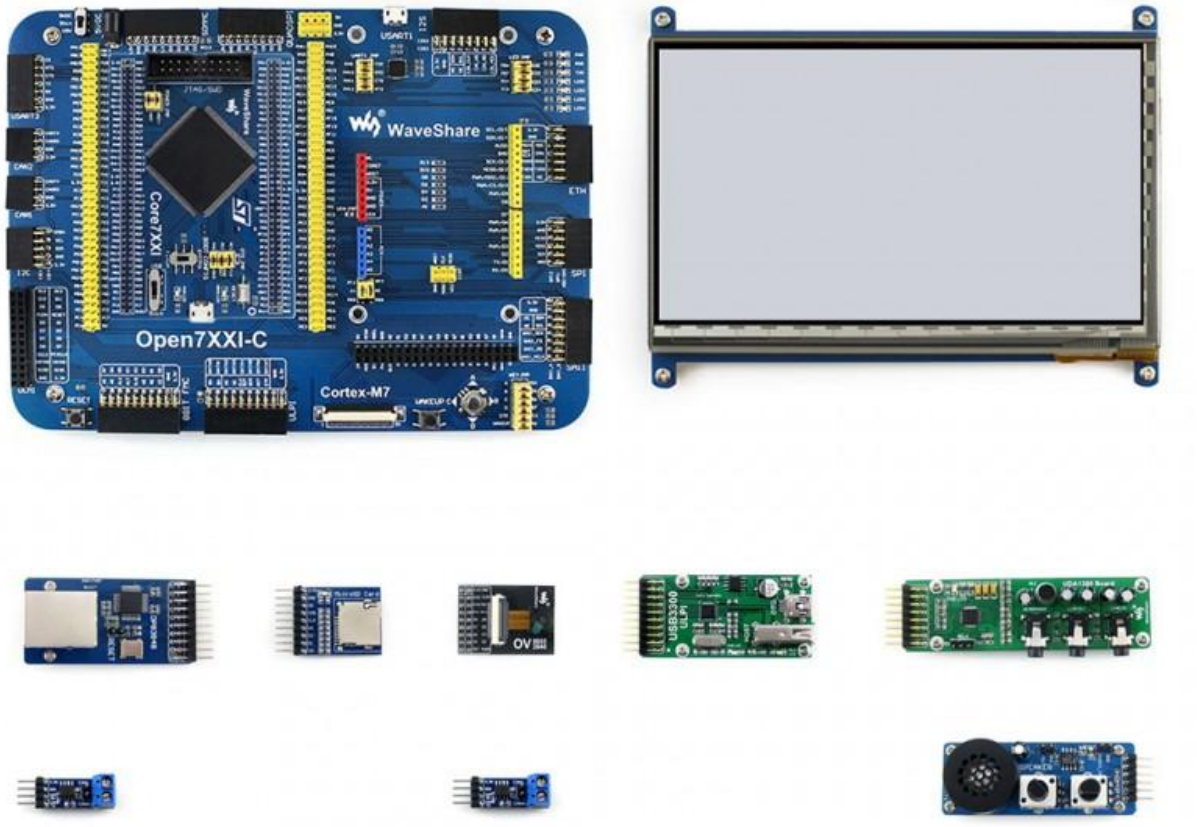
Open746I-C Package B



Introduction

STM32 development board designed for STM32F746I, features the STM32F746IGT6 MCU, and integrates various standard interfaces, pretty easy for peripheral expansions.

More

Getting Started with modules

We provide various modules for Open746I-C development board aiming to improve your development efficiency. These modules are not only designed for STM32 but also can be used for many other MCU platforms. Let's begin with the demos.

Development Environment

- KEILMDK Version：5.12 or above.
- Programmer/Debugger: ST-LINK V2
- Programming/Debugginginterface:JTAG/SWD
- Results of demo which based on serial port are all checked via onboard CP2102; connect the USB cable to the USART1 interface.
- Serial port settings:

Baud rate	115200
Data bits	8
Stop bits	1
Parity bits	None
Flow control	None

Note: All the below Demo results are available when push the reset button after program downloaded.

Conventions

- The following table provides the conventions used for the ON and OFF settings in the present document.

Convention	Definition
Jumper JP1 ON	Jumper fitted
Jumper JP1 OFF	Jumper not fitted

Sample Program Description

- **LED:**

Name	Description	Hardware Connection	Expected result
LED	GPIO output.	LED JMP ON	LED1 to LED4 will be turned on in sequence.

