MXCHIP®

User Manual

MXKit Development Kit

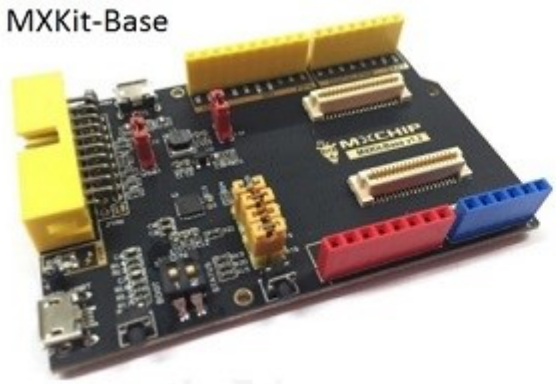
MXCHIP Development Board

Version：1.2 **Date：2022-8-12** **Number：UM0010CN**

Introduction

MXKit Development Kit is an IoT development kit from MXCHIP. It can be used for smart hardware prototype development. The development kit provides an out-of-the-box intelligent hardware solution that facilitates the verification of the user's software and functions, allowing the product to quickly and securely connect to cloud service platforms and mobile phones, shortening the development cycle, and quickly reaching the market. The MXKit development kit consists of MXKit-Base, MXKit-Core.

- MXKit-Base board is a general-purpose development board with rich peripheral interfaces that is compatible with all MXKit-Core boards;
- MXKit-Core board is an IOT access core board containing MXCHIP wireless communication module.



Update Record

Date	Version	Description
2018-1-17	1.0	Initial document
2022-08-12	1.1	Add MXKit-Core model. Add development example.
2023-01-18	1.2	Add MXKit-Core model including EMC3080 and EMC3180.

Catalog

Figure

Table

1. Characteristic

- 1 * Standard Arduino interface
- 1 * MX Port interface, connected with MXKit-Core board through board-to-board connector
- 1 * JTAG/SWD
- 2 *USB. Optional power supply:

  - When the jumper cap is connected to USB1 of J2 interface, select USB1-UART at the bottom left to supply power to the system. In addition to supplying power to the system, USB1-UART can also be used as two UART interfaces as user serial ports and debugging serial ports. Where, Enhanced COM Port is the user serial port and Standard COM Port is the debugging serial port.

  - When the jumper cap is connected to USB2 of J2 interface, select USB2 above to supply power to the system. In addition to supplying power to the system, USB2 can also be used as a USB data transmission interface.

- 1 * Power indicator D1
- 1 * 5V to 3.3V step-down conversion circuit, output 3.3V voltage for module use.
- 2 * position dial switch: BOOT and STATUS
- 2 * key: RESET and ELINK

Designed with automatic control circuit, which can control the DTR of CP2015 through the upper computer_ SCI and RTS_ SCI, and then realize the function of resetting the automatic control system and entering the Bootloader mode.

The functional block diagram of MXKit-Base is shown in Figure 1, and the functional description is shown in Table 1.

