Tutorial for DSI2599

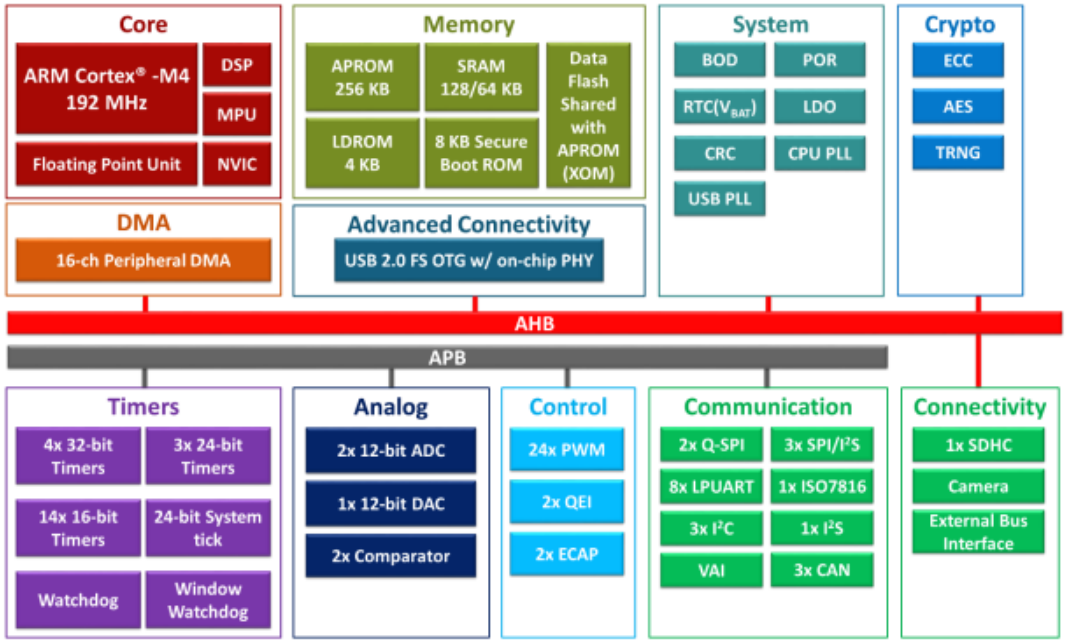
NuMaker-IoT-M487

16 Oct 2020

Danny Chou

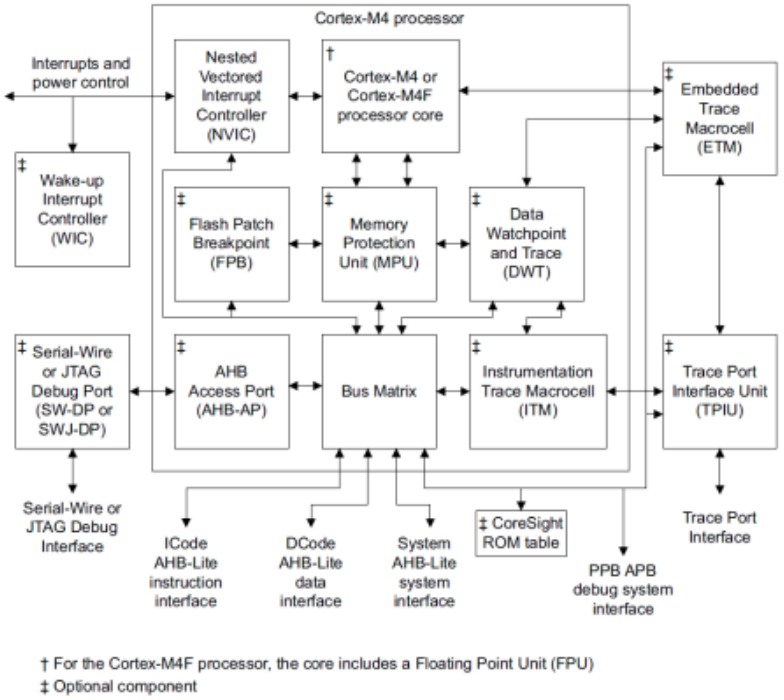
Microcontroller and microprocessor

微控制器(Microcontroller, MCU)



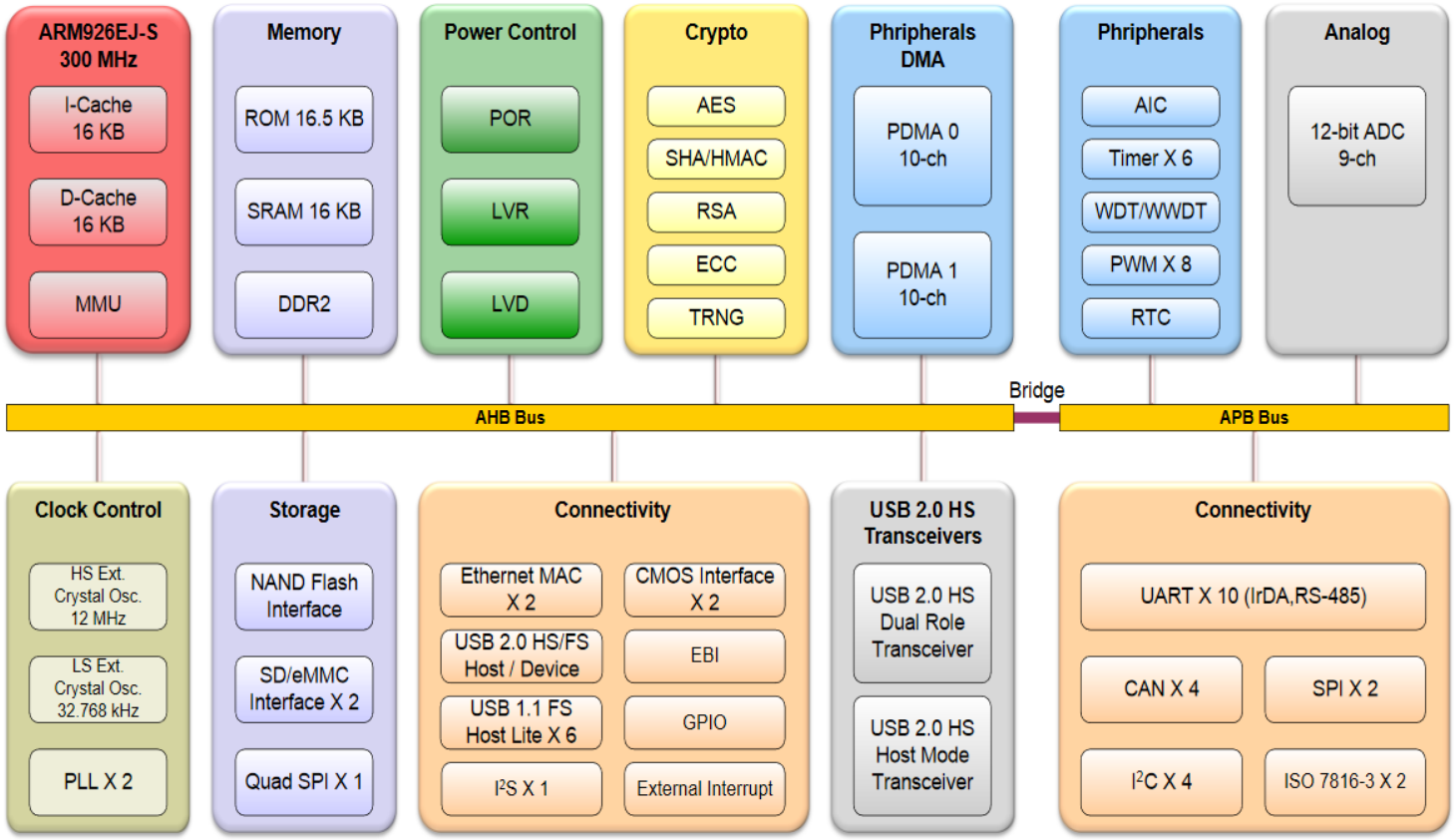
Nuvoton M480 series

微處理器(Microprocessor, MPU)