- **KEY:**

Name	Description	Hardware Connection	Expected result
KEY	GPIO input/output.	LED JMP ON, KEY JMP ON	The LED status will keep changing when push the buttons.

- **Interrupt:**

Name	Description	Hardware Connection	Expected result
Interrupt	GPIO interrupt.	LED JMP ON, KEY JMP ON	LED1 status will be changed by the button pressed.

- **TIM:**

Name	Description	Hardware Connection	Expected result
TIM	Timer.	LED JMP ON	LED1 flashes.

- **PWM:**

Name	Description	Hardware Connection	Expected result
PWM	Using PWM timer.	LED JMP ON	LED1 status changes gradually.

- **USART:**

Name	Description	Hardware Connection	Expected result

USART_Printf	Retarget the *printf* function using HAL polling.	Connect the on-board UART1 to a PC with a USB cable. UART1 interface is connected to USART1 by default, which can be changed to other USART interface by setting UART1 JMP.	Download the program and press RESET button. Serial output:
			`UART Printf Example:`
			`retarget the C library`
			`printf function to the UART`
			`welcome to`
			`www.waveshare.com !!!`

USART_IT	HAL interrupt for UART.	Ditto.	Download the program and press RESET button, then enter 10 characters (e.g. Open7XXI-C) and send them. Serial output:
			`****UART-Hyperterminal communication based on IT ****`
			`Enter 10 characters using keyboard:`
			`Open7XXI-C`
			`Example Finished`

USART_DMA	HAL DMA for UART.	Ditto.	Download the program and press RESET button. Serial output:
			`**** UART-Hyperterminal communication based on DMA ***`
			`WaveShare Open7XXI-C`

			Board

- **ADC+DMA:**

Name	Description	Hardware Connection	Expected result
ADC+DMA	AD acquisition demo, DMA transfer.	Connect Analog Test Board to SPI1 (ADC+DAC) connector. Image	Rotate the onboard potentiometer. Serial output: ******** ADC DMA Example ******** AD1 value = 3.298V AD2 value = 1.647V ******** ADC DMA Example ******** AD1 value = 3.298V AD2 value = 1.647V

- **DAC:**

Name	Description	Hardware Connection	Expected result
DAC	DA output demo, output via DMA channel.	Connect the Analog Test Board to the SPI1 (ADC+DAC) connector. Connect the Analog Test Board onboard 5V interface to the board onboard 5V interface via jumper wire. Image	You may hear sound from the Analog Test Board.

- **I2C-AT24C02:**

Name	Description	Hardware Connection	Expected result
I2C-AT24C02	Read and write data on E2PROM via I2C protocol.	Connect the AT24/FM24 Board to the board via I2C connector (I2C1or I2C2, depending on the software configuration). Image	Serial output: ***************I2C Example******************************* EEPROM 24C02 Write Test OK EEPROM 24C02 Read Test OK

- **SPI-W25QXX:**

Name	Description	Hardware Connection	Expected result
SPI-W25QXX	Drive the W25QXX DataFlash Board via SPI interface.	Connect the W25QXX DataFlash Board to SPI1 connector. Image	Serial output: SPI-W25Qxxx Example W25Qxxx ID is : 0xEF 0x17 QSPI Erase Block ok QSPI Write ok QSPI Read ok QSPI Read Data : 0x00 0x01 0x02

			0x03 0x04 ..... 0xFF W25Q128FV QuadSPI Test OK

- **CAN:**

Name	Description	Hardware Connection	Expected result
CAN	CAN modules communication.	Connect the two CAN modules to the onboard CAN1 and CAN2 interface. Connect the two CAN modules via jumper wire (CANL <-> CANL, CANH <-> CANH) Image	Serial output: **** This is CAN test program **** StdId : 123 RxMsg : CAN Test StdId : 123 RxMsg : CAN Test

- **PWR:**

Name	Description	Hardware Connection	Expected result
PWR	STM32 low power mode demo.		Press the WAKEUP button to enter Stop Mode, in this case, LED1 stops flashing. Press the WAKEUP button again or wait 20s to quit Stop Mode, in this case LED1 flashes. (Note: You can modify the macro definition in stm32f7xx_lp_modes.h for choosing different low power modes.) Serial output: ******** STM32F7 LowPower Test ******* Press button to enter LP modes StopMode! Automatic Wake-up using RTC clocked by LSI (after ~20s) ... StopMode wake up ,system running continue Press button to enter LP modes

