

uArm Controller

User Manual









V1.0.1 2018-12-28

Table of Contents

Product Introduction — — — — — — — — — 01
Hardware Parameters — — — — — — — 02
Hardware Installation — — — — — — — — 05
Preset Demo Tutorial — — — — — — — — 06
Software Installation — — — — — — — — 06
Github Demo 08
Release Note - - - - 19
\otimes ζ $\dot{\lambda}$
a the mars of
\sim

Product Introduction

uArm Controller is an open-source hardware based on Arduino MEGA 2560, and has a wide range of extendable functions. It's compatible with various peripherals and great for implementing your creative ideas.





Hardware Parameters

Specification				
Weight	0.15kg			
Dimension(L*W*H)	150mm*132mm*281mm			
Connection with PC	Micro USB			
Input Voltage	USB 5V			
Main Controller	ATMEGA2560 (Arduino compatible)			
Display	128x64 OLED			
Buttons	4			
Rotary Potentionmeter	1			
ТуреС	4 (only for uarm connection)			
RGB LED	1			
Micro SD	1			
4P Connector	2xDigital IOs / 2xIIC			
Extendable I/O	6xdigital IOs			
Operation Temperature & Humidity	0°C-35°C 30%RH-80%RH noncondensing			
Storage Temperature & Humidity	-20°C-60°C 30%RH-80%RH noncondensing			





128X64 Display Pin Description

LCD_CS	D 4 2
LCD_RES	D 4 3
LCD_CD	D 4 4
LCD_SCK	D 3 0
LCD_MOSI	D 3 5

8 Micro SD Card Pin Description

TF_CS	 D13
TF_SCK	 SCK(D53)
TF_MOSI	 MOSI(D51)
TF_MISO	 MISO(D50)

5 Joystick Pin Description

JOY_UP		D33
JOY_DOWN		D41
JOY_LEFT		D32
JOY_RIGHT		D37
JOY_CENTRE		D40
(Press Vil, Loose	n Vih)	

Buzzer Pin Description

BUZZER D45

(Hardware PWM Control)

2 BUTTON Pin Description

BUTTON_A		D47		
BUTTON_B		D49		
BUTTON_C		D12		
BUTTON_D		D11		
(Press Vil, Loosen Vih)				

4 RGB LED Pin Description

RGB_LED_R	····· D 2
RGB_LED_G	D 3
RGB_LED_B	····· D 5
(Vil On, Vih Off, O	Controlled by Hardware PWM)

6 IIC Device Address

EEPROM 24C128 Address	- D13
ADXL345 Address SCK	(D53)

8 Rotary Potentiometer Pin Description

ROTARY		A15
(Measurem	ent Analog)	

• Peripheral Port Pin Description (1.27mm 4P)

Port	1	2	3	4
Port1	SCL	SDA	5V	GND
Port 2	SCL	SDA	5V	GND
Port 3	D25	D24	5V	GND
Port 4	D23	D22	5V	GND



• IO Port Pin Description

1: 12V (NC)	2: GND	3: TXD0	4: D4	5: D8	6: D53
7: 5V	8: 3.3V	9: RXD0	10: D6	11: D9	12: D10



• TypeC Pin Description

Note: the four-core TypeC cable in the market is not supported for this port, please link it to the test point in the back.



Hardware Installation



Connect the uArm Controller to your computer using the Micro USB cable.
Power on the uArm Controller (push up).

(For the Peripheral Port, a 1.27mm 4P cable should be purchased separately)

Preset Demo Tutorial

- Press the "Reset" button, reset the system.
- Use the Joystick to control the movement (e.g. move around, up and down) when playing the snake game.



