Banana Pi BPI-BC95

1. Hardware introduction
2. Software introduction
3. How to setup and start

— Hardware introduction

There are two major components on the board,namely NB-IoT module and MCU, their specs are below:

1. NB-IoT module:

NB-IoT module： Quectel BC95-B5/8/20 or BC95-G

Model and Frequency band：

Model	BC95-B20	BC95-B5	BC95-B8	BC95-G
Frequency	800MHz	850MHz	900MHz	B1/B3/B5/B8/B20/B28 @FDD-LTE (TBD)

SIM card：

1).Micro SD SIM Slot

2).e-SIM optional(footprint reserved on the board)

Antenna： IPEX antenna connector

2. MCU:

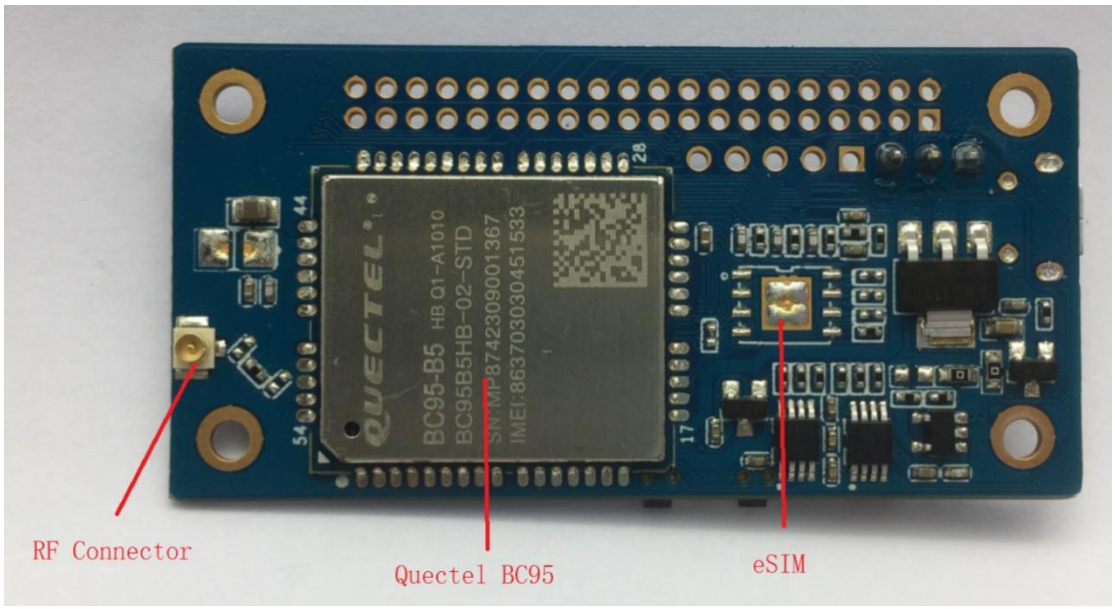
	BC95-B5/8/20	BC95-G
SoC	STM32F103RCT6	STM32F103RBT6
CPU	ARM 32-bit Cortex™-M3 CPU @72MHz	
FLASH	256KB	128KB
RAM	48KB	20KB
Functions	UART、I2C、SPI、PWM	