ARM Cortex-M4 processor

The modern microprocessor

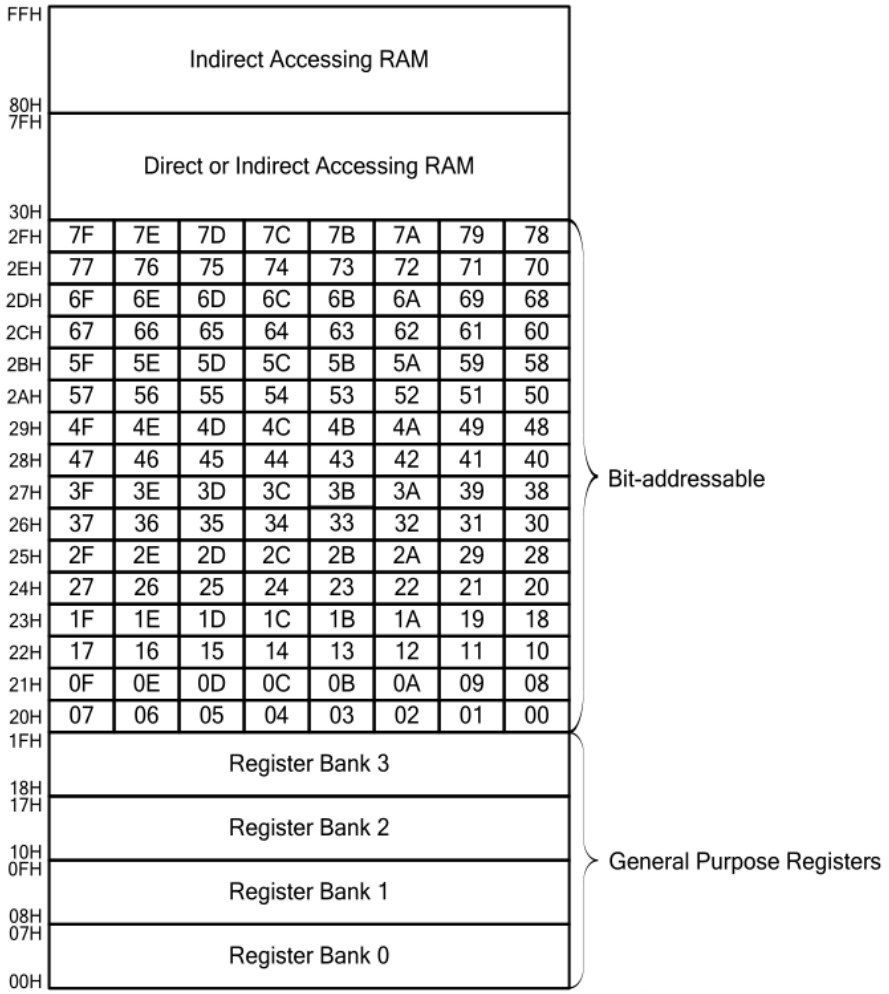


Nuvoton NUC980 series microprocessor

Memory

- Program memory / code memory
  - Flash, EEROM, PROM etc.
  - LDROM, APROM, SPROM

- Data memory
  - SRAM or DRAM
  - IRAM (Internal RAM), XRAM (External RAM)

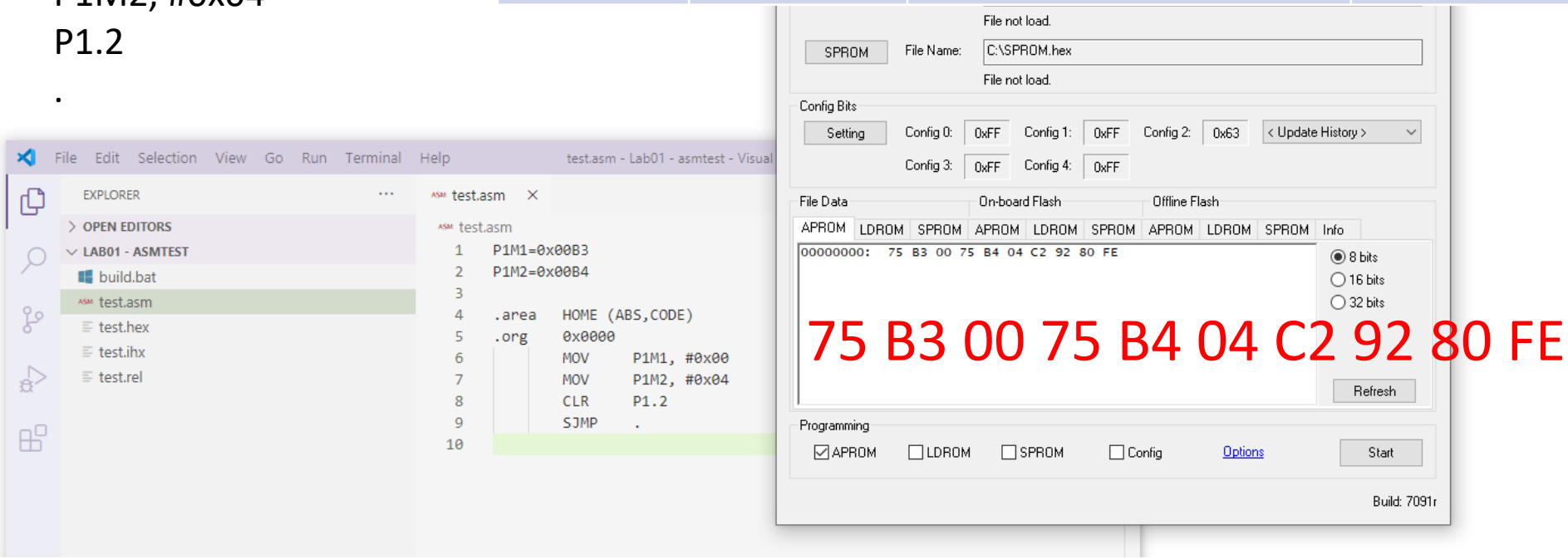


Instructions

```
P1M1=0x00B3
P1M2=0x00B4

.area    HOME (ABS,CODE)
.org     0x0000
MOV      P1M1, #0x00
MOV      P1M2, #0x04
CLR      P1.2
SJMP     .
```

Hex code	Mnemonic	Operands	Number of bytes
0x75	MOV	data address, data value	3
0xC2	CLR	bit address	2
0x80	SJMP	relative code address	2



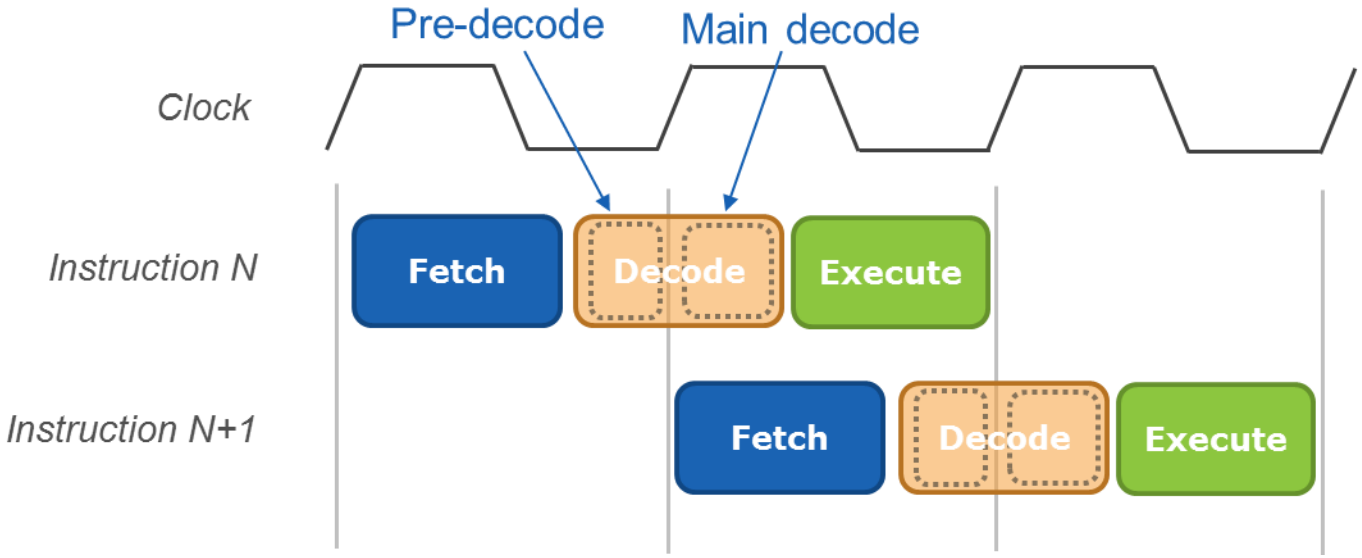
Instruction cycle

- Fetch -> Decode -> Execute -> Restore
- Fetch: Get instruction from program memory to instruction register.
- Decode: Resolve the instruction.
- Execute: Execute the instruction, get operands from program memory if needed.
- Restore: Result will store in specific register.

24	2	ADD	A,#data
25	2	ADD	A,data addr
26	1	ADD	A,@R0
27	1	ADD	A,@R1
28	1	ADD	A,R0
29	1	ADD	A,R1
2A	1	ADD	A,R2
2B	1	ADD	A,R3
2C	1	ADD	A,R4
2D	1	ADD	A,R5
2E	1	ADD	A,R6
2F	1	ADD	A,R7

Pipeline

- Multiple instruction shift register



https://microchipdeveloper.com/32arm:m0-pipeline

Hardware acceleration

- FPU : Float Point Unit
- DSP : Digital Signal Unit
- IPU : Image Processing Unit
- GPU : Graphics Processing Unit

From C language to executable binary

- Compile -> Assemble -> Link
- Compile: From C source code to assembly (with flag).
- Assemble: From assembly to binary code with flag (objects).
- Link: Link multiple objects together output an executable binary.

```c
#include "MS51_16K.h"


void main (void)
{
/* UART0 initial setting
  * include sys.c in Library for modify HIRC valu
  * include uart.c in Library for UART initial se
*/
    MODIFY_HIRC(HIRC_24);
    P06_PUSHPULL_MODE;
    UART_Open(24000000,UART0_Timer3,115200);
    ENABLE_UART0_PRINTF;

    printf("\n Hello world!");
    while(1);


}
```