∞ sketch_dec26	ia Arduino 1.8.8 — 🗆 🗙	
File Edit Sketch	Tools Help	
	Auto Format	Ctrl+T
	Archive Sketch	
sketch_dec26a	Fix Encoding & Reload	
<pre>void setup() {</pre>	Manage Libraries	Ctrl+Shift+I
// put your se	Serial Monitor	Ctrl+Shift+M
1	Serial Plotter	Ctrl+Shift+L
	WiFi101 / WiFiNINA Firmware Updater	
void loop() {	Board: "Arduino/Genuino Mega or Mega 2560"	, ,
,,, p j	Processor: "ATmega2560 (Mega 2560)"	>
}	Port: "COM11 (Arduino/Genuino Mega or Mega 2560)"	>
	Get Board Info	-
	Programmer: "AVRISP mkII"	>
	Burn Bootloader	

Software Installation

Download the Arduino IDE (www.arduino.cc)
GitHub: https://github.com/uArm-Developer/Controller
Arduino Port Reference: https://www.arduino.cc/reference/en/
Development Language: C/C++

Note: When uArm Controller is connected with a computer, please press the "Reset" button.

• When configure the Arduino IDE Hardware Platform and the COM Port (the COM Port is allocated randomly by your computer), please use IDE for code writing or use Github routine code to develop.

• As shown in the following picture, please press the "Upload" button to upload firmware.



Github Demo

Branch: master 🕶	New pull request		Create new file Upload files Find file Clone or down					
🤨 TopgunZh Ado	2018-12-25	5 Latest commit 8963a4b Dec 2						
doc		Add 2018-12-25	Dec 25, 2					
driver		Update 2018-12-25				Dec 25, 2018		
im image		Update 2018-12-25	Dec 25, 2					
scene_demo		Update 2018-12-25	Dec 25, 2					
in sch		Add 2018-12-25	Dec 25,					
README.md		update	Nov 9					

doc→ User Manual

driver→ Peripheral Routine image→ Product Image

scene demo→ Scene Demo

sch→ Schematic Diagram of uArm Controller

• Peripheral driver

Branch: master - Controller / driver /		Create new file	Uploa
🚺 TopgunZh Update 2018-12-25			Latest
button_driver	Update 2018-12-25		
joystick_driver	Update 2018-12-25		
led_RGB_diver	Update 2018-12-05		
led_12864_driver	Update 2018-12-05		

At the moment, there are four routines under the "driver" folder (we will keep updating the routine, please check our Github content).

button_driver

For hardware connection, please refer to (Hardware Parameters→ Button Pin Description)

1.Download Project File: button_driver.ino

(https://github.com/uArm-Developer/Controller/tree/master/driver/button_driver)

About how to download a single file from Github, please refer to

(https://github.com/uArm-Developer/SwiftProForArduino/

wiki/How-to-download-single-file-from-GitHub)

2.Configure Arduino IDE, select the board as "Arduino/Genuino Mega or Mega 2560", Select the COM that allocated randomly by your computer, please refer to the following picture:

00	button_driver	Arduino 1.8.7		_	×
File	Edit Sketch T	ools Help			
		Auto Format	Ctrl+T		Ø
		Archive Sketch			
k	utton_driver	Fix Encoding & Reload			
1	/*****	Manage Libraries	Ctrl+Shift+I		^
2	#define BUTT	Serial Monitor	Ctrl+Shift+M		
3	#define BUTT	Serial Plotter	Ctrl+Shift+L		
4	#define BUTT				
5	#define BUTT	WiFi101 Firmware Updater			
7	word setup()	Board: "Arduino/Genuino Mega or Mega 2560"	>		
8	// put you	Processor: "ATmega2560 (Mega 2560)"	>		
9	Serial be	Port: "COM95 (Arduino/Genuino Mega or Mega 2560)"	>		
10	Serial.pr	Get Board Info		•	
11	pinMode(BU				
12	pinMode(BU	Programmer: "Arduino as ISP"	>		
13	pinMode(BU	Burn Bootloader			
14	pinMode(BUTT	ON_D, INPUT_PULLUP);			
15	۲ ا				
10	void loon() {				
18	// put your	main code here, to run repeatedly:			
19	if(digitalRe	ad(BUTTON_A) = LOW) {			

Note: When uArm Controller is connected with a computer, please press the "Reset" button.