Figure 1 MXKit-Base Function Diagram

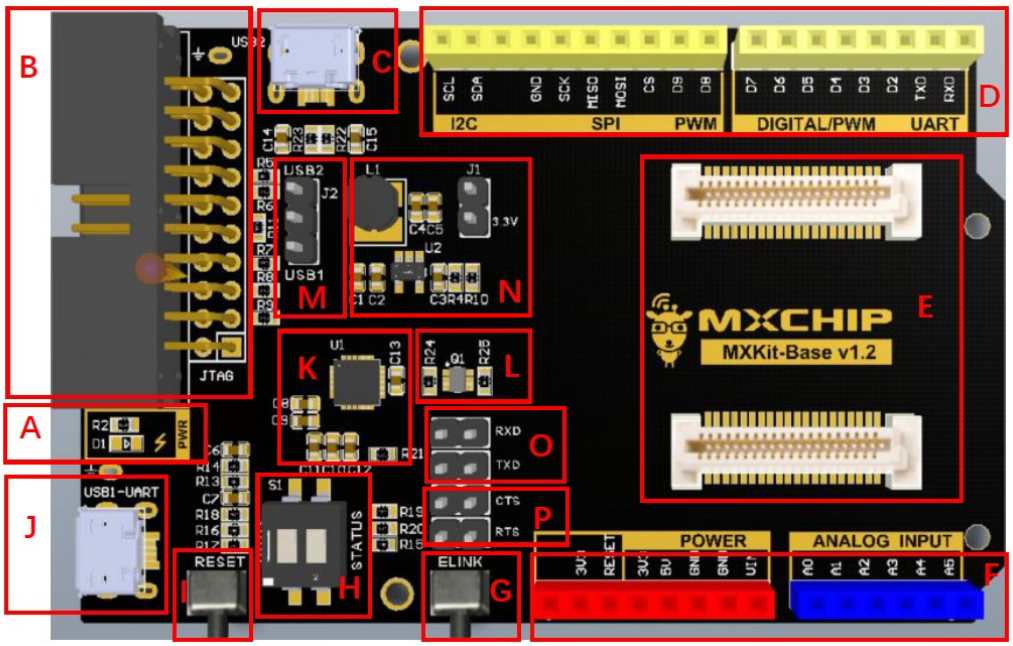


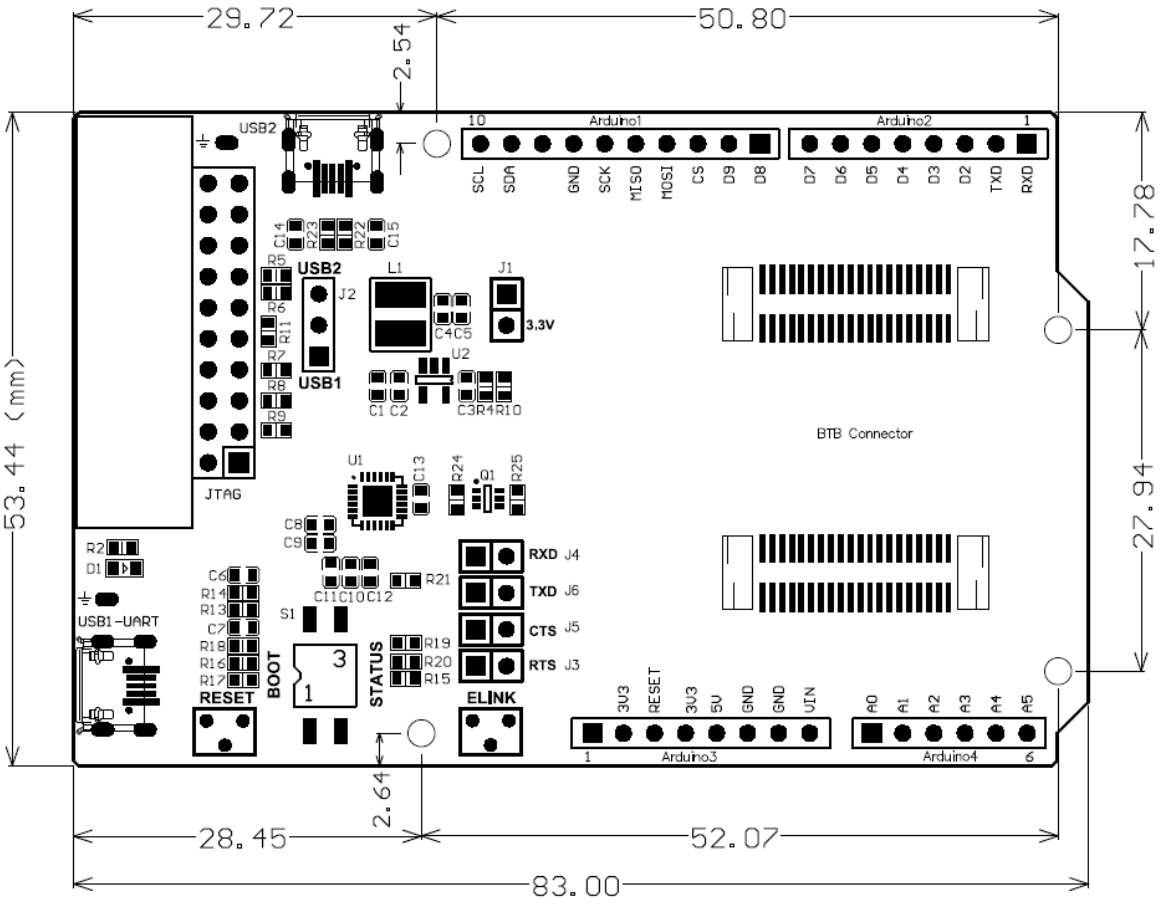
Table 1 MXKit-Base Function Description

Name	Description
A	Power indicator LED
B	JTAG / SWD
C	MicroUSB port to support power for system and USB data transmission.
D, F	Arduino
E	MXPort interface
G	Easylink button
H	2 bit dial switch (BOOT & STATUS)
I	RESET button
J	MicroUSB port to support power for system and USB to dual UART function.
K	USB to dual UART circuit
L	Automatic control circuit
M	Power selection
N	5V to 3.3V voltage convertor

Name	Description
O	Jumper terminal for UART_ TXD & USER_ UART_ RXD of wireless module ● Default is connected. At this time, the user serial port and debug serial port log information of the module are output from J; ● When the module is not connected, the user serial port log information of the module will be output from the UART port in D.
P	Jumper terminal for UART_ CTS & USER_ UART_ RTS of wireless module ● Default is connected., which enables the user serial port hardware flow control function of the module; ● When it is not connected, turn off the user serial port hardware flow control function of the module.