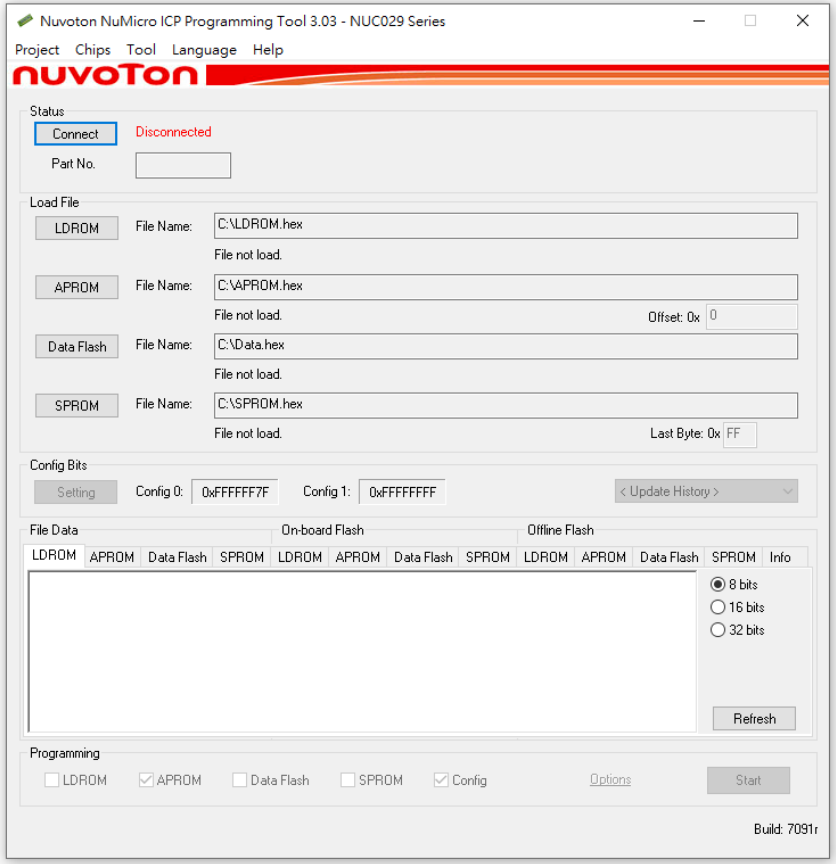
```
assignBit
clr _EA
mov _TA,#0xaa
mov _TA,#0x55
mov _SFRS,#0x00
assignBit
mov c,_BIT_TMP
mov _EA,c
anl _P0M1,#0xbf
orl _P0M2,#0x40
..\..\helloworld.c:28: UART_Open(24000000,UART0_
mov _UART_Open_PARM_2,#0x01
clr a
mov _UART_Open_PARM_3,a
mov (_UART_Open_PARM_3 + 1),#0xc2
mov (_UART_Open_PARM_3 + 2),#0x01
mov (_UART_Open_PARM_3 + 3),a
mov dptr,#0x3600
mov b,#0x6e
mov a,#0x01
```

```
A CONST size F flags 20 addr 0
A XINIT size 0 flags 20 addr 0
A CABS size 0 flags 28 addr 0
T 00 00 00
R 00 00 00 02
T 00 00 00
R 00 00 00 03
T 00 00 00
R 00 00 00 04
T 00 00 00
R 00 00 00 06
T 00 00 00
R 00 00 00 06
T 00 00 00
R 00 00 00 0E
T 00 00 00 02 00 00
R 00 00 00 0E 02 04 00 E4
T 00 00 00 02 00 03
R 00 00 00 16 00 04 00 0E
T 00 00 03
```

```
:1300A600740EC0E07480C0E012055F15811581158180FED
:0F0E62000A2048656C6C6F20776F726C642100FA
:1000B900AE82AF8310990280FB8E998E828F832244
:2000C900AF82A2AF9200C2AF75C7AA75C755759100A2009
:2000E900080F800A75A638800875A630800375A630A2AF9
:20010900A20092AF75A70075AF04A2AF9200C2AF75C7AA7
:20012900A604F5A6A2AF9200C2AF75C7AA75C75543A401A
:20014900C7AA75C755539FFEA20092AFBF082E8E047F00E
:20016900EA2CFCEB3FFFEC24F1FCEF34FFFF8C0374015BF
:2001890075C7558E8475C7AA75C7558D85A2AF9200C2AF7
:2001A9005387EF22AF82BF0202800ABF03028011BF04538
:2001C90084758203120284A2AF9200C2AF75C7AA75C7555
:2001E900107582021202847582041202107582041202 84A
:2002090097DFA20092AF22AF82BF02028005BF043C8022A
:200229009720A20092AFAE967F004306DFBEDF1DBF001A8
:200249007F004306F7BEF705BF000280F122AF82759100B
:2002690000C2AF75C7AA75C7555397DFA20092AF2275C7A
:20028900A2AF9200C2AFBF0202800ABF0302802DBF04658
:2002A900555396FBA20092AFA2AF9200C2AF75C7AA75C75
:2002C90000C2AF75C7AA75C755439604A20092AFA2AF920
```

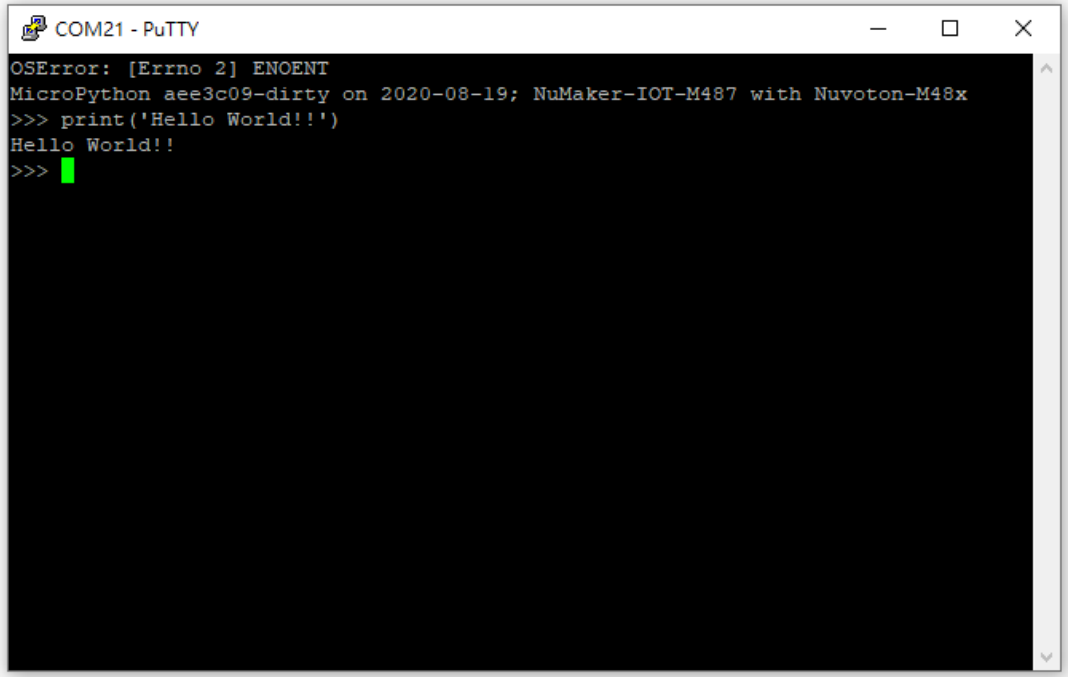
From C language to executable binary

- Dump the binary by using a programming tool.