- **RTC:**

Name	Description	Hardware Connection	Expected result
RTC	Real-Time Clock in the STM32 MCU.		You can modify the MX_RTC_Init function in rtc.c to set the time. Serial output: 2015/09/08 18:50:00 2015/09/08 18:50:01

- **MCU TEMPERATURE:**

Name	Description	Hardware Connection	Expected result
MCU	STM32 inner temperature		Serial output:

TEMPERATURE	measurement.		MCU Temperature : 32.6 ℃ MCU Temperature : 32.6 ℃ MCU Temperature : 32.6 ℃

- **IWDG:**

Name	Description	Hardware Connection	Expected result
IWDG	Independent watchdog program.		Serial output: ***** WaveShare Open7XXI-C Board ***** Refreshes the IWDG !!! Refreshes the IWDG !!! Refreshes the IWDG !!!

- **WWDG:**

Name	Description	Hardware Connection	Expected result
WWDG	Window watchdog program.		If the watch dog is not updated, the program will restart. Serial output: ***** WaveShare Open7XXI-C Board ***** waveshare.net !!! waveshare.net !!! waveshare.net !!!

- **RNG:**

Name	Description	Hardware Connection	Expected result
RNG	Random number generator.		Generates a 32-bit random number. Serial output: Random 32bit Numbers : 0x3664130B !!! Random 32bit Numbers : 0xFF7D82B4 !!! Random 32bit Numbers : 0xD1BAFF04 !!! Random 32bit Numbers : 0xAAC48854 !!!

- **CRC:**

Name	Description	Hardware Connection	Expected result
CRC	CRC checking.		Serial output: ****** CRC Test Example ***** CRC right value

- **SDIO:**

Name	Description	Hardware Connection	Expected result

SDIO	Read and write SD card.	Connect the Micro SD Storage Board to the board via SDIO interface. Insert the SD card to the Micro SD Storage Board socket. Image	Warning: This program may erase all the TF card data. Make sure you have backed up. Serial output:
			`Warning: this program may erase all the TF card data. Make sure you have backed up. Press 'y' to continue.`
			`Initialize SD card successfully!`
			`SD card information!`
			`CardCapacity : 8053063680`
			`CardBlockSize : 512`
			`RCA : 2`
			`CardType : 2`
			`Enable wide bus operation successfully!`
			`Write block successfully!`
			`00:0x15151515 01:0x15151515 ..... 7f:0x15151515`
			`Read block successfully!`
			`00:0x15151515 01:0x15151515 ..... 7f:0x15151515`
			`Erase block successfully!`
			`Read block successfully!`
			`00:0xffffffff 01:0xffffffff ..... 7f:0xffffffff`

- **FATFS:**

Name	Description	Hardware Connection	Expected result

FATFS	Read and write SD card, of which file system is FAT.	Connect the Micro SD Storage Board to the board via SDIO interface. Insert the SD card to the Micro SD Storage Board socket. Image	Note: Please first make sure the FATFS file system is exist in your SD card. Serial output:
			`****** FatFs Example ******`
			`Mounted successfully!!!`
			`Opened file successfully!!!`
			`Wrote successfully!!!`
			`Write Data : This is STM32 working with FatFs`
			`Closed successfully!!!`
			`Opened file successfully!!!`
			`Read successfully!!!`
			`Read Data : This is STM32 working with FatFs`
			`Closed successfully!!!`
			`FatFs is working well!!!`

- **DCMI-OV2640:**

Name	Description	Hardware Connection	Expected result
DCMI-OV2640	The camera snaps pictures.	Connect the OV2640 Camera Board to the onboard DCMI interface. Run the software	Press WAKE UP button to take a picture. Image (Note: You can modify the OV2640_320x240_JPEG parameters

		camera test.exe (in the *Software* directory). Choose a COM port and set parameters. Image	in ov2640.c for changing the resolution of a image.)