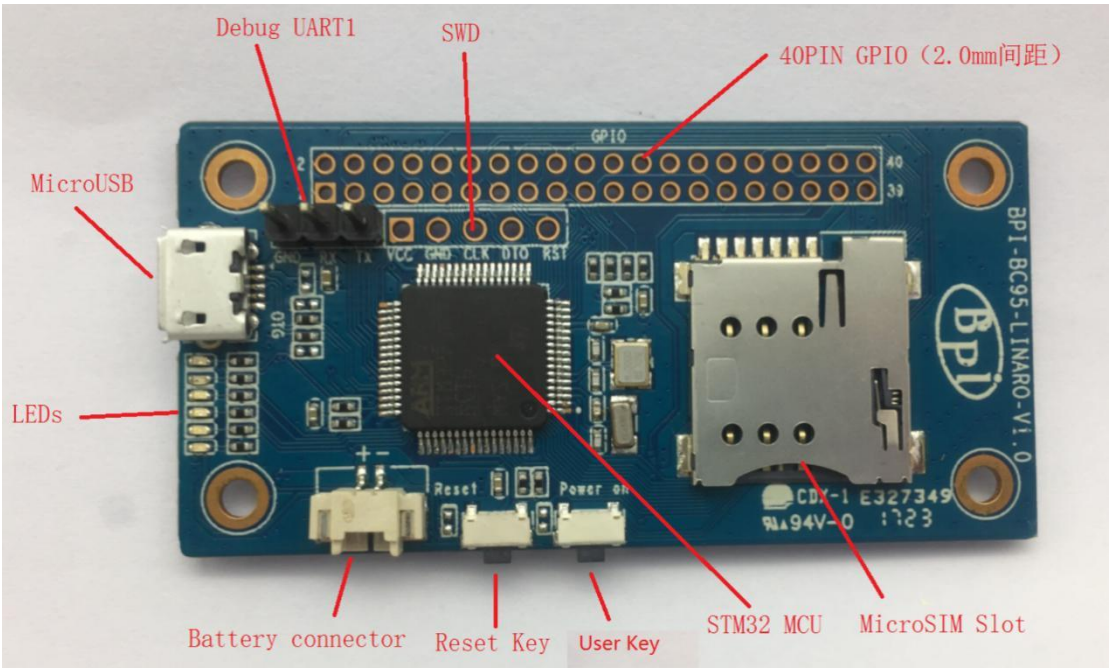
3. Development board :

Hardware Specification of BPi BC95-LINARO

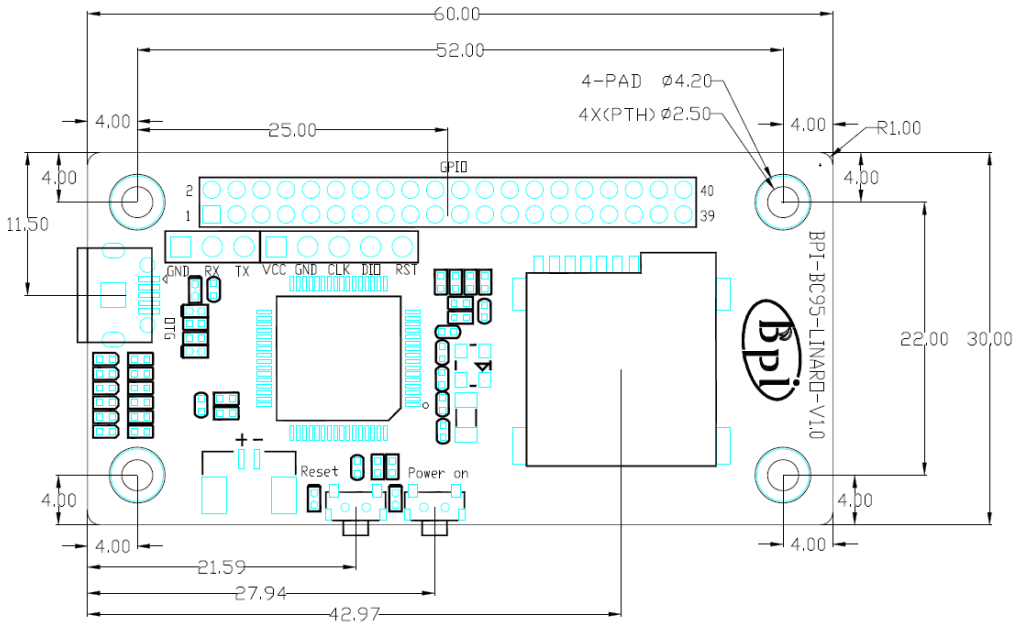
Soc	STM32F103RBT6
CPU	ARM 32-bit Cortex™-M3 CPU @72MHz
SDRAM size	20 KB
Flash size	128KB
Power	5V/2A via MicroUSB / 2PIN Battery Connector
Features	
Low-level peripherals	40 Pins Header (2.0mm Pitch), 32×GPIOs, Some of which can be used for specific functions including UART, I2C, SPI, PWM
On board Network	Quectel BC95
On board SIM	MicroSD SIM Slot eSIM(option)
USB	1 USB 2.0 host
Buttons	Reset button, User button
Leds	1 Power status Led and 5 other Leds
Sizes	60 mm × 30mm
Weight	10g