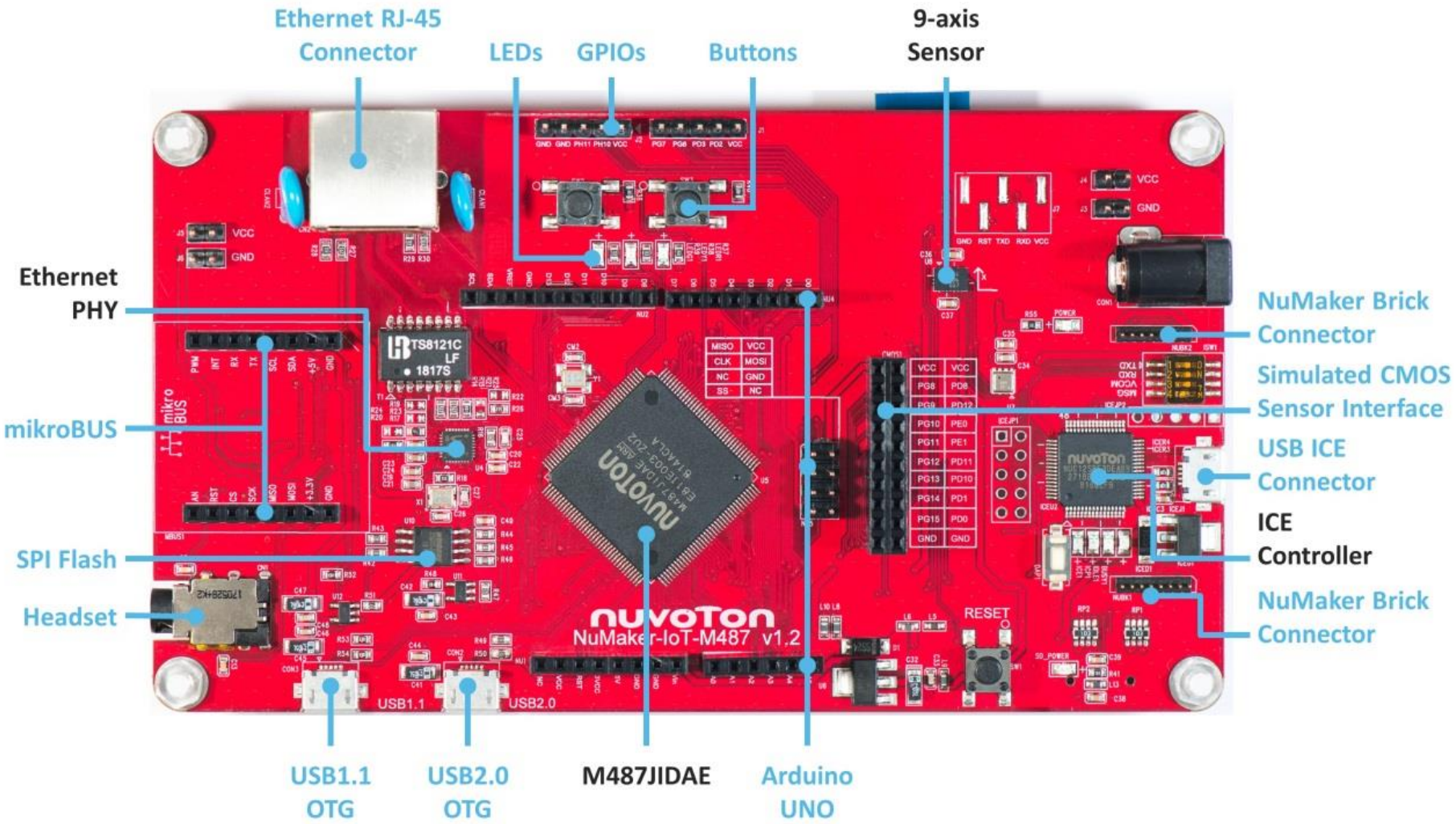


Python

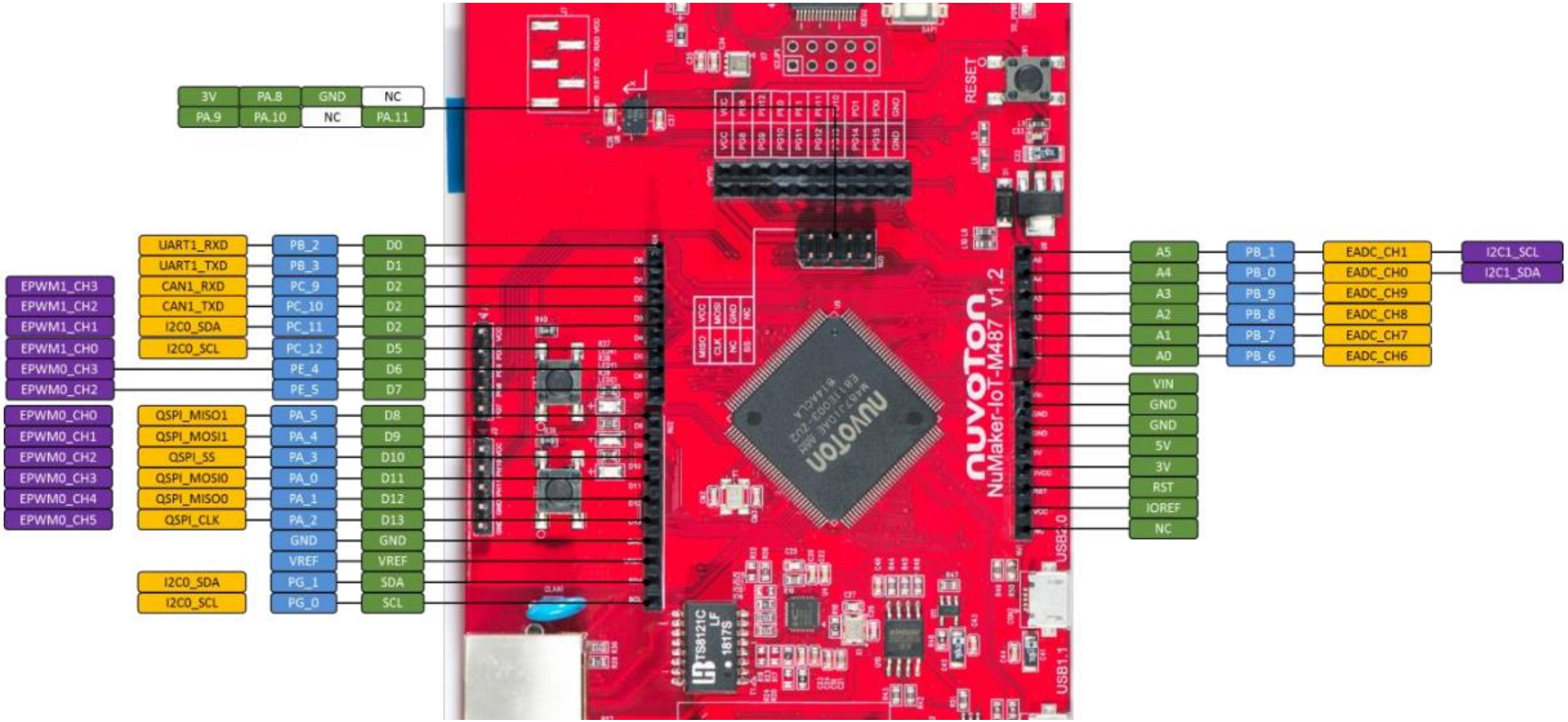
- Interpreted language, different from compiled language.
- User’s code does not need to be compile.



Introduction of DSI2599 (NuMaker-IoT-M487)

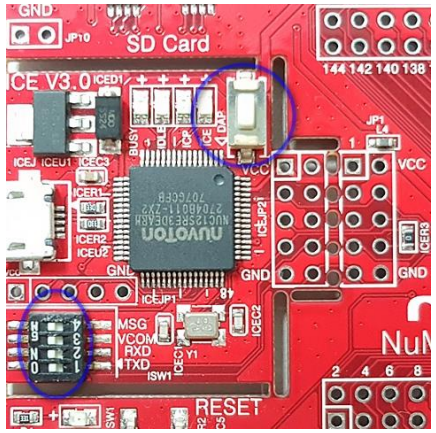


Introduction of DSI2599 (NuMaker-IoT-M487)

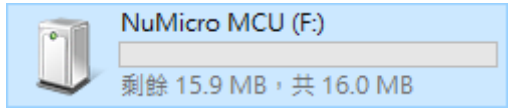


NuMicroPy

- Switch on the dip switch on Nu-Link programmer.

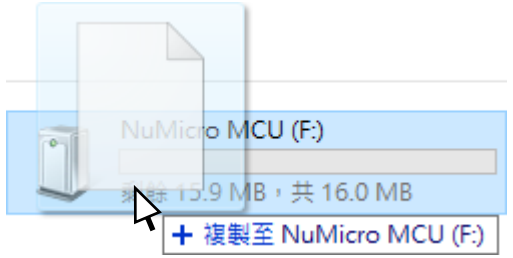


- Insert the programmer to your laptop.



NuMicroPy

- Download NuMicroPy firmware.
  - https://github.com/OpenNuvoton/NuMicroPy
- Flash the NuMicroPy firmware. Simply just copy the file into it.



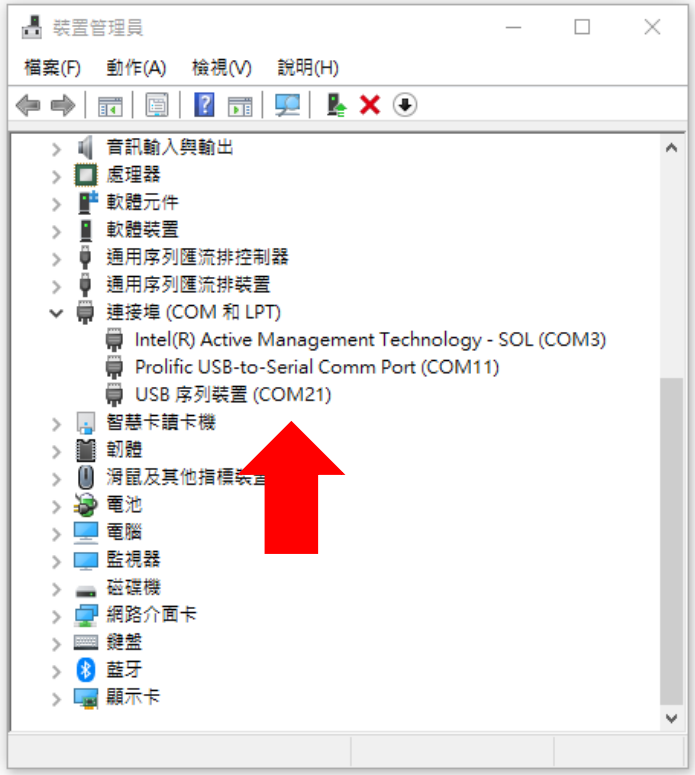
- The drive will disappear for a short time while flashing the firmware.
- Wait 5-10 seconds make sure process has been complete.

