAR   CHART card ID 1  data origin mylistx data origin	gin Y mylisty
empera and umidity	Update card label Turn on the "ght in the bedroo	om card type button  card ID 1  value 0	
Web Editor PRO	Get card value card label Turn on the light in the	e bedroom card type button   card ID 1  return value	
<b>=</b> *	Get joystick value labe joystick card type bo	thway ▼ card ID 1 ▼ return x,y	

## Step 3Write the Program

Change the WiFi name and password in the code to your own WiFi name and password, and connect to the WiFi hotspot.

	1 - 1											
when Ard	uino begin							-				
ি শ	vifi connect	ssid:	Chi	naNe	t_2.40	g p	asswo	ord:	China	aNet@	D233	

Display the WiFi IP address on the OLED.

init oled width	128 he	eight	64 <b>I2C a</b> d	ldress	0x7	′8 (0x3o	c) 🔹	
set text size	12x16 🔻	color	white 🔻	back	groun	d color	black	•
set cursor x:	0 y:	0						
clear oled								
print 🛜	WiFi read	l ip	warp 👻					
refresh oled								

Initialize pin IO5 of the temperature and humidity sensor, and select the mode dht11.

E.	DHT pin	IO5 🔻	mode	dht11 🔻
			/	

Initialize the pin IO33 of the soil moisture sensor, the pin IO39 of the water level sensor, and the pin IO36 of the photoresistor, and set them to "**Input**" mode.



Initialize the pin IO16 of the LED module, the pin IO26 of the relay module and the pin IO2 of the passive buzzer, and set them to "**Output**" mode.



Define four global variables, namely "Val1", "Val2", "Val3" and "Val4".

Declare	Global 🔻	variable Type	int 💌	Name	Val1	Assigned to	0
Declare	Global 🔻	variable Type	int 💌	Name	Val2	Assigned to	0
Declare	Global 🔻	variable Type	int 💌	Name	Val3	Assigned to	0
Declare	Global 🔻	variable Type	int 💌	Name	Val4	Assigned to	0

Set up multiple cards of the web page, namely TemperatureAir humiditySoil humidityWater level and Brightness.

forever	a a a									
	Update card label	Temperature	card unit	℃ card type	temperatur	re 🔹 card I	D 1 🔻	value 👔	DHT get	temperature 👻
	Update card label	Air humidity	card unit	6RH card ty	pe humidity	card ID	2 🔹	value	DHT get hu	umidity 🔻
	Update card label	Soil humidity	card unit	card type	e tags 🔹 d	card ID 3 •	value	read analog pi	n IO33 🕶	
	Update card label	Water level	card unit	card type	tags 🔹 ca	rd ID 4 💌	value	read analog pin	IO39 -	
	Update card label	Brightness	card unit	card type	tags 🔹 car	rd ID 5 💌	value	read analog pin	IO36 🗸	
wait	1 seconds									
	ال									

Set the button card to control LED on and off.

Get card value card label LED card ty	/pe (	butto	n 🔻	card II	D	6 🔻	retur	n valu
if variable Val1 = 0 then								
Set Val1 variable by 1								
set digital pin IO16 ▼ out high ▼								
Set Val1 variable by 0								
set digital pin IO16 ▼ out Iow ▼								

Set the button card to control the fan on and off.

Get card value car	rd label Fan	card type	button 👻	card ID	7 • r	eturn value		
if variable V	(al2) = 0	then						
Set Val2 variab	le by 1							
Sc Motor INA#	IO18 🔹 St	ate LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	150
else Set Val2 variab	le by 0							
Motor INA#	IO18 - St	ate LOW 🔻	INB#	IO19 🔻	channle	CH0 (LT0) 🔻	analogWrite	0

Set a button card to control the relay's on and off, thereby controlling whether the water pump swater.

Get card value card label Watering	card type	button -	card ID	8 🗸	return value
f variable Val3 = 0 th	nen				
set digital pin IO26 ▼ out high ▼					
wait 0.5 seconds					
set digital pin IO26 ▼ out low ▼					
wait 0.7 seconds					

Set the button card for playing music.