2. Dimension

Figure 2 MXKit-Base Overall Demension



3. Hardware layout

Figure 3 Top layout

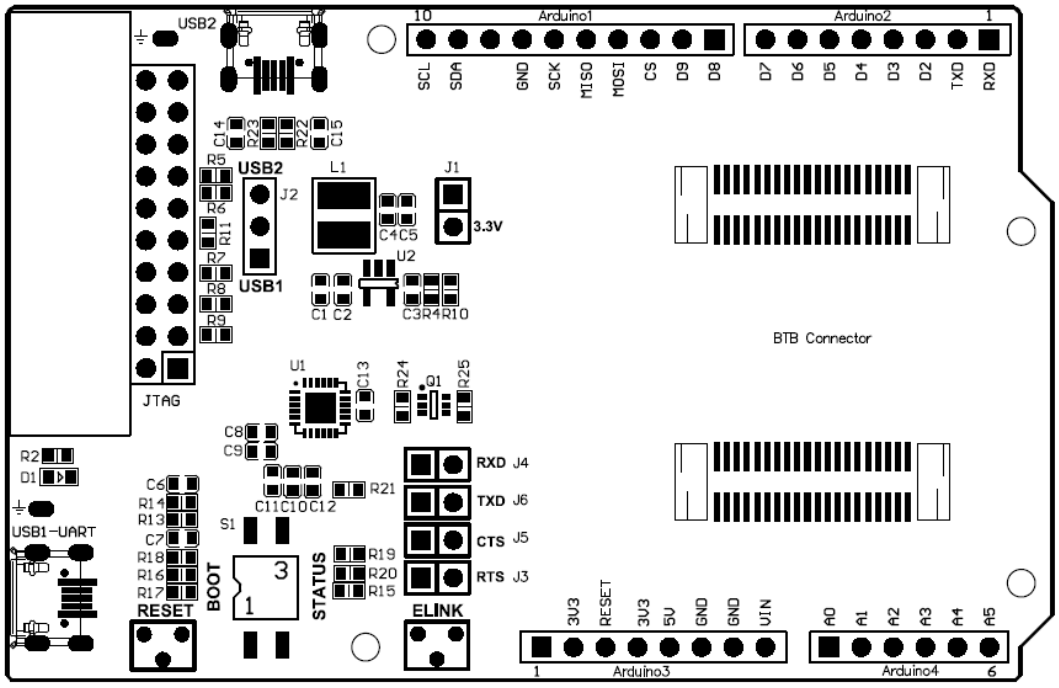
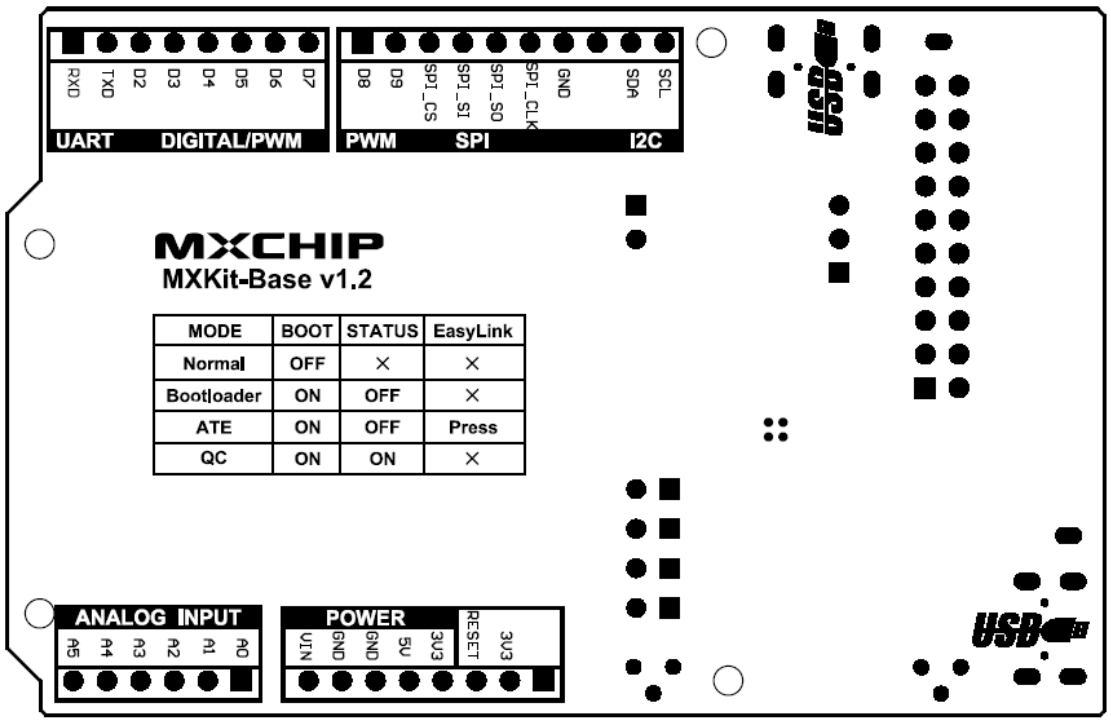


Figure 4 Bottom layout



4. MXKit-Base Detail

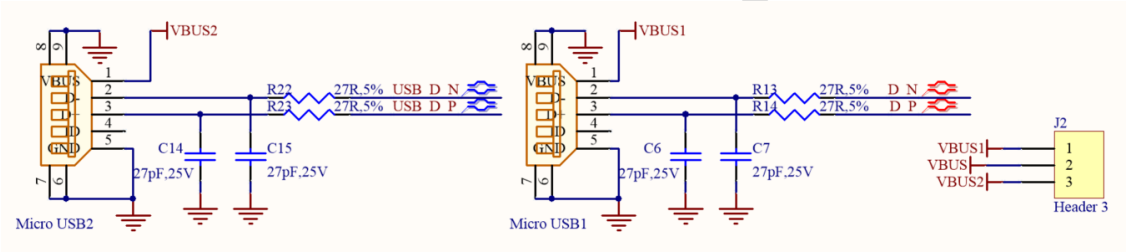
4.1 Power

MXKit-Base uses USB interface for power supply, which can be selected from USB1 or USB2 through J2. The power selection mode is shown in Table 2.