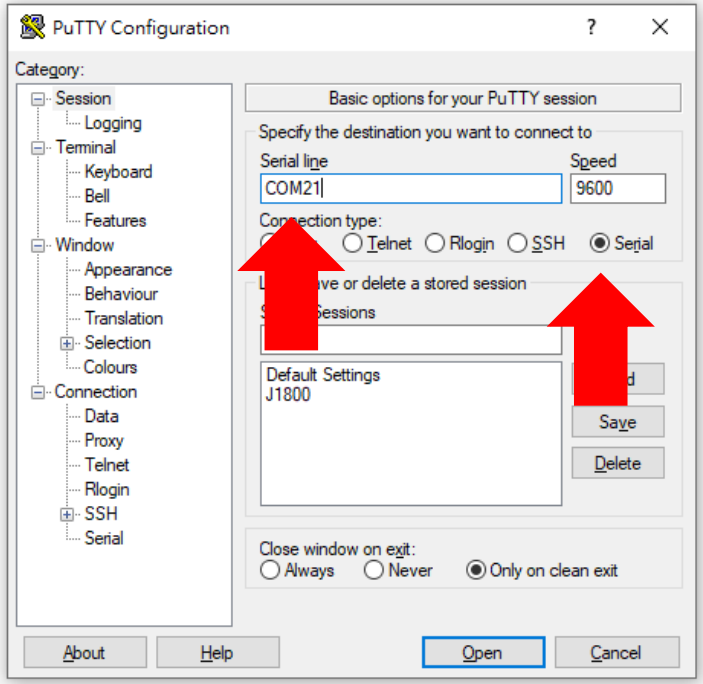
NuMicroPy

- Remove USB cable from Nu-Link programmer.
- Connect USB1.1 in the lower left corner to your laptop.

NuMicroPy

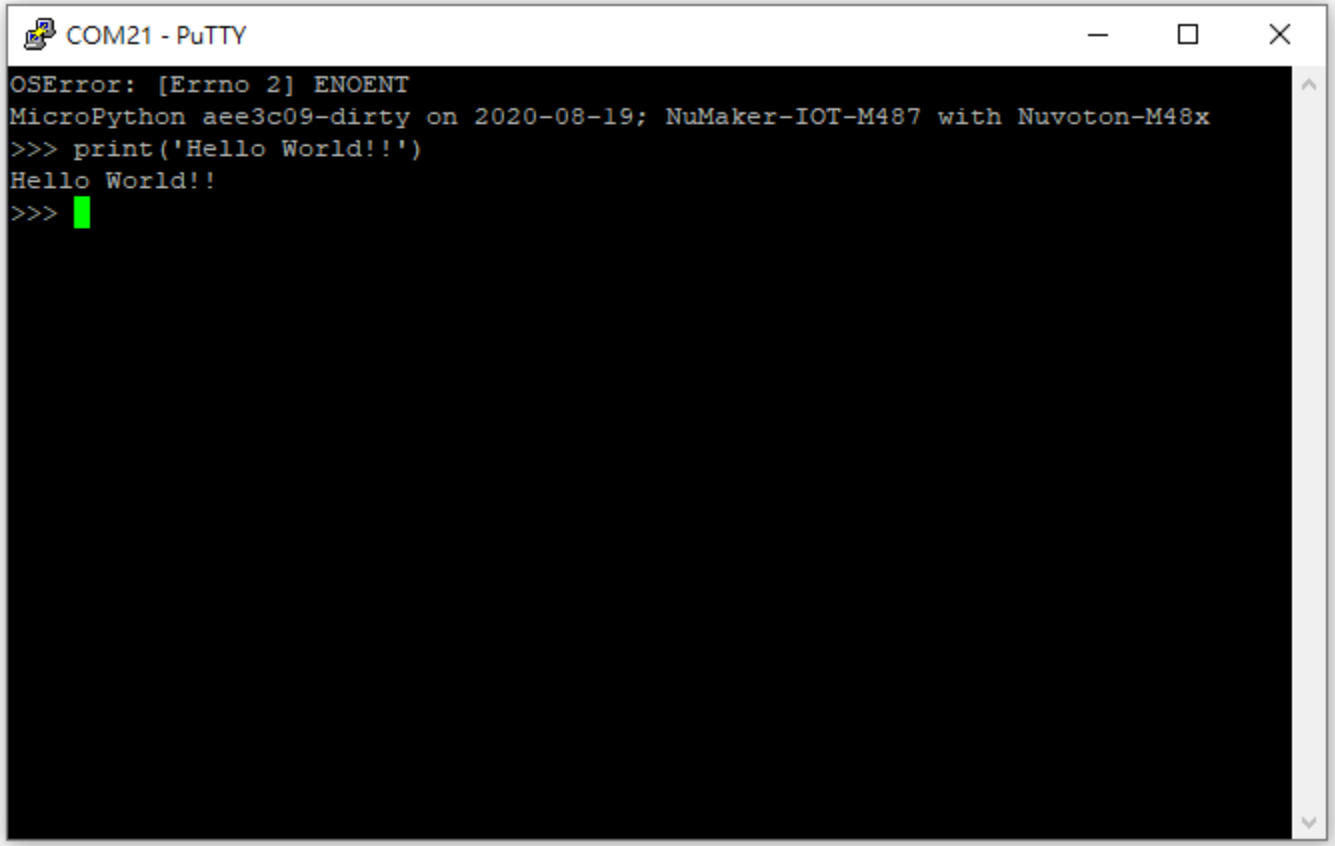
- Download PuTTY.
  - https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html





NuMicroPy

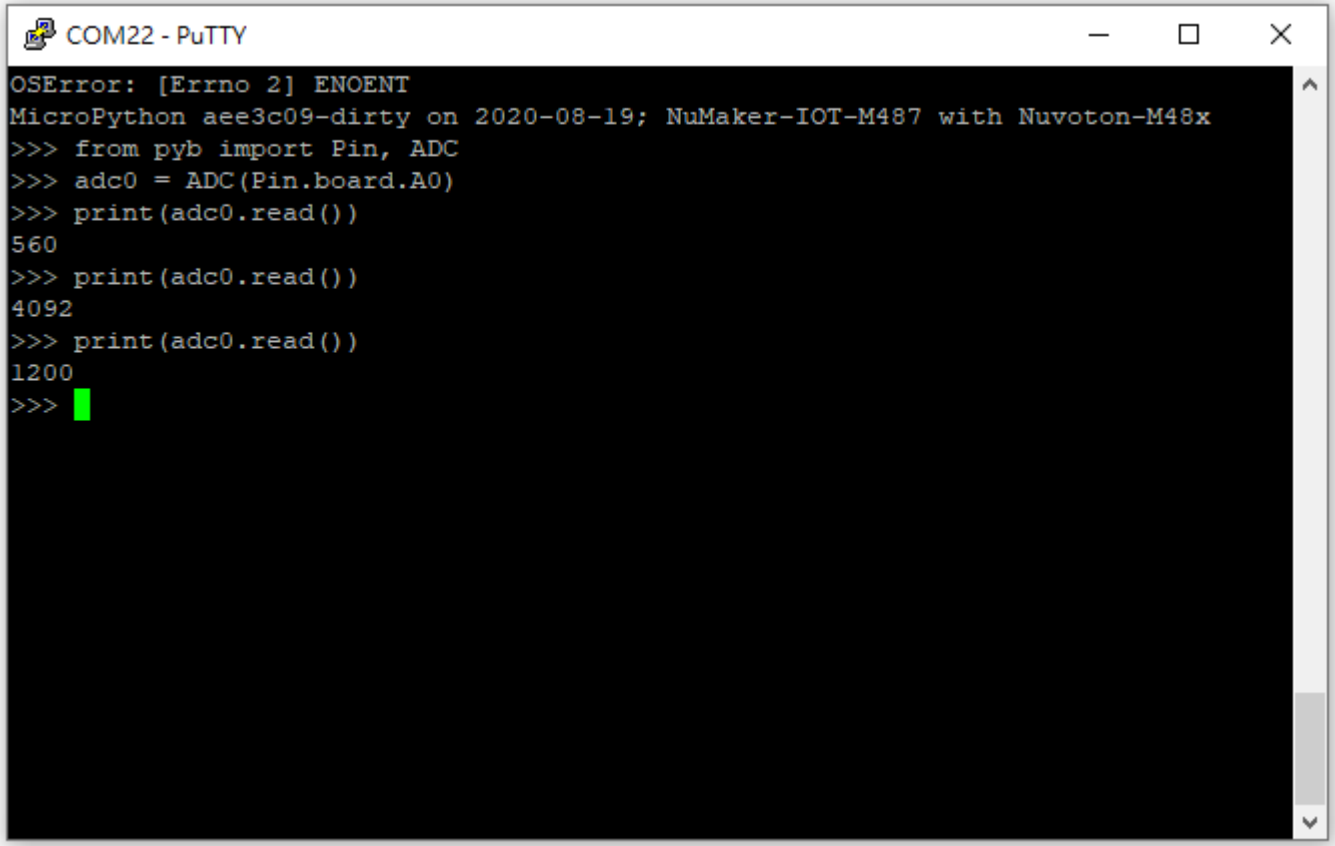
```
print(‘Hello World!!!’)
```



Exercise 1: ADC

```
from pyb import Pin, ADC

adc0 = ADC(Pin.board.A0)
print(adc0.read())
```