Get card value card label	Music card	type	butto	n 🔻	d ID	9 🗸	ret	urn va	lue
f variable Val4 =	0 the	1							
Tone PIN# IO2 -	play music	Birth	day 🔻						

Complete Program

when Arduino begin
wifi connect ssid: ChinaNet_2.4G password: ChinaNet@233
init oled width 128 height 64 12C address 0x78 (0x3c) -
set text size 12x16 - color white - background color black -
set cursor x: 0 y: 0
clear oled
print 🛜 WiFi read ip warp 🔹
refresh oled
DHT pin 105 • mode dhi11 •
set pin 1033 • mode input •
set pin 1039 • mode input •
setnin 1038 * mode innut *
and part (1990 - model might -
ser prin Tozo V mode bulput V
set pin IO2 V mode output V
Declare Global • variable Type int • Name Val1 Assigned to 0
Declare Global • variable Type int • Name Val2 Assigned to 0
Declare Global  variable Type int  Name Val3 Assigned to 0
Declare Global • variable Type int • Name Val4 Assigned to 0
forever
🖳 Update card label 🚺 Temperature card unit 🐨 card type temperature 🔹 card ID 1 🔹 value 🕼 DHT get temperature
Update card label 🗛 humidity card unit 🛞 RH card type humidity 🔹 card ID 2 🔹 value 🕕 DHT get humidity 🔹
Update card label Soil humidity card unit card type tags • card ID 3 • value read analog pin 1033 •
Update card label Water level card unit card type tags - card ID 4 - value read analog pin 1039 -
The line and label Brightness card unit card type target card ID 5 x value read analog pin 1036 x
Cet card value card label     ED     card type     button •     card ID     6 •     return value       f     variable     Vall     =     0     then       Set     Vall     variable     0     then       set digital pin     1016 •     out     hph •       set     Vall     variable     0       set     tigital pin     1016 •     out       set     tigital pin     1016 •     out
Cet card value card label Fan card type button • card ID 7 • return value variable Val2 = 0 then Set Val2 variable by 1 Set Val2 variable by 0 Set
U     Certe tails     Value     Certe type     button     Certe type     button     Certe type       If     Certe type     Dutton     Certe type     button     Certe type     button       is     et algolate type     Dutton     Certe type     button     Certe type     button       is     et algolate type     Dutton     Certe type     button     Certe type     button       is     et algolate type     Dutton     Certe type     button     Certe type     button       is     et algolate type     Dutton     et algolate type     certe type     button     et algolate type       is     et algolate type     Dutton     et algolate type     certe type     certe type       is     et algolate type     Dutton     et algolate type     tope     certe type       is     et algolate type     tope     out     tope     et algolate type
Get card value card label Music card type button • card ID 9 • return value variable Val4 • 0 then Tone PINM IO2 • play music Birthday •

# (2). Test Result



Click to upload the above complete code to the kidsIOT mainboard and power on. Once connected to WiFi, the OLED on the board will display the corresponding IP address. You can enter "http://[IP address displayed on the OLED]" in the browser to view the web page . The sensor data can be viewed on the web page, and LEDs and fans can also be controlled.



PC



Mobile phone

	Temperature 27 °C	
•	Air humidity 58 %RH	
•	Soil humidity 1252	
	Water level 1493	
ite	Brightness 1697	
LED		×
Fan		×
Wateri	ing	×
Music		×
	L1 keyes	88 🖸

=

Sensor data	Control	
Ambient temperature(°C)	LED on/off	
Ambient humidity(%RH)	Fan on/off	
Soil moisture	Water pump on/off	
water level of sink	Play music	
Light(0~4095)		

#### 7. Common Problems

#### Q1You can't connect the WiFi?

A: Please move the kidsIOT board to the vicinity of the router, press the reset button on the board to restart kidsIOT, and wait patiently for the connection. If you still can't connect, please check whether the WiFi name and password are filled in correctly.

#### Q2When remotely operating other sensors on the web page, the response is very slow?

A: Reasons:

- When multiple people are connecting it, the router's CPU resources are insufficient. Restart the router and reconnect to the network.
- The router system has been used for too long. Restart the router.
- When wireless interference occurs and the wireless signal is unstable, please do not use it through the wall.

For router related knowledge, please search on google yourself.