Table 2 USB Power Selection

J2	Description
Fitted: 1-2	Power supply by USB1
Fitted: 2-3	Power supply by USB2

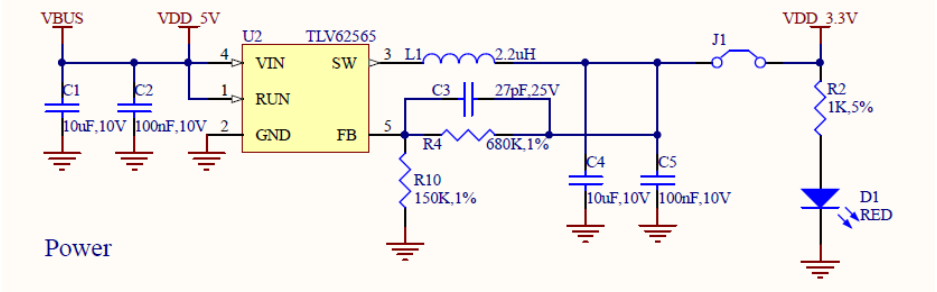
Figure 5 USB Power Supply Circuit



MXKit-Base board adopts a 3.3V high-efficiency DC-DC with the model of TLV62565. TLV62565 is a high efficiency (95%), constant frequency (1.5MHz) monolithic synchronous step-down voltage regulator. The device has adjustable output voltage versions and 1.5V, 1.8V fixed voltage output versions. The maximum output current is 1.5A, but the static loss is only 50uA. The input voltage range of 2.7V to 5.5V makes TLV62565 very suitable for single-cell lithium-ion battery power supply applications. The switching frequency is set at 1.5MHz internally, which is convenient to use small surface-mount inductors and capacitors. The device has the function of over-temperature protection. The internal synchronous switch improves the efficiency and eliminates the external Schottky diode.

The circuit schematic diagram of 5V to 3.3V is shown in Figure 6. When the power supply works normally, the power indicator D1 will be on. The current test function is designed. The current of the board can be detected through the J1 socket, and then the power consumption can be calculated.

Figure 6 5V to 3.3V Convertor Circuit



Users should pay attention to: during normal use, J1 must be short-circuited with jumper cap, otherwise 3.3V power supply cannot be output, resulting in system failure to work.

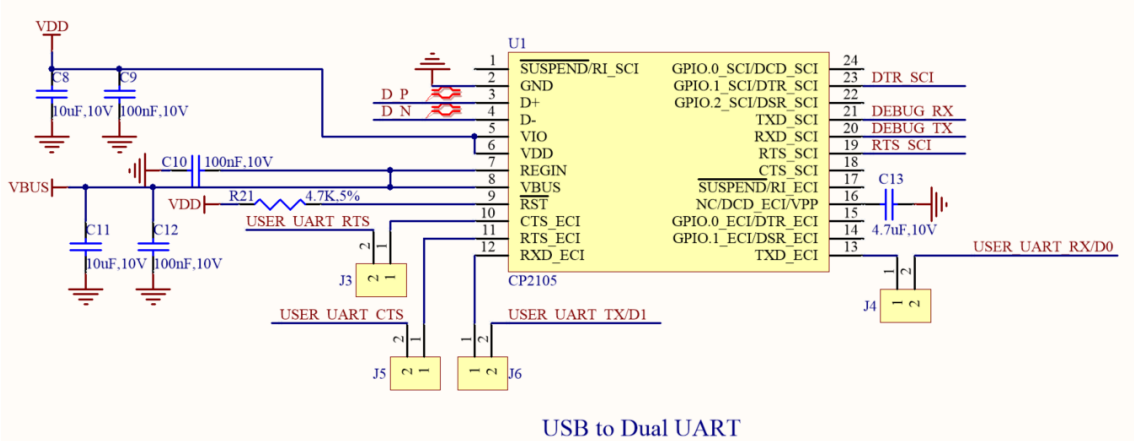
4.2 USB to Dual UART

In order to facilitate users to use and debug, the MXKit-Base board is equipped with USB dual UART circuit and chip CP2105.

- MODEM interface signal to support hardware and software flow control
- Interface supports UART 7 / 8 Bit data bits, 1 / 2 stop bits, parity check, Mark/Space check or no check;
- Communication speed: 300bps-2Mbps (TTL), 320 Byte transmission buffer;
- The level converter is integrated, and the I/O port level supports 5V-3.3V level logic.
- Fully compatible with USB 2.0

USB to dual UART is shown in Figure 7.