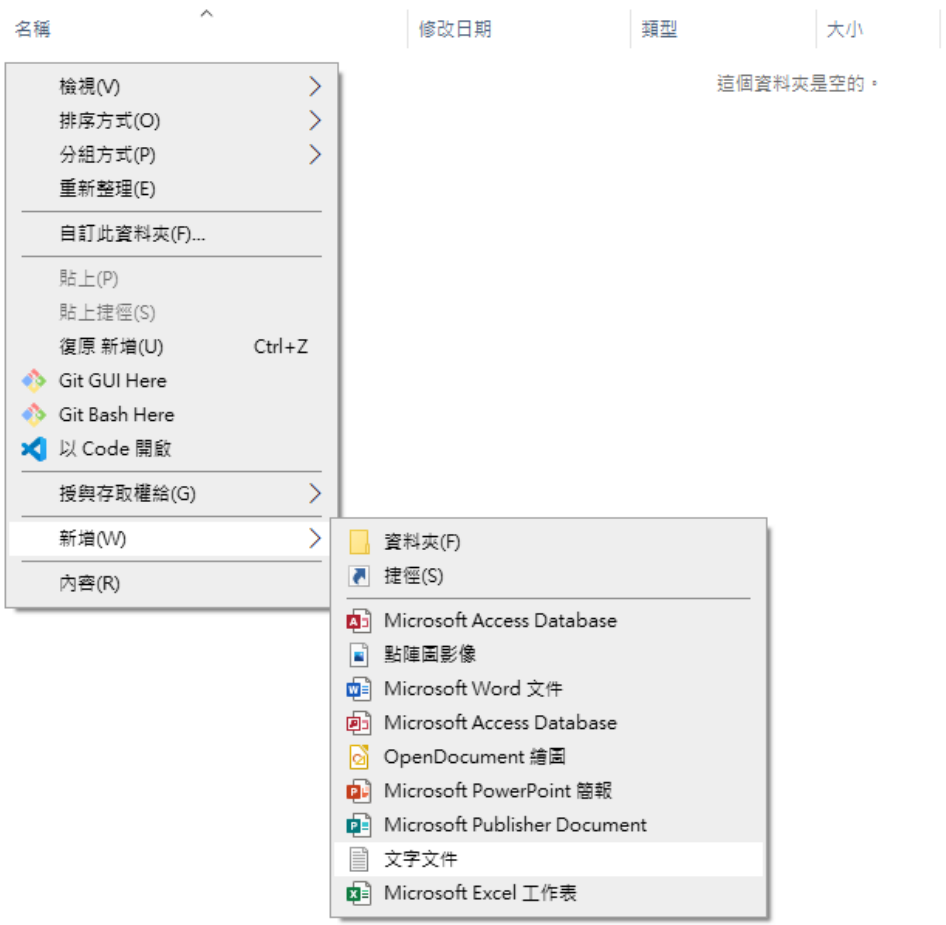
Exercise 2: Wi-Fi

```
import network

wlan = network.WLAN()
wlan.connect(‘YourSSID’, ’YourPasswd')
wlan.ifconfig()
```

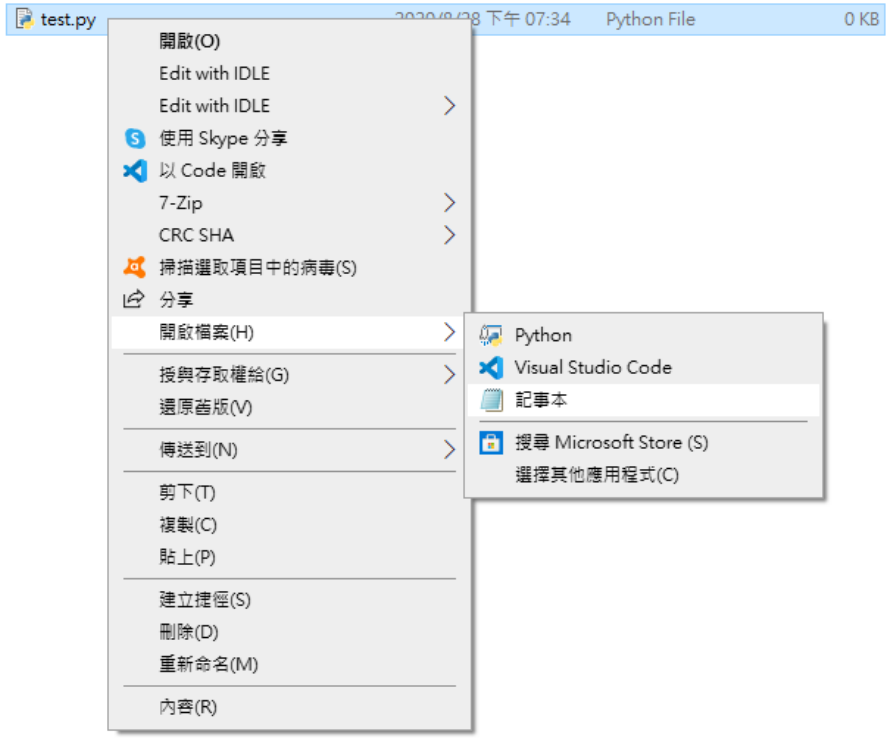
Exercise 3: Script

- Create a new text file, named “ledtest.py” in your file manager.



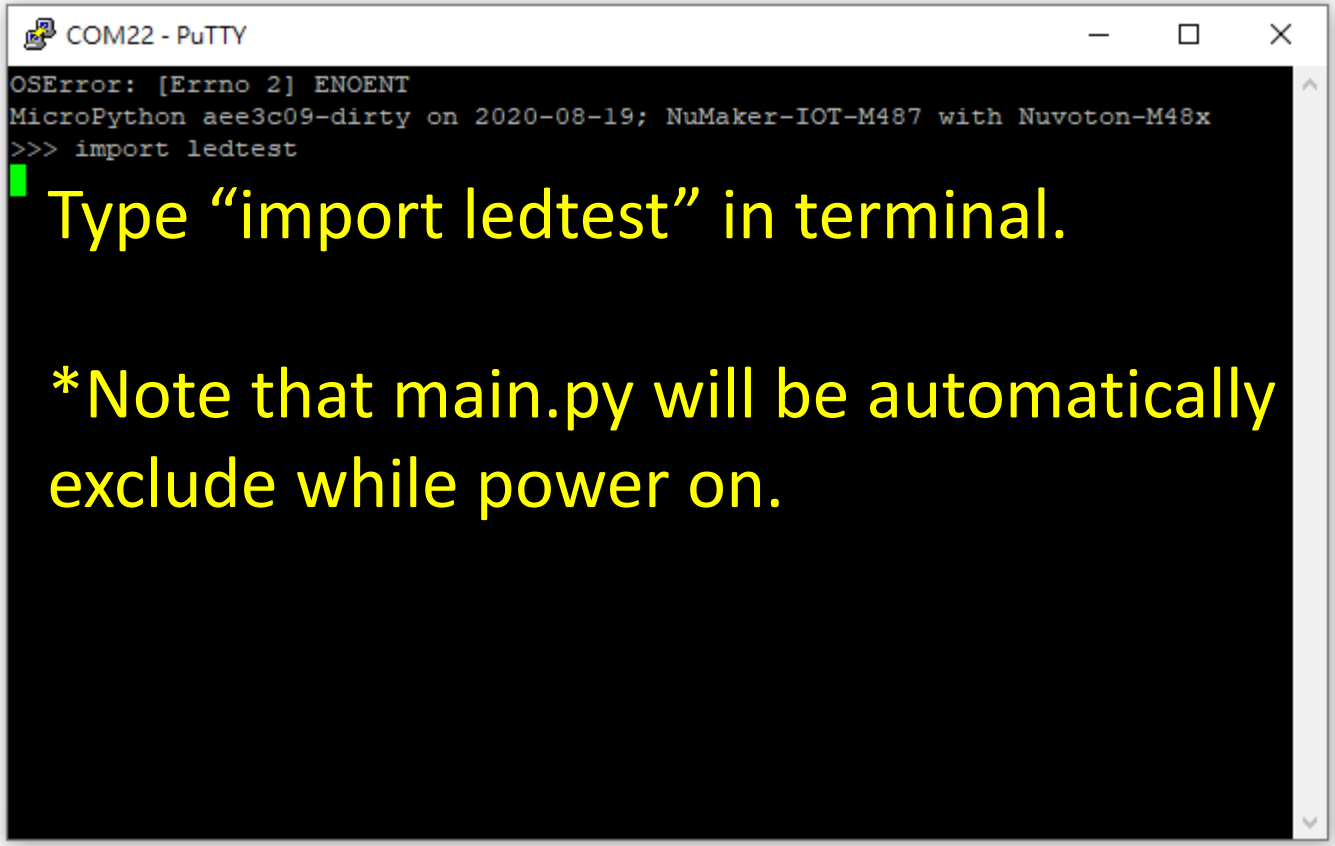
Exercise 3: Script

- Open the file with Notepad.
- Note the indent if using some others text editor.