- **I2S-UDA1380:**

Name	Description	Hardware Connection	Expected result

I2S-UDA1380	Drive the UDA1380 Board to play music via I2S protocol	Connect the UDA1380 Board to the board via I2S interface. Connect an earphone to the UDA1380 Board via LINEOUT connector. Image	Press RESET button to play music. Serial output:
			`Welcome to use UDA1380 I2S test:`
			`WaveDataLength:1003324`
			`UDA1380 Init OK!`
			`AudioRemSize:370563`
			`AudioRemSize:305028`
			`AudioRemSize:239493`
			`AudioRemSize:173958`
			`AudioRemSize:108423`
			`AudioRemSize:42888`
			`AudioRemSize:0`
			`The data is completely transmitted.`

- **SAI:**

Name	Description	Hardware Connection	Expected result

SAI	Drive the UDA1380 Board to play music via SAI interface.	Connect UDA1380 Board to the board via SAI1 interface. Connect the earphone to the UDA1380 Board via LINEOUT connector. Image	Press RESET button to play music. Serial output:
			`Welcome to use UDA1380 SAI test:`
			`UDA1380 Init OK!`
			`Data transmission begin...`
			`AudioRemSize:370485`
			`AudioRemSize:304950`
			`AudioRemSize:239415`
			`AudioRemSize:173880`
			`AudioRemSize:108345`
			`AudioRemSize:42810`
			`AudioRemSize:0`
			`The data is completely transmitted.`

- **FSMC-NANDFLASH:**

Name	Description	Hardware Connection	Expected result

FSMC-NANDFLASH	Read and write NAND FLASH via FMC.	Connect the Nand Flash Board to the board via 8BIT FMC interface. Image	Serial output:
			`***** NandFlash Example ******`
			`Nand Flash ID = 0xEC,0xF1,0x00,0x95 Type = K9F1G08U0B`
			`Written to the number of : 0x00 0x01 0x02 0x03 ..... 0xFF`

			Read several: 0x00 0x01 0x02 0x03 ...... 0xFF NandFlash Read Write Test OK

- **FSMC-SDRAM:**

Name	Description	Hardware Connection	Expected result
FSMC-SDRAM	Read and write SDRAM via FMC.		Serial output: ****** SDRAM example !!! ****** /* Write data to the SDRAM memory */ 00:0xA244250F 01:0xA2442510 ……FF:0xA244260E /* Read back data from the SDRAM memory */ 00:0xA244250F 01:0xA2442510 ……FF:0xA244260E SDRAM Test OK

- **LDTC:**

Name	Description	Hardware Connection	Expected result
LDTC	LCD display demos. These 3 demos, 4.3inch 480x272, 7inch 800x600 and 7inch 1024x600, are used for 3 kinds of LCD separately.	For 4.3inch 480x272 Touch LCD (B) module: Connect it to the LCD interface. For 7inch 800x600 LCD or 7inch 1024x600 LCD, connect it to the FFC LCD interface with a FFC cable. (Note: multiple LCDs are not allowed to connect at the same time.)	Static pictures are shown on the LCD.

- **DMA2D:**

Name	Description	Hardware Connection	Expected result
DMA2D	This LCD display demo is compatible with 4.3inch 480x272, 7inch 800x480 and 7inch 1024x600 LCD to show dynamic pictures.	For 4.3inch 480x272 Touch LCD (B) module: Connect it to the LCD interface. For 7inch 800x600 LCD or 7inch 1024x600 LCD, connect it to the FFC LCD interface with a FFC cable. (Note: multiple LCDs are not allowed to connect at the same time.)	LCD shows 2 moving and stacked pictures. • 4.3inch 480x272 Display • 7inch 800x480 and 7inch 1024x600 LCD Display

- **LCD_DISPLAY:**

Name	Description	Hardware Connection	Expected

			result
LCD_DISPLAY	This LCD display demo is compatible with 4.3inch 480x272, 7inch 800x480 and 7inch 1024x600 LCD to show characters.	For 4.3inch 480x272 Touch LCD (B) module: Connect it to the LCD interface. For 7inch 800x600 LCD or 7inch 1024x600 LCD, connect it to the FFC LCD interface with a FFC cable. (Note: multiple LCDs are not allowed to connect at the same time.)	Characters are shown on the LCD. Image

- **Touch:**
  - This LCD display demo is compatible with 4.3inch 480x272 (Resistive), 7inch 800x480 (Capacitive) and 7inch 1024x600 (Capacitive) LCD.