3.Press the "Upload" button to upload firmware.

💿 button_driver Arduino 1.8.8 —	×
File_Edit_Sketch Tools Help	
	ø
button_driver	
/*************************************	^
#define BUTTON_A 47	
#define BUTTON_B 49	
#1.f: DIDTON 0 10	

4. Press the "Serial Monitor" button to open Port Assistant.

⊚ joystick_driver Arduino 1.8.8	_		×
<u>F</u> ile <u>E</u> dit <u>S</u> ketch <u>T</u> ools <u>H</u> elp			
	Seria	al Monitor	P
joystick_driver			

5. Press button A, B, C, D to check the log of terminal printing.

💿 COM6 (Arduino/Genuino Mega or Mega 2560)	_	\times
		Send
utton test		
utton c		
utton a		
utton b		
untton d		

Autoscroll 🗌 Show timestamp	Newline N	115200 baud $$	Clear output

joystick_driver

For hardware connection, please refer to (Hardware Parameters→ Joystick Pin Description).

1. Download Project File: joystick_driver.ino

(https://github.com/uArm-Developer/Controller/tree/master/driver/joystick_driver) About how to download a single file from Github, please refer to: (https://github.com/uArm-Developer/SwiftProForArduino/ wiki/How-to-download-single-file-from-GitHub) 2.Configure Arduino IDE, select the board as "Arduino/Genuino Mega or Mega 2560", Select the COM that allocated randomly by your computer, please refer to the following picture:

⊚ joystick_drive	r Arduino 1.8.8 — 🗆 🖸	×	
<u>F</u> ile <u>E</u> dit <u>S</u> ketch	<u>T</u> ools <u>H</u> elp		
	Auto Format Archive Sketch	Ctrl+T	
joystick_driver	Fix Encoding & Reload		
/10101010101010101010101010101010101010	Manage Libraries	Ctrl+Shift+I	1
#define JOY_UP	Serial Monitor	Ctrl+Shift+I	м
#define JOY_DOWN #define JOY_LEFT	Serial Plotter	Ctrl+Shift+I	L
#define JOY_RIGH	WiFi101 / WiFiNINA Firmware Updater		
#define JU1_CENI.	Board: "Arduino/Genuino Mega or Mega 2560"		>
<pre>void setup() {</pre>	Processor: "ATmega2560 (Mega 2560)"		>
// put your se	Port: "COM6 (Arduino/Genuino Mega or Mega 2560)"		>
Serial begin(Get Board Info		J

Note: When uArm Controller is connected with a computer, please press the "Reset" button.

3.Press the "Upload" button to upload firmware.

💿 joystick_driver Arduino 1.8.8	_	×
<u>File Edit Sketch Tools H</u> elp		
		ø
joystick_driver		

4. Press the "Serial Monitor" button to open Port Assistant.



5. Push the Joystick to check the log of terminal printing.

joystick testjoystick test	
joystick up	
joystick down	
joystick left	
joystick right	
joystick center	
	_
🗹 Autoscroll 🗌 Show timestamp Newline 🗸 115200 baud	✓ Clear

• led_RGB_driver

For hardware connection, please refer to (Hardware Parameters→ RGB led Pin Description)

1.Download Project File: led_RGB_diver.ino

https://github.com/uArm-Developer/Controller/tree/master/driver/led_RGB_diver About how to download a single file from Github, please refer to:

(https://github.com/uArm-Developer/SwiftProForArduino/

wiki/How-to-download-single-file-from-GitHub)

2.Configure Arduino IDE, select the board as "Arduino/Genuino Mega or Mega 2560", Select the COM that allocated randomly by your computer, please refer to the following picture:

	Auto Format	Ctrl+T	1	Ø
	Archive Sketch			Name of Street o
led_RGB_diver	Fix Encoding & Reload			
1 /*******	Manage Libraries	Ctrl+Shift+I	-	
2 #define RGB_	Serial Monitor	Ctrl+Shift+M		
3 #define RGB_ 4 #define RGB	Serial Plotter	Ctrl+Shift+L		
5	WiFi101 Firmware Updater			
6 enum rgb_mod	Board: "Arduino/Genuino Mega or Mega 2560"	>		
GREEN.	Processor: "ATmega2560 (Mega 2560)"	>		
BLUE,	Port: "COM95 (Arduino/Genuino Mega or Mega 2560)"	>		
} rgb_mode =	Get Board Info		<u>.</u>	
2 void setup()	Programmer: "Arduino as ISP"	>		
3 // put you	Burn Bootloader			
4 pinMode(RGB_)	LED_R, OUTPUT);		ŝ	
5 pinMode(RGB_)	LED_G, OUTPUT);			
6 pinMode(RGE_	LED_B, OUTPUT);			
7 }				
8				
9 void loop() {				

Note:When uArm Controller is connected with a computer, please press the "Reset" button.

3.Press the "Upload" button to upload firmware.

💿 led_RGB_diver Arduino 1.8.7	<u></u>	×
<u>File Edit Sketch Tools H</u> elp		
🕑 📀 💽 🔛 Upload		ø
led_RGB_diver		

4.Test, and the RGB light will alternates with red, green and blue lights.

oled_12864_driver

For hardware connection, please refer to (Hardware Parameters→ 128x64 Display Pin Description).

1.Download Project File: oled_12864_driver.ino (https://github.com/uArm-Developer/Controller/tree/master/driver/oled_12864_driver) About how to download a single file from Github, please refer to: (https://github.com/uArm-Developer/SwiftProForArduino/

wiki/How-to-download-single-file-from-GitHub)

2.Configure Arduino IDE, select the board as "Arduino/Genuino Mega or Mega 2560", Select the COM that allocated randomly by your computer, please refer to the following picture:



Note:When uArm Controller is connected with a computer, please press the "Reset" button.

3. Open the "Manage Libraries" .

💿 oled_12864_driver Arduino 1.8.8				_		\times		
File Edit Sketch Tools Help								
00	Verify/Compile	Ctrl+R				Ø		
	Upload	Ctrl+U				-		
oled_1:	Upload Using Programmer	Ctrl+Shift+U						
#include	Export compiled Binary	Ctrl+Alt+S				^		
/>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	Show Skatch Folder	Ctrl . K	· olok/			_		
#define 1	Show Skellin Policer	CUITEN		Man	age Li	braries	Ctrl+Shift+I	
#define 1	Include Library	;	L				Carrometri	
#define 1	Add File			Add	.ZIP Li	brary		
#define LCD_SCK 30								
#define LCD_MOSI 35				Arduino libraries				
				Brid	ge			
USGLIB_SH1106_128X64 u8g(LCD_SCK, LCD_MOSI, LCD_CS, LCD_				EEPROM				
				Esplora				
// put your setup code here, to run once:				Ethernet				
pinMode (LCD_RES, OUTPUT);				Firmata				
digitalWrite(LCD_RES, HIGH);				GSM				
u8g.firstPage();				HID				
		1						

4.Install u8glib.

30 Library Manager	×
Type All v Topic All v USglib	
<u>More info</u>	
U8g2 by oliver Monochrome LCD, OLED and eInk Library. Display controller: SSD1305, SSD1306, SSD1309, SSD1322, SSD1325, SSD1327, SSD1329, SSD1600, SSD1607, SH1106, SH1107, SH1108, SH1122, T6963, RA8835, LC7981, PCR8812, HX1230, UC1601, UC1604, UC1608, UC1610, UC1611, UC1701, ST7565, ST7567, ST7588, ST75256, NT7534, IST3020, ST7920, LD7032, KS0108, SED1520, SBN1661, IL3820, MAX7219. Interfaces: I2C, SPI, Parallel. Monochrome LCD, OLED and eInk Library. Successor of U8glib. Supported display controller: SSD1305, SSD1306, SSD1309, SSD1322, SSD1327, SSD1327, SSD1606, SSD1607, SH1106, SH1107, SH1108, SH1122, T6963, RA8835, LC7981, PCD8544, PCF8812, HX1230, UC1601, UC1608, UC16108, UC1610, UC1611, UC1701, ST7565, ST7567, ST7588, ST75256, NT7534, IST3020, ST7920, LD7032, KS0108, SED1520, SBN1661, IL3820, MAX7219. Supported interfaces: I2C, SPI, Parallel. Features: UTF8, >700 fonts, U8x8 char output. More info	
U8glib by oliver Version 1.19.1 INSTALLED A library for monochrome TFTs and OLEDs Supported display controller: SSD1306, SSD1309, SSD1322, SSD1325, SSD1327, SH1106, UC1601, UC1610, UC1611, UC1701, ST7565, ST7920, KS0108, LC7981, PCD8544, PCF8812, SBN1661, TLS8204, T6963. More info Select version Install	