Pictures of the board:





Dimensions:



GPIO specification

Banana Pi 40-pin GPIO(2.0mm pitch)

Following is the Banana Pi GPIO Pin-out:

BPI-BC95 40PIN GPIO (CON1)

GPIO Pin Name	Default Function	Function2：GPIO
CON1-P01	GND	
CON1-P02	GND	
CON1-P03	MCU_USART2_RTS	PA1
CON1-P04	GPIO_PC6	PC6
CON1-P05	MCU_USART2_RX	PA3
CON1-P06	RST_BTN	
CON1-P07	MCU_USART2_TX	PA2
CON1-P08	SPI1_SCK	PA5
CON1-P09	MCU_USART2_CTS	PA0
CON1-P10	SPI1_MISO	PA6
CON1-P11	GPIO_PB8	PB8
CON1-P12	SPI1_NSS	PA4
CON1-P13	GPIO_PB9	PB9
CON1-P14	SPI1_MOSI	PA7
CON1-P15	I2C1_SCL	PB6
CON1-P16	GPIO_PC13	PC13
CON1-P17	I2C1_SDA	PB7
CON1-P18	GPIO_PA15	PA15
CON1-P19	I2C2_SCL	PB10
CON1-P20	GPIO_PB0	PB0
CON1-P21	I2C2_SDA	PB11
CON1-P22	GPIO_PB1	PB1
CON1-P23	USART1_CK	PA8
CON1-P24	SPI2_NSS	PB12
CON1-P25	USART1_TX	PA9
CON1-P26	SPI2_SCK	PB13
CON1-P27	USART1_RX	PA10
CON1-P28	SPI2_MISO	PB14
CON1-P29	USART5_CK	PB5
CON1-P30	SPI2_MOSI	PB15
CON1-P31	USART5_TX	PB3
CON1-P32	USART4_RX	PC11
CON1-P33	USART5_RX	PB4
CON1-P34	USART4_TX	PC10
CON1-P35	VCC_1.8V	
CON1-P36	NC	

CON1-P37	VCC_5V	
CON1-P38	NC	
CON1-P39	GND	
CON1-P40	GND	

SWD(CN1)

CSI Pin Name	**Default Function**	**Function2：GPIO**
CN1-P1	VCC	
CN1-P2	GND	
CN1-P3	SWCLK	PA14
CN1-P4	SWDIO	PA13
CN1-P5	RESET-ST	

Debug UART(CON2)

CON2-P1	GND	
CON2-P2	USART1_RX	PA9
CON2-P3	USART1_TX	PA10

二 Software introduction

1. Lite OS brief

Huawei LiteOS is a lightweight open-source IoT OS and a smart hardware development platform. It simplifies IoT device development and device connectivity, makes services smarter, delivers superb user experience, and provides better data protection. Huawei LiteOS is designed for smart homes, wearables, IoV, and intelligent manufacturing applications.