Name	Description	Hardware Connection	Expected result
Touch 4.3inch 480x272	This demo is used for 4.3inch 480x272 (Resistive) LCD.	Connect a 4.3inch 480x272 Touch LCD (B) module to the LCD interface.	Tap the Adjust area for calibration and then goto a sketchpad interface, on which you can paint with different colors. Image
Touch 7inch 800x480 / Touch 7inch 1024x600	This demo is used for 7inch 800x480 / Touch 7inch 1024x600 (Capacitive) LCD.	Connect the LCD to the FFC LCD interface with a FFC cable.	The LCD shows the tapped locations up to 5. Image

- **STemWin:**

Name	Description	Hardware Connection	Expected result
STemWin	STemWin interlayer migration GUI program.	This demo is compatible with 4.3inch 480x272, 7inch 800x480 and 7inch 1024x600 LCD but you should download the corresponding project.	The LCD shows STemWin demo GUI which looks so cool. Image

- **USB FS:**

Name	Description	Hardware Connection	Expected result
USB FS Device (CDC_Standalone)	USB FS device CDC demo. Your PC will recognize the board as a Virtual COM Port.	• OTG JMP ON, UART1 JMP OFF. By default, you should connect a 7 inch 1024x600 LCD to your board, but the program should be modified	Please unzip and install the driver, stsw.zip, in the *Software* directory and your PC will recognize the board as a "STMicroelectronics Virtual COM Port" which is listed on the Device Manager. Information sent from PC through COM Port will be shown on the LCD. For example, what

		appropriately for other LCD. • Connect the core board USB port to your PC using a micro USB cable. Image	you sent: `WaveShare Open7XXI-C Board STM32 Virtual COM Port Driver` `WaveShare Open7XXI-C Board STM32 Virtual COM Port Driver` `WaveShare Open7XXI-C Board STM32 Virtual COM Port Driver` what you get: Image
USB FS Device (HID_Standalone)	USB FS device HID demo. Your PC will recognize the board as a mouse.	Ditto.	A new mouse will be listed on the Device Manger and you can move the onboard joystick to control the cursor on PC. Image
USB FS Device (MSC_Standalone)	USB FS device MSC demo. Your PC will recognize the board as a USB drive.	Ditto. Besides, you shall connect a Micro SD Storage Board, with a Micro SD card inserted, to the SDIO interface.	A "USB Mass storage device" will be listed on the Device Manager and a removable hard drive will appear on "this PC". Image
USB FS Host (HID_STandalone)	USB FS host HID demo. The board can recognize a mouse device.	OTG JMP ON, UART1 JMP OFF. By default, you should connect a 7 inch 1024x600 LCD to your board, but the program should be modified appropriately for other LCD.	Connect a mouse to the core board with a OTG cable (type A to micro USB) and then the green dot on the LCD will move following the mouse. Image
USB FS Host (MSC_STandalone)	USB FS host MSC demo. The board can recognize a USB drive.	Ditto.	Connect a USB flash drive to the core board with a OTG cable (type A to micro USB), then press the User button to get the info of your USB drive and directories. Image
USB FS Host (DynamicSwitch_Standalone)	USB FS host Dynamic Switch demo. Using the same program, the board can recognize a USB drive or a mouse	Ditto.	The board can recognize either a USB drive or a mouse device in use. Image of using USB drive

	device.		

- **USB HS:**

Name	Description	Hardware Connection	Expected result
USB HS Device (HID_STANdalone)	USB HS Device HID demo. Your PC will recognize the board as a mouse.	• Connect a USB3300 module to the ULPI interface. • Connect the OTG port, on the USB3300 module, to your PC with a OTG cable (type A to mini USB). Image	A new mouse will be listed on the Device Manger and you can move the onboard joystick to control the cursor on PC.
USB HS Device (MSC_Standalone)	USB HS Device MSC demo. Your PC will recognize the board as a USB drive.	Ditto. Besides, you shall connect a Micro SD Storage Board, with a micro SD card inserted, to the SDIO interface.	A "USB Mass storage device" will be listed on the Device Manager and a removable hard drive will appear on "this PC".
USB HS Host (MSC_Standalone)	USB HS Host MSC demo. The board can recognize a USB drive.	Connect a USB3300 module to the ULPI interface.	Connect a USB flash drive to the USB3300 module with a OTG cable (type A to mini USB), then press the User button to list the files of your USB drive.