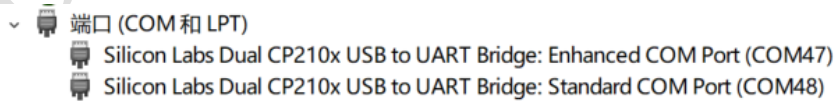
Figure 7 USB to Dual UART circuit



USB to Dual UART

MXKit-Base is connected with PC through Micro-USB. After installing the driver, two serial ports will be added to the PC port, as shown in Figure 8

Figure 8 Serial port display in PC



Enhanced COM Port is user serial port, Standard COM Port is debugging serial port.

Download address of CP2105:

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers .

The following two points need to be paid attention to during using process：

（1）In Figure 1, the O region RXD (J4) and TXD (J6) default using a jumper cap by default, when the module user serial port communicates with the PC by USB1-UART; when the module

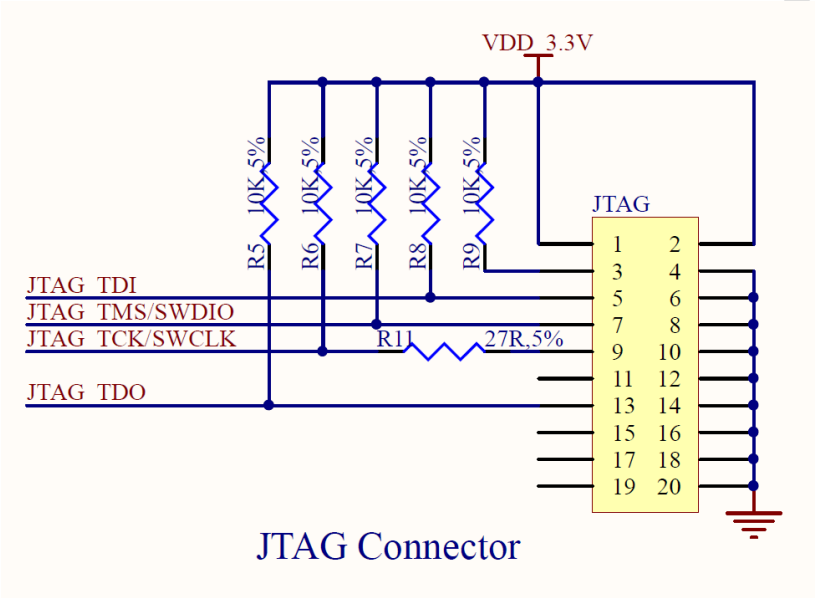
communicates with the TXD and RXD of the Arduino interface, the RXD (J4) and the jump line terminals need to be disconnected.

（2）When using the hardware flow control function of the user's serial port, the P area CTS (J5) and RTS (J3) in Figure 1 need a short jump cap, and please disconnect when the hardware flow control function is not used.

4.3 JTAG Debug Interface

MXKit-Base supports the standard JTAG (20PIN) download debugging interface, which can easily connect debugging tools such as J-LINK or U-LINK2, and supports JTAG and SWD modes. The schematic diagram of JTAG interface circuit is shown in Figure 9

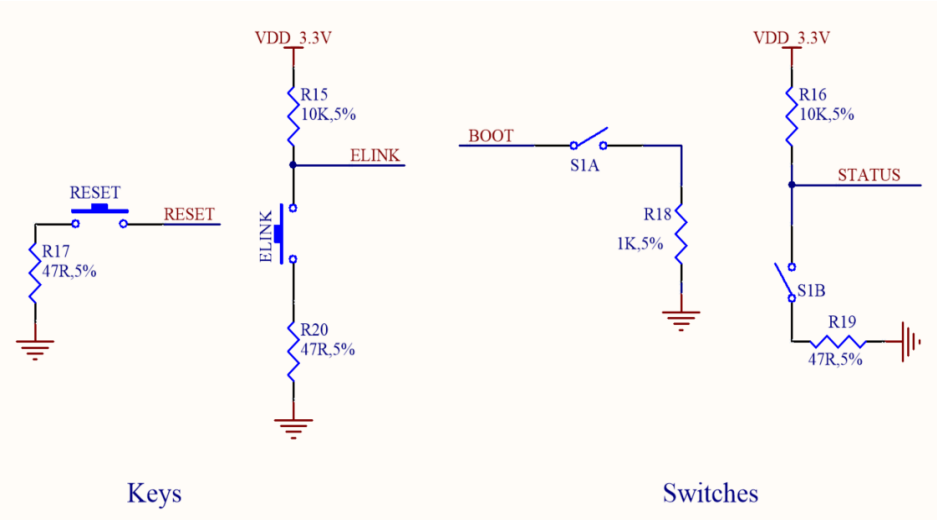
Figure 9 JTAG Debug Circuit



4.4 Switch and Button

To facilitate user debugging, the MXKit-Base board provides a 2 digit dialing switch and two keys near the bottom left side of the board. The state of BOOT and STATUS can be controlled by the code switch, and RESET and Easylink are controlled by buttons. The schematic diagram of this part is shown in

Figure 10 Switch and Button Circuit



4.5 Work Mode Selection

By controlling the dialing switch and key switch on the MXKit-Base, the user can make the MXCHIP wireless module into different working modes, for example, when the BOOT is allocated to the ON state, it will automatically enter the Bootloader mode after the power boot is started.