Exercise 3: Script

```
from pyb import Switch, LED
sw = Switch('sw2')
led = LED('led0')
while True:
    if sw.value():
        led.on()
    else:
        led.off()
```

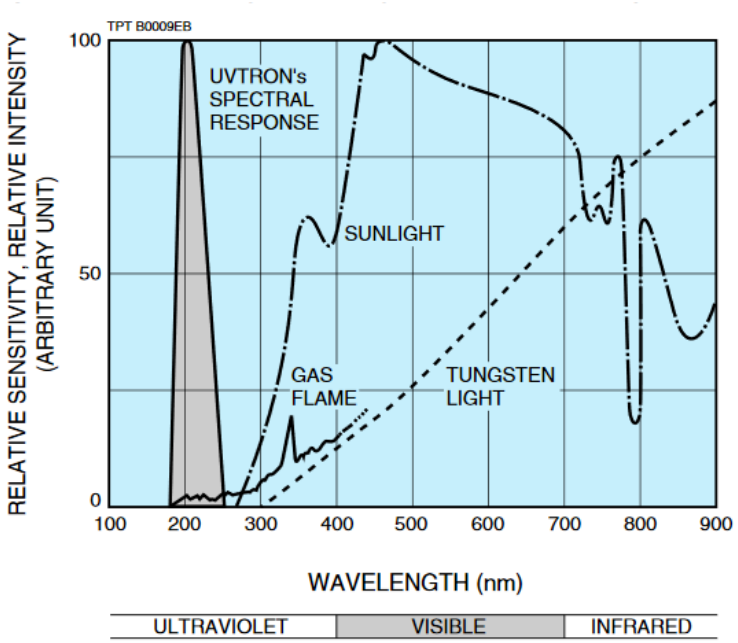


Type “import ledtest” in terminal.

*Note that main.py will be automatically exclude while power on.

Flame detector

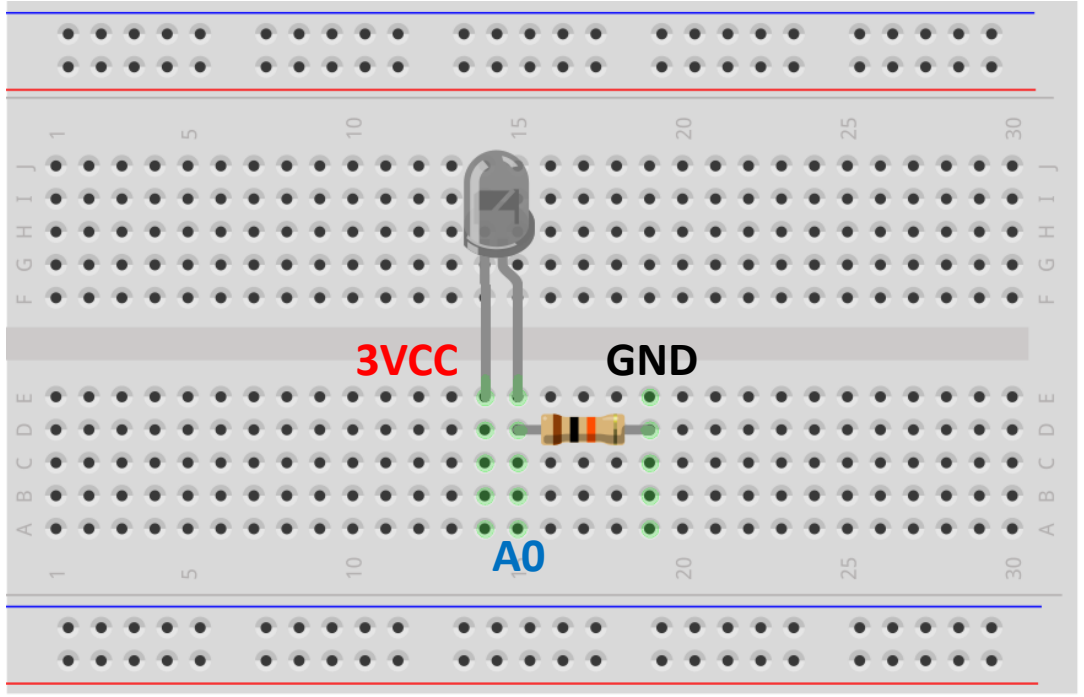
- Sensing the specific spectrum from the flame.
- Faster and more sensitive then smoke detector.



https://www.hamamatsu.com

Flame detector

- The cheaper solution: Using a photo diode with visible light filter.
- Note that the photo diode working at reverse bias.
- https://reurl.cc/7oMOv9



MCU selection guide

- System on a Chip
  - Interface: GPIO, UART, SPI, I^2C, I2S, SD, USB, CAN
  - Module: Timer, PWM, ADC, DAC, RTC
  - Storage: Data flash, EEPROM
- Performance and power consumption
  - Hardware acceleration
  - VBAT
- Reliability
  - EMI, EMF, EFT
  - AEC-Q100/101/102

MCU selection guide

- Standard 8051 / 6T 8051 / 4T 8051 / 1T 8051
- ARM Cortex-M0
- ARM Cortex-M23
- ARM Cortex-M23 with TrustZone
- ARM Cortex-M4 with FPU and DSP

MCU selection guide

- Cost effective: MS51, Mini51, N76E
- General purpose: NUC029, M031, M051
- USB: M032, NUC029, M452
- Low power: ML51, Nano100
- IoT security: M2351
- Performance: M480 Series, NUC505
- Automotive (CAN): NUC131, NUC130, M453, M483
- https://direct.nuvoton.com/

Q&A

Appendix 1: Resource

Resource

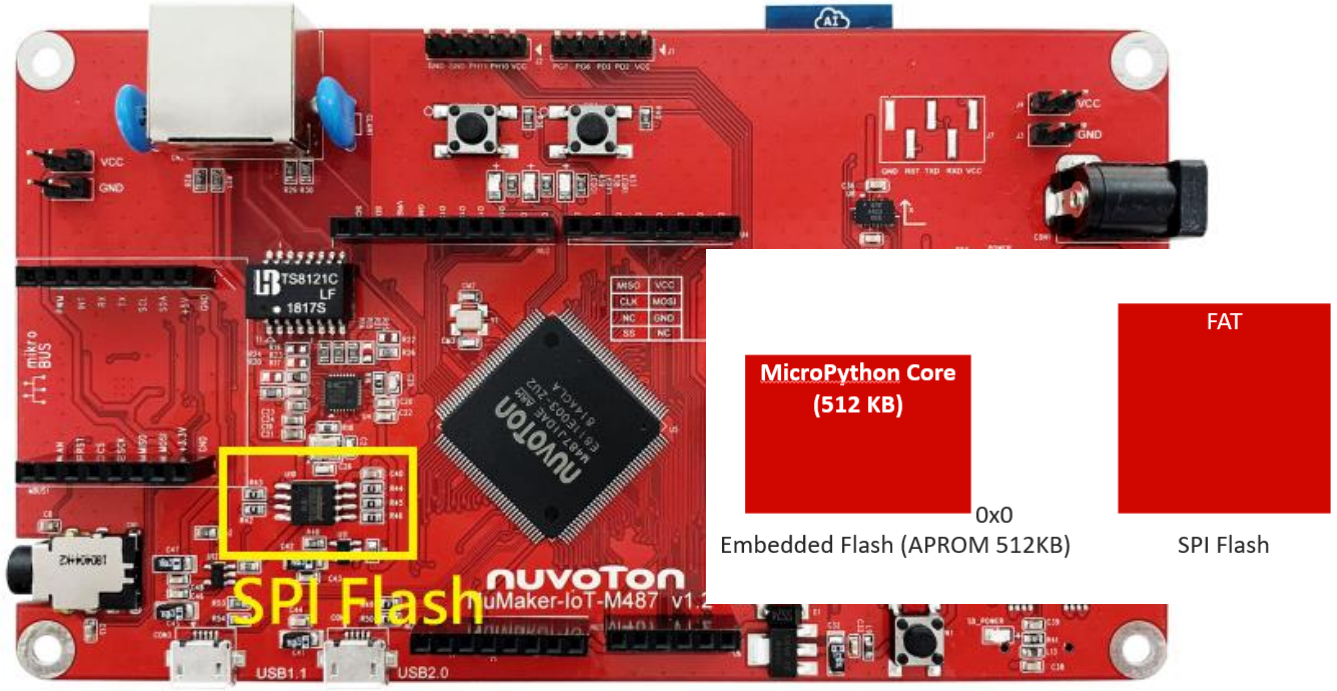
- NuMicroPy
- DSI2599 NuMaker-IoT-M487 User Manual
- IDEAS Chain
- Nuvoton Direct

Appendix 2: Factory Reset

While the board does not work properly even reflash the NuMicroPy firmware.