For more about Lite OS please refer to: *http://developer.huawei.com/ict/en/site-iot/product/liteos*

IDE：recommended development environment is MDK521 + STM32F1xxx library

2. BC95 software brief:

Command format:

Test Command	**AT+<cmd>=?**	Check possible sub-parameter values
Read Command	**AT+<cmd>?**	Check current sub-parameter values
Set Command	**AT+<cmd>=p1[,p2[,p3[.....]]]**	Set command
Execution Command	**AT+<cmd>**	Execution command

Searching network steps:

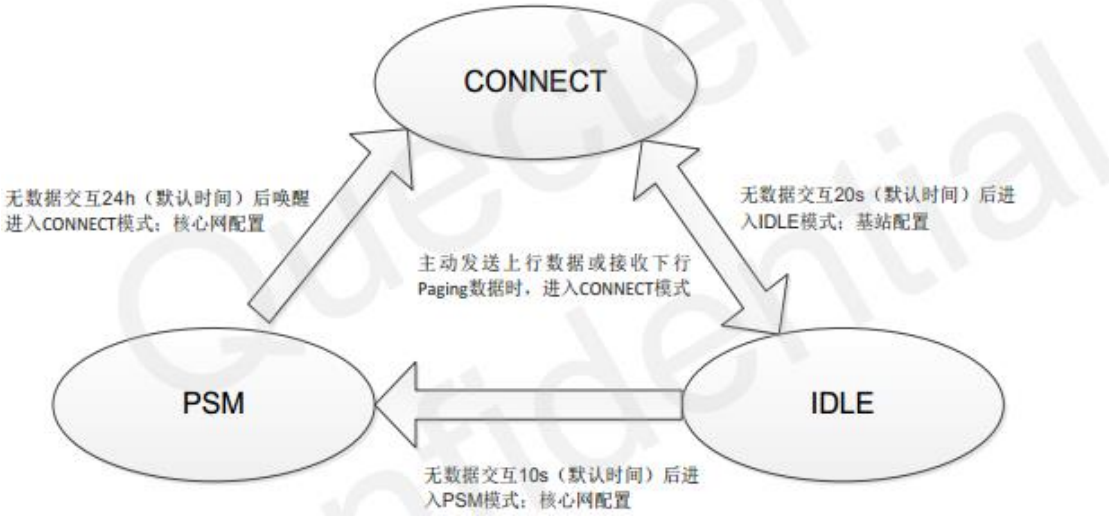
```
AT+NBAND?          //查询 Band
AT+CFUN?           //值为 1
AT+CIMI            //查询 IMSI 号
AT+CSQ             //查询信号强度
AT+NUESTATS        //查询模块状态
AT+CGATT?          //返回+CGATT:1 表示附着成功，有时延约 30s
AT+CEREG?          //查寻注网状态，1 为注册上网络，2 为正在找网
AT+CSCON?          //查询连接状态，1 为 CONNECT，0 为 IDLE
```

Manually configure network if fails to register network automatically:

```
AT+CFUN=1
AT+CIMI                    //执行 CFUN=1，等待 4 秒后查询 IMSI，如果能查到表示卡已识别；
                           若查不到，请检查卡是否插好并确认是否是 USIM 卡。
AT+NBAND?                  //查询频段信息。
AT+CEREG=1                 //设置自动上报网络注册状态，当模块注册上网络，会上报 URC。
AT+CGDCONT=1,"IP","APN"    //APN 为本地入网方式，自行配置（也可不配置）。
AT+COPS=1,2,"46000"        //指定 PLMN 搜索，PLMN 自行配置。
AT+CSQ                     //查询信号强度。
AT+NUESTATS                //查询模块状态。
AT+CGATT?                  //返回+CGATT:1 表示附着成功，有时会有约 30s 的延迟。
AT+CEREG?                  //查寻注网状态，1 为注册上网络，2 为正在找网。
AT+CSCON?                  //查询连接状态，1 为 CONNECT，0 为 IDLE。
```

Network connectivity status chart:

：1. Connect 状态（+CSCON:0,1，模块注网后即处于该状态），该状态持续的时间由基站配置，由不定时活动器来控制，范围为 1-3600s，默认 20s。
2. Idle 状态（ +CSCON:0,0），该状态持续的时间由核心网配置，由 Active timer（T3324）来控制，范围为 0-11160s，默认 10s。
3. PSM 状态（可通过功耗判断，最大功耗 5uA），该状态持续的时间由核心网配置，由 TAU 定时器（T3412）来控制，范围为 0h-310h，默认 24h。