The corresponding state of the pin in each mode of work is shown in Table 3.

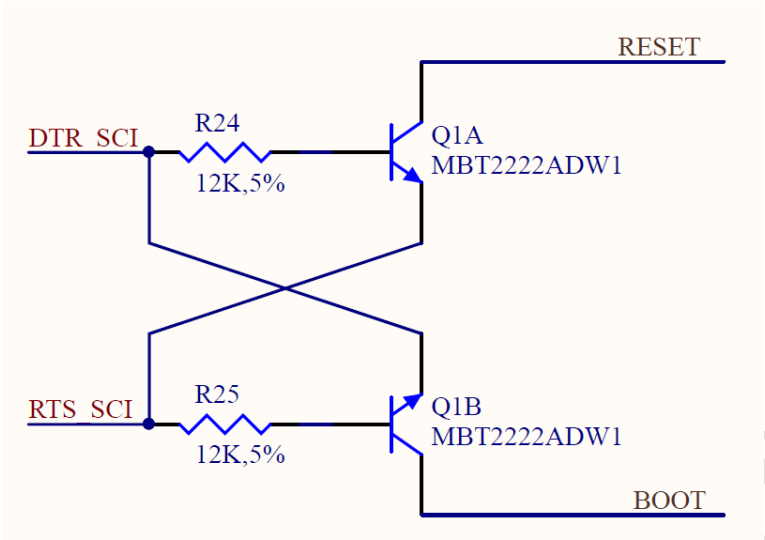
Table 3 MXKit-Base Work Mode

Mode	BOOT	STATUS	EasyLink
Normal	OFF	×	×
Bootloader	ON	OFF	×
ATE	ON	OFF	Press
QC	ON	ON	×

4.6 Automatic Control Circuit

The automatic control circuit is designed on the MXKit-Base board, which can control the DTR_SCI and RTS_SCI of CP2015 through the upper computer, and then realize the automatic control system reset and the function of entering the Bootloader mode. As shown in

Figure 11 Automatic control circuit



The automatic control logic table is shown in

Table 4 Logic Table

Auto Control Circuit			
RTS_SCI	DTR_SCI	RESET	BOOT
1	1	1	1
0	0	1	1
1	0	1	0
0	1	0	1

4.7 Arduino Port

MXKit-Base board supports the standard Arduino interface, which can be easily inserted into various Arduino interface expansion boards. The definition of the Arduino interface pin is shown in

Table 5 Arduino Pin Function

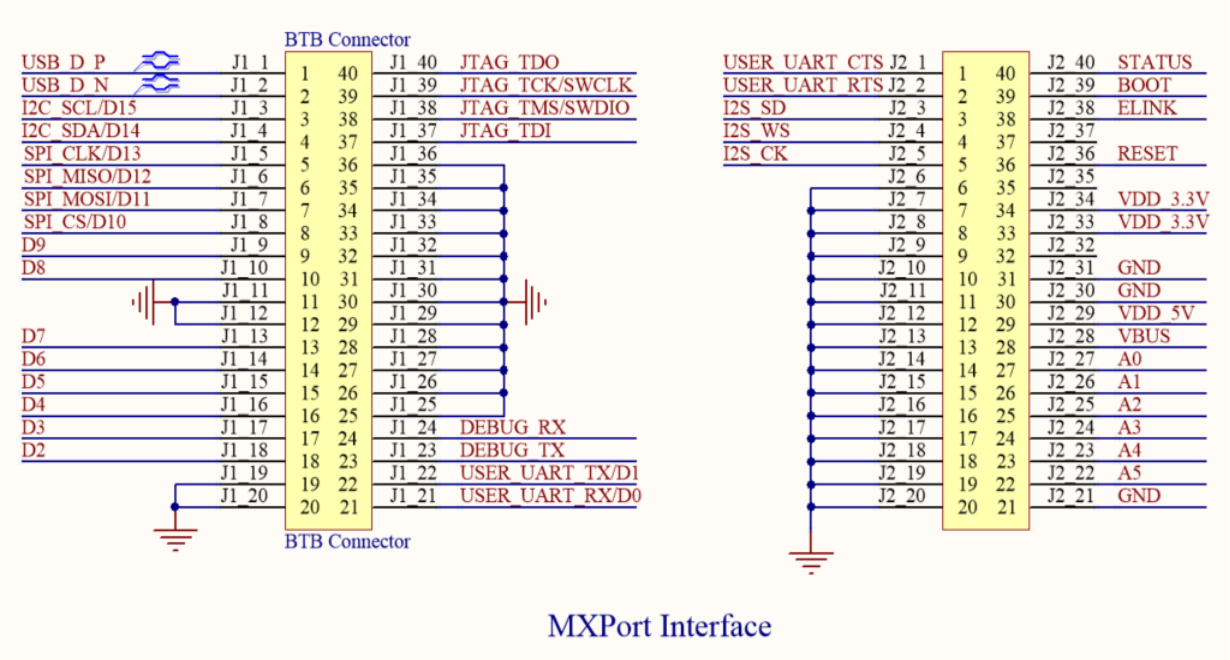
Connector	Pin Number	Pin name	Function
Arduino1 digital	10	D15	I2C1_SCL
	9	D14	I2C1_SDA
	8	NC	NC
	7	GND	ground
	6	D13	SPI_CLK
	5	D12	SPI_MISO
	4	D11	SPI_MOSI
	3	D10	SPI_CS
	2	D9	D9
	1	D8	D8
Arduino2 digital	8	D7	D7
	7	D6	D6
	6	D5	D5
	5	D4	D4
	4	D3	D3
	3	D2	D2
	2	D1	USER_UART_TX
	1	D0	USER_UART_RX
Arduino3 power	1	NC	NC
	2	VDD_3.3V	3.3V input/output
	3	RESET	RESET
	4	VDD_3.3V	3.3V input/output
	5	VDD_5V	5V output
	6	GND	ground
	7	GND	ground
	8	VIN	Power input
Arduino4 analog	1	A0	A0
	2	A1	A1