- **ETH:**

Name	Description	Hardware Connection	Expected result
LwIP_TCP_Echo_Client	TCP echo client demo.	• Copy the echotool.exe to the root directory of Drive C: • Connect a DP83848 module to the ETH interface. • Connect the DP83848 module to a router of a LAN or connect to a PC directly with a cable. Image	• Please ensure that the remote PC IP address is the same IP address as the one defined in the mxconstants.h file (192.168.1.189 by default). • Run the Command Prompt (Win + R then run the command *cmd*). • At the command prompt, enter: `C:\>echotool /p tcp /s`

			where: – /p tcp is the TCP protocol (TCP protocol) – /s is the actual mode of connection (Server mode) • When you press the User button on the board, the client sends a string and the server echoes back the same string to the client. Image
LwIP_TCP_Echo_Server	TCP echo server demo.	Ditto.	• At the command prompt, enter: `C:\>echotool IP_address /p tcp /r 7 /n 15 /t 2 /d Testing LwIP TCP echo server` where – IP_address is the actual board's IP address. By default, the following static IP address is used: 192.168.1.189 – /p tcp is the protocol (TCP protocol) – /r is the actual remote port on the echo server (echo port) – /n is the number of echo requests (for example, 15) – /t is the connection timeout in seconds (for example, 2) – /d is the message to be sent for echo (for example, "Testing LwIP TCP echo server") Image
LwIP_UDP_Echo_Client	UDP echo	Ditto.	• Please ensure that the

	client demo.		remote PC IP address is the same IP address as the one defined in the mxconstants.h file (192.168.1.189 by default). • Run the Command Prompt (Win + R then open *cmd*). • At the command prompt, enter: `C:\>echotool /p udp /s` where: – /p udp is the UDP protocol (UDP protocol) – /s is the actual mode of connection (Server mode) • When you press the User button on the board, the client sends a string and the server echoes back the same string to the client. Image
LwIP_UDP_Echo_Server	UDP echo server demo.	Ditto.	• At the command prompt, enter: `C:\>echotool IP_address /p udp /r 7 l/ 7 /n 15 /t 2 /d Testing LwIP UDP echo server` where – IP_address is the actual board's IP address. By default, the following static IP address is used: 192.168.1.189 – /p udp is the protocol (UDP protocol) – /r is the actual remote port on the

			echo server (echo port) – /l is the actual local port for the client (echo port) – /n is the number of echo requests (for example, 15) – /t is the connection timeout in seconds (for example, 2) – /d is the message to be sent for echo (for example, "Testing LwIP UDP echo server") Image
LwIP_HTTP_Server_Raw	Httpserver demo.	• Connect a DP83848 module to the ETH interface. • Connect the DP83848 module to a router of a LAN or connect to a PC directly with a cable. Image • By default, you should connect a 7 inch 1024x600 LCD to your board, but the program should be modified appropriately for other LCD.	• A message is displayed on the LCD screen indicating the success or failure of the DHCP IP address allocation (for example, 192.168.1.160). Image • On the remote PC, open a web client and type the board's IP address in a web browser. Home page of the HTTP server demo • You can click *LED control* to change the LED status on the board.

Name	Description	Hardware Connection	Expected result
FreeRTOS	This demo is based on the examples generated by STM32cubeMX. This demo contains 11 examples such as threads, mutexes, queues, signals, messages, timers and so on.	LED JMP ON	LEDs flash.

Name	Description	Hardware Connection	Expected result

uCOS III	This demo shows the uCOS III OS which is based on the HAL libraries generated by STM32cubeMX.	LED JMP ON	LED1 flashes.

Resource

- Core746I Schematic
- Open746I-C Schematic
- Demo

STM32 Software

IDE

- Keil MDK
- STM32CubeMX

Programmers

- Flash Loader for ISP
- STVP
- STM32 ST LINK Utility

USB Driver

- PL2303 Windows Driver
- ST-Link V2 USB Driver
- Virtual_COM_Port_Driver

Other Software

- Stlinkupgrade
- TCP UDP Debugger
- IpTool
- EMWToolBox_Setup
- BonjourSetup
- SecureCRT
- Camera test
- BusHound

STM32F7 Datasheets

- STM32F745_STM32F746-Datasheet
- STM32F7-Reference-EN
- STM32F7xx HAL drivers

STM32 Documents

ST libraries

- STM32_I2C_CPAL.7z

UCOS Source

- UCGUI3.90_Source.zip

Support

Emali: service@waveshare.com

Our working time: 09:00-18:00 (**UTC+8** Monday to Saturday)