三 How to setup and start to use

1. Hardware preparation:

BPI OPEN DEBUGGER or J-Link DEBUGGER *1

BPI BC95-Linaro development board *1

Micro USB cable *2

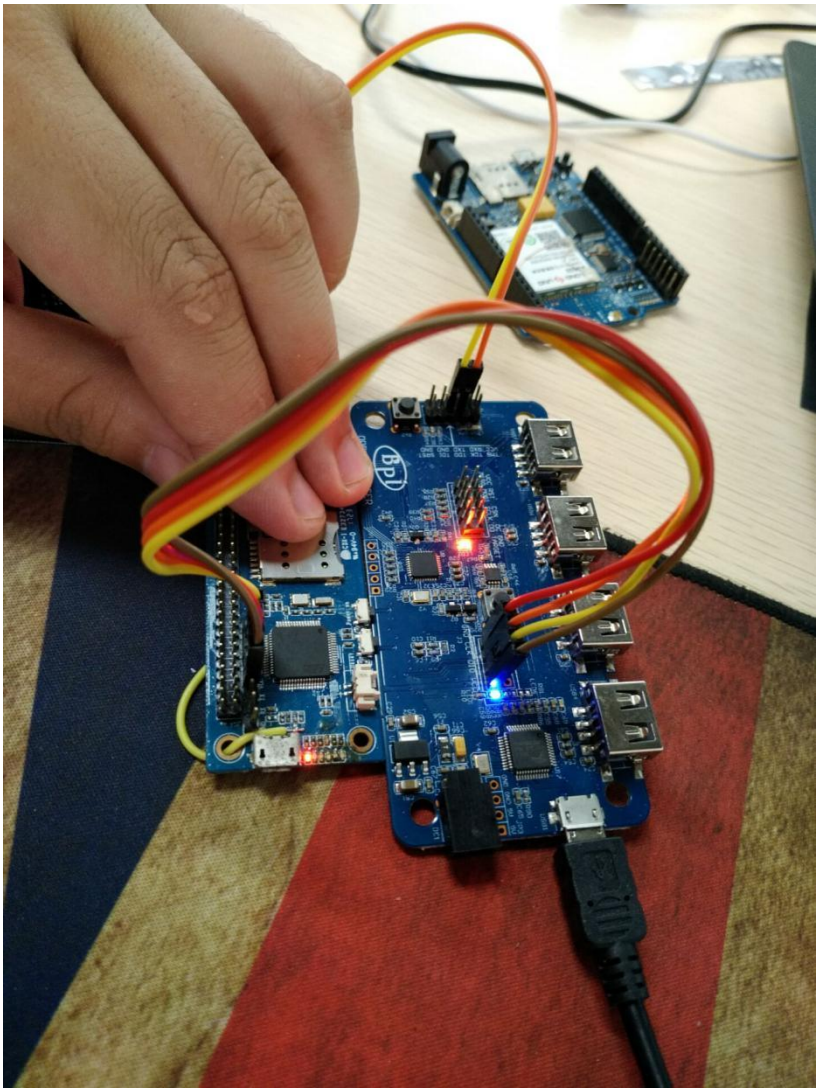
Dupont lines (female female) *4

2. Hardware hookup

There are two ways to flash by either BPI OPEN DEBUGGER or J-Link DEBUGGER:

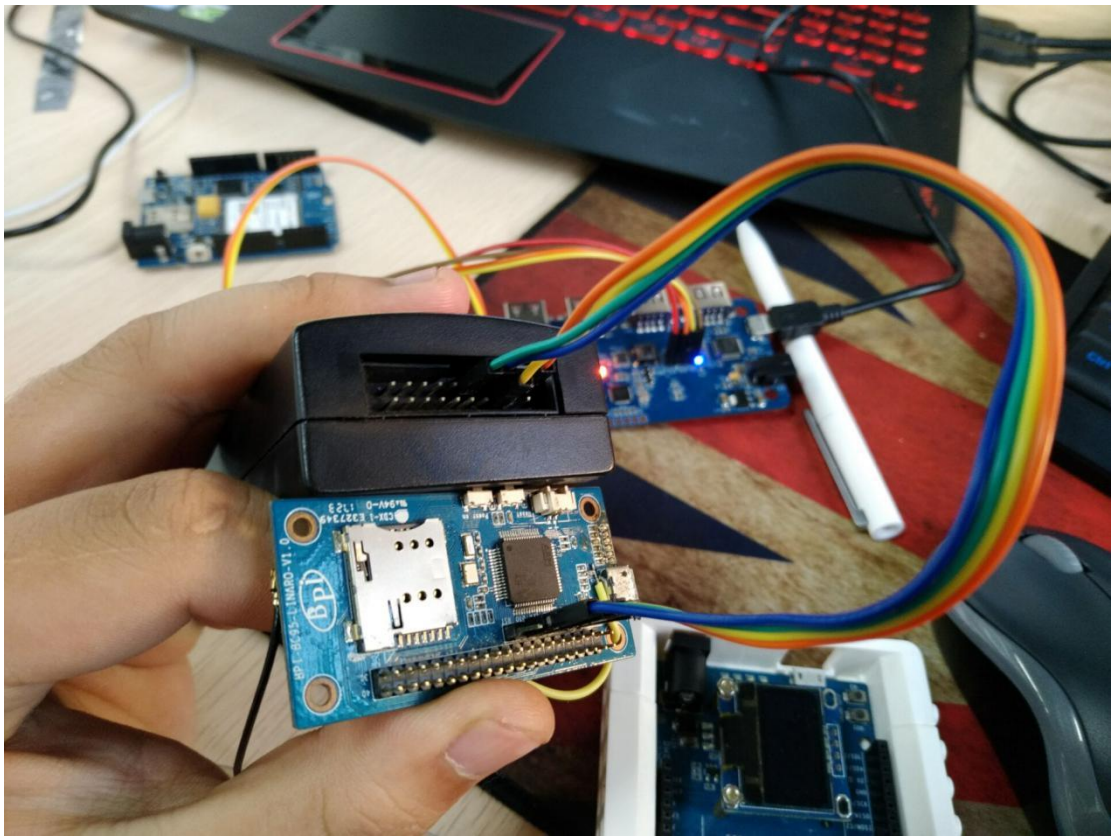
1) BPI OPEN DEBUGGER：

Using Dupont lines to connect BPI OPEN DEBUGGER with BPI BC95-Linaro SWD port，the connection are below:



BPI OPEN DEBUGGER	BPI NB-IoT Linaro SWD
3V3	VCC
GND	GND
DIO	DIO
CLK	CLK

2) J-Link DEBUGGER：



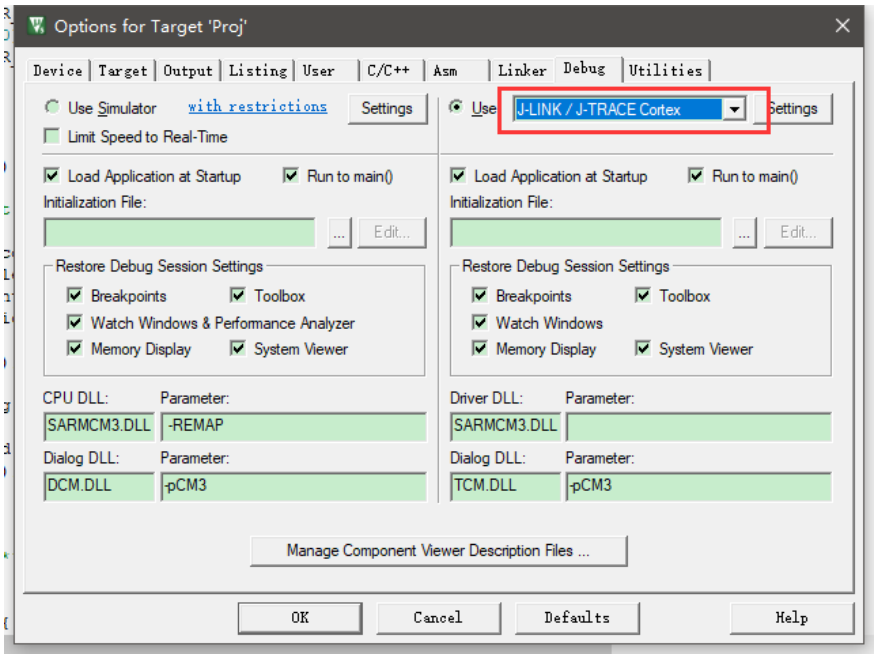
The corresponding connection between JLink and BPI BC95 are the following:

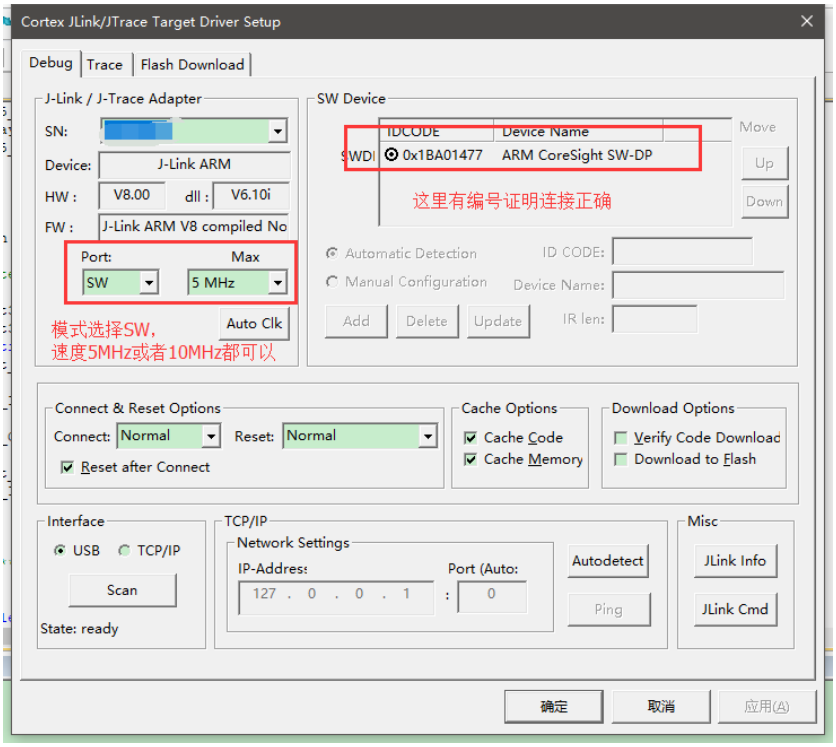
J-link	BPI NB-IoT Linaro SWD
3V3 (Pin1)	VCC
GND (Pin4.6.8.10……)	GND
SWDIO (Pin7)	DIO
SWCLK (Pin9)	CLK
nJTRST （Pin3）optional	RST （optional）

3. Software debugging setting:

MDK5 setting:

Create a project choosing the right chip,and set the Options of Target as follows:





4. Software sample cases:

There are two samples in BPI forum, please refer to the following links:

1. http://forum.banana-pi.org.cn/thread-2200-1-1.html
2. http://forum.banana-pi.org.cn/thread-2215-1-1.html

And there are some more information about the development board in the below link:

http://forum.banana-pi.org.cn/forum-121-1.html

5. Github code :

https://github.com/yelvlab/BPI_NB-IoT_Linaro_96Boards