Close

5.Press the "Upload" button to upload firmware.

Image: Second Secon

6.Check the Display Screen.



• Scene Demo

Branch: master - New pull request	Create new file Upload files Find fil
i TopgunZh Update 2018-12-25	Latest co
in doc	Add 2018-12-25
a driver	Update 2018-12-25
image	Update 2018-12-25
scene_demo/joystick_with_uArm	Update 2018-12-25
in sch	Add 2018-12-25
README.md	update

At the moment, there is one routine under the "scene_demo" folder (we will keep updating the routine, please check our Github content).

joystick_with_uArm

1.Connect the uArm Controller and uArm Swift Pro with Type-C. (The program use the UART2-TTL interface)



2.Plug in the uArm Swift Pro.



3.Connect the uArm Controller and Computer with USB.





4.Download Project File: joystick_with_uArm.ino

(https://github.com/uArm-Developer/Controller/tree/master/scene_demo/joystick_with_uArm)

About how to download a single file from Github, please refer to: (https://github.com/uArm-Developer/SwiftProForArduino/

wiki/How-to-download-single-file-from-GitHub)

5.Configure Arduino IDE, select the board as "Arduino/Genuino Mega or Mega 2560", Select the COM that allocated randomly by your computer, please refer to the following picture:

💿 joystick_with_uAr	×		
File Edit Sketch To	ols Help		
	Auto Format	Ctrl	+T
	Archive Sketch		
joystick_with_u/	Fix Encoding & Reload		
#include <u8glib< td=""><td>Manage Libraries</td><td>Ctrl</td><td>+Shift+I</td></u8glib<>	Manage Libraries	Ctrl	+Shift+I
#include <stdio.1< td=""><td>Serial Monitor</td><td>Ctrl</td><td>+Shift+M</td></stdio.1<>	Serial Monitor	Ctrl	+Shift+M
#include "coord	Serial Plotter	Ctrl	+Shift+L
winciale coord_	WiFi101 / WiFiNINA Firmware Updater		
#define LCD_CS	Board: "Arduino/Genuino Mega or Mega 2560"		
#define LCD_RES	Processor: "ATmega2560 (Mega 2560)"		
#define LCD_CD	Port: "COM6 (Arduino/Genuino Mega or Mega 2560)"	•	
#define LCD_SCK Get Board Info			
#define LCD_MOSI			

Note:When uArm Controller is connected with a computer, please press the "Reset" button.

6.Open the "Manage Libraries".



7.Install u8glib.



8.Press the "Upload" button to upload firmware.

00	joysti	ick_with_uArm Arduino 1.8.8 -	-	×
<u>F</u> ile	<u>E</u> dit	<u>Sketch T</u> ools <u>H</u> elp		
0	0	🗈 🛨 🗶		Ð
in	vetick	with uArm & coord convertion _ coord convertib		

9.Test, and use the Joystick to control the movement of uArm (i.e. move around); use the rotary potentiometer to control the up-and-down motion of uArm; press button C to adjust the speed; press button D to open/close the gripper.

Release Note

Versior	Description	
V1.0.0	Establish	Topgun
V1.0.1	Add Preset Demo Tutorial	Topgun