	3	A2	A2
	4	A3	A3
	5	A4	A4
	6	A5	A5

4.8 MXPort

MXKit-Base board provides a flexible MXPort interface for developers, which is compatible with all MXKit-Core boards. The MXPort interface is shown in

Figure 12 MX Port Circuit



MXPort Interface

5. MXKit-Core Detail

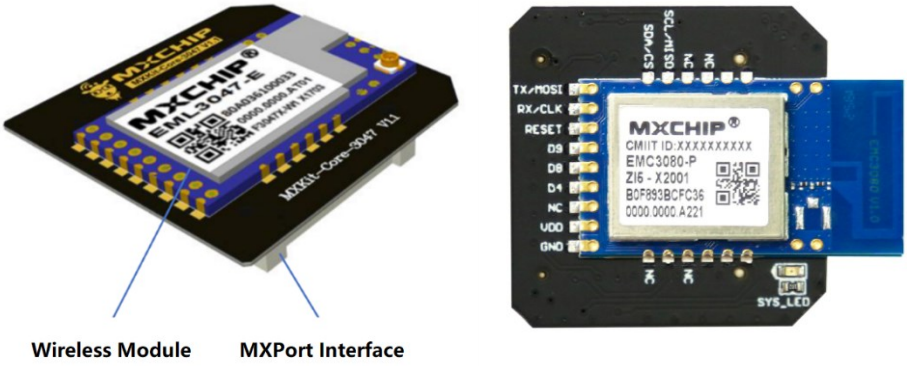
5.1 Product Components

On the MXKit-Core board, the MXCHIP wireless communication module is connected to the MXKit-Base board through the MXPort interface, mainly including:

- MXPort interface is connected with MXKit-Base board through board-to-board connector.
- MXCHIP wireless communication module, including Wi-Fi module, BLE module, Wi-Fi+BT module.

The schematic diagram of MXKit-Core board is shown in Figure 13.

Figure 13 MXKit-Core Picture



5.2 Product Model

The existing MXKit-Core board models are shown in Table 6.

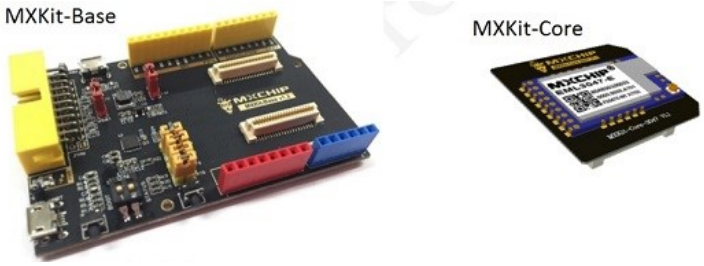
Table 6 MXKit-Core Model

MXKit-Core 型号	适用模块
MXKit-Core-3070	EMW3070 系列
MXKit-Core-3090	EMW3090、EMW3090V2 系列
MXKit-Core-3020	EMC3020 系列
MXKit-Core-3380	EMC3380 系列
MXKit-Core-3080	EMW3080、EMW3080V2 系列
MXKit-Core-3080/3280	EMC3080、EMC3280 系列
MXKit-Core-3072	EMW3072 系列
MXKit-Core-3060	EMW3060 系列
MXKit-Core-3010	EMW3010 系列
MXKit-Core-1082	EMB1082 系列

6. Development board application example

6.1 List of necessary hardware

Figure 14 MXKit Development Hardware



JLink Debugger micro USB Data Cable

- MXKit-Base *1（Used for power supply, providing reset and status keys for the core board, and user serial port connection functions）;
- MXKit-Core * 1;
- JLink V9 Debugger * 1（Used for Simulation download function for supporting software secondary development）;
- Micro-USB Data Cable * 1 （Used to power the development board and communicate data between PC and module）。

6.2 Hardware connection mode

- Connecting serial port: connect the MXKit development kit to the PC through Micro-USB cable to provide power and connect serial port for the development kit, namely:

  User PC USB port -->Micro USB cable -->Development kit microUSB port

- Connecting simulator: connect the JTAG port of the development kit to the PC through the JLink or STLink (only for ST chips) simulator, namely:

  User PC USB port ->USB cable ->simulator ->JTAG cable ->development kit

6.3 Development based on AT firmware

Based on the AT application firmware development, it is not necessary to connect the emulator, but only to connect the serial port.

- Software preparation:

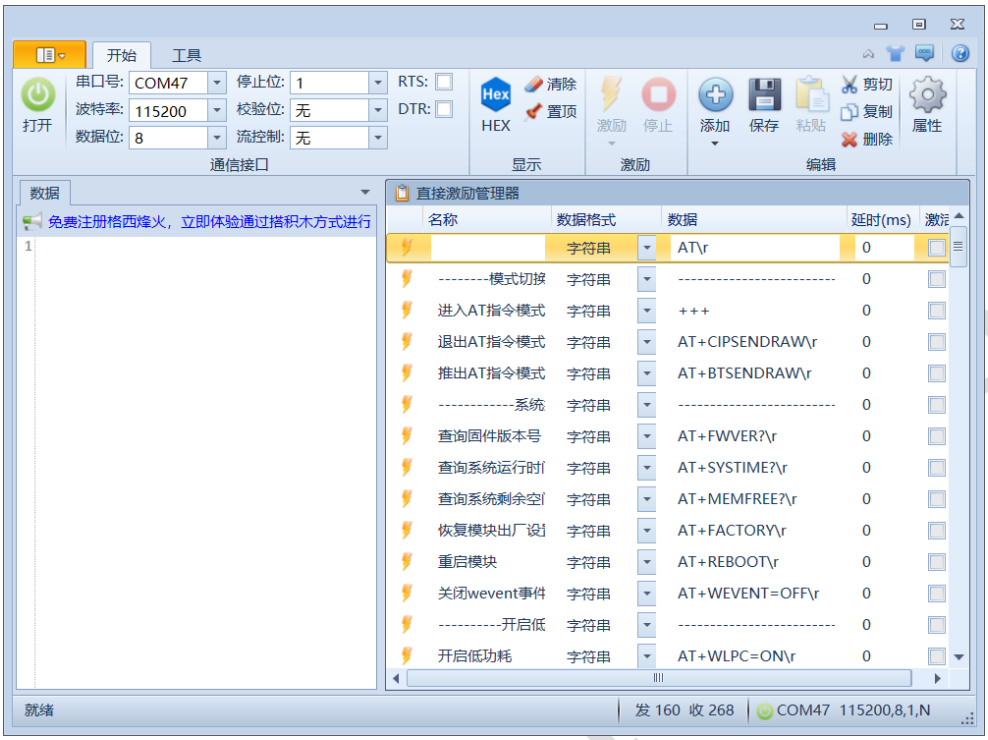
  1. The AT firmware of the corresponding module, download address: https://docs.mxchip.com/en/ohlmv1/xstr46.html

  2. AT command serial port debugging software, download address: http://www.geshe.com/zh-cn/products/ggenius.

- Open the serial port:

  Connect the user serial port of the development kit to the PC through the Mini USB cable, and find the enhancement COM port number in the PC's control panel - device manager, open the serial port tool, and create the connection. The configuration is shown in Figure 15 (taking EMW3080 as an example):

Figure 15 Serial Port Tool Configuration



- **Enter AT Command Mode**

  Open Serial tool, and send "+++", then return "OK", which means enter to AT Command Mode successfully. As shown in Figure 16.

Figure 16 Enter into AT Command Mode

```
11
12 [2017-08-01 13:50:35.587 T]+++
13 [2017-08-01 13:50:35.702 R]
14 OK
```

For more usage of AT instruction, refer to the document:
https://docs.mxchip.com/en/ohlmv1/eca5x9.html .

- **Ensure AT Command Mode**

  Send "+++", and return "OK", which means enter to AT Command Mode successfully. As shown in Figure 17.

Figure 17 Ensure in AT Command Mode

```
1 [2018-05-10 17:14:11.910 T]AT
2
3 [2018-05-10 17:14:11.939 R]AT
4
5 OK
```

6.4 Development Based on MXOS firmware

For module firmware development based on MXOS, it is necessary to connect the JLinkV9 simulator and the serial port for debugging.

To use the MXKit development kit for the secondary development of module firmware, a development environment needs to be built first. See the MXOS development document on the official website for details:https://docs.mxchip.com/en/xgk73y/fq9v4g/yaqqcr2oa1za0xs3.html .

7. Sales and Technical Support Information

If you need to consult or purchase this product, please call Shanghai MXCHIP Technology Co., Ltd. during office hours.

Office hours: Monday to Friday morning: 9:00~12:00, afternoon: 13:00~18:00

Tel:+86-21-52655026

Contact address: 9/F, No. 5, Lane 2145, Jinshajiang Road, Putuo District, Shanghai

Post code: 200333

Email: sales@mxchip.com