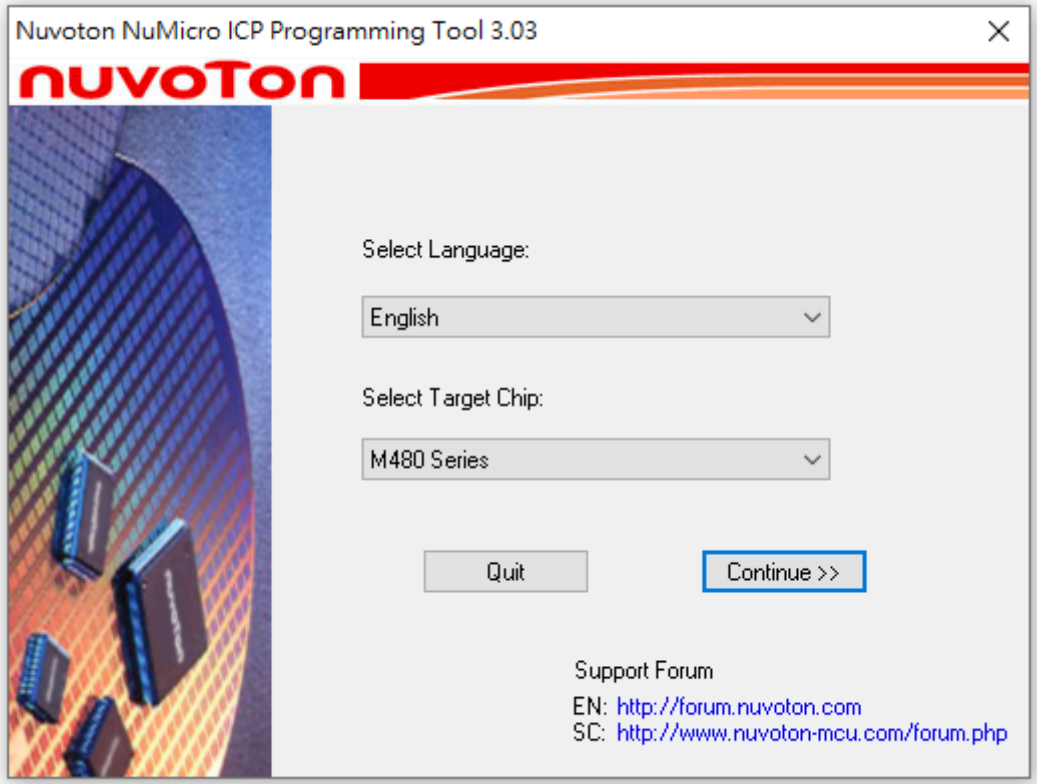
Factory reset

- Python script stored in the SPI Flash.

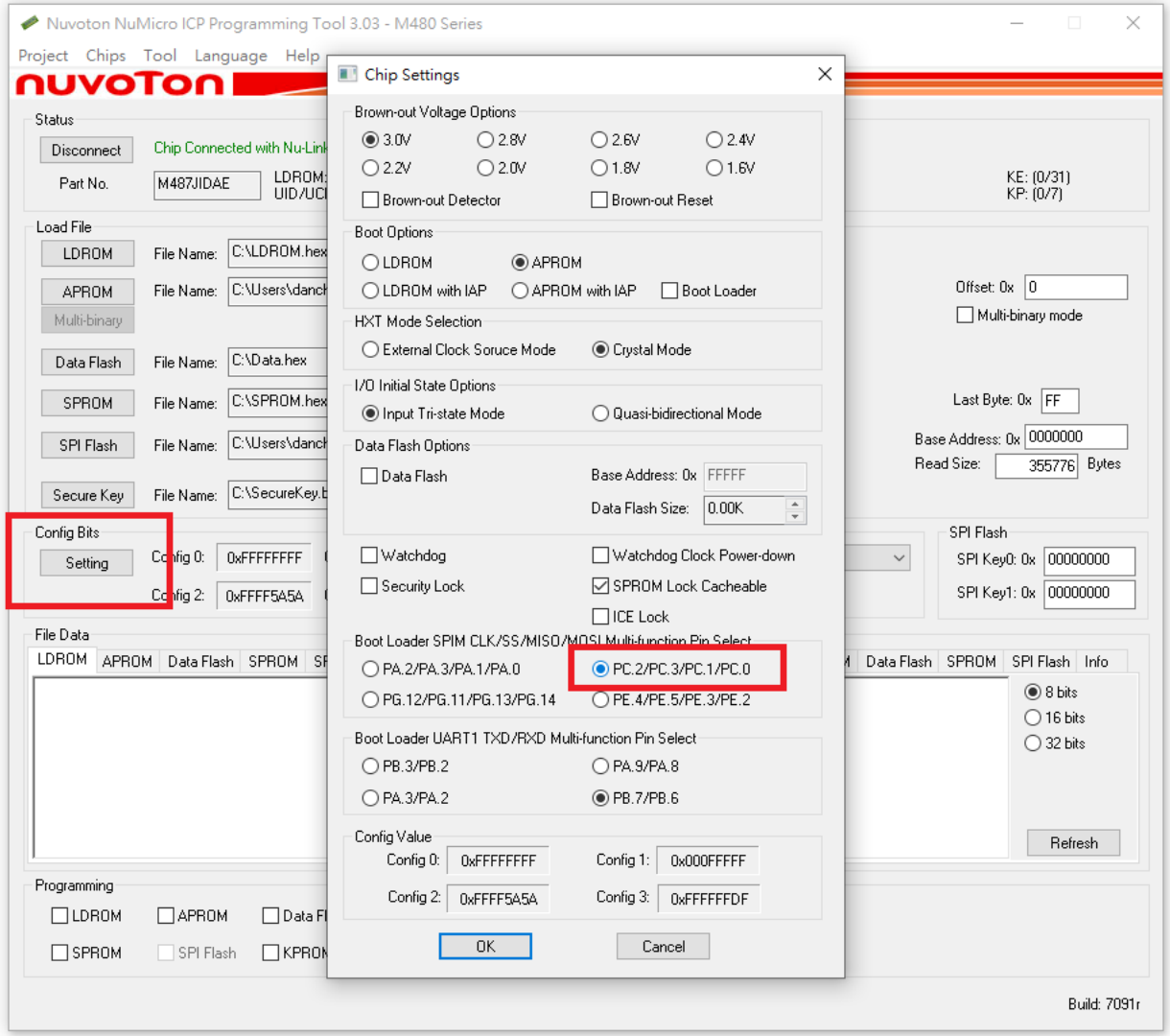


Factory reset

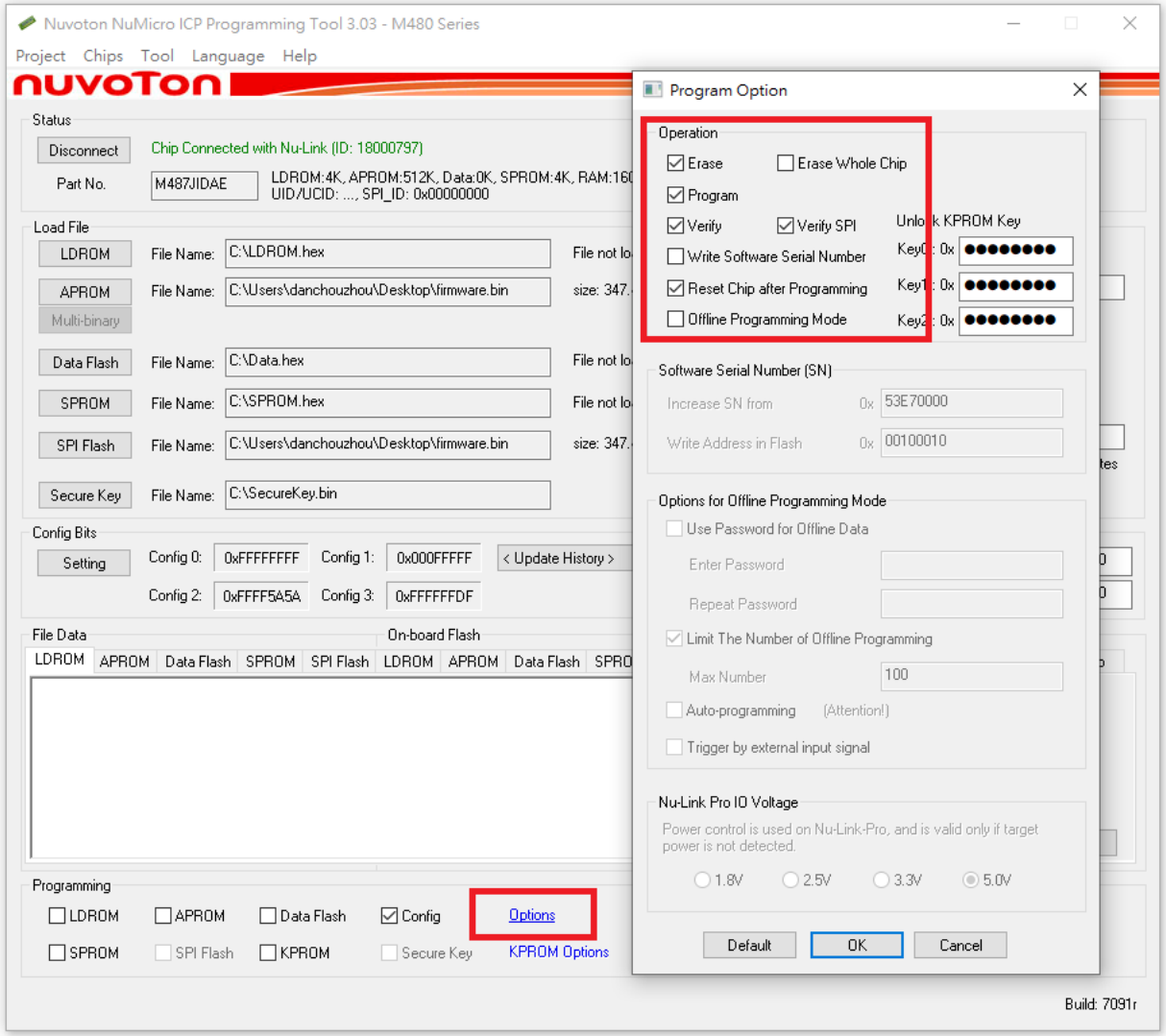
- Solution: Using Nuvoton ICP Programming Tool to access SPI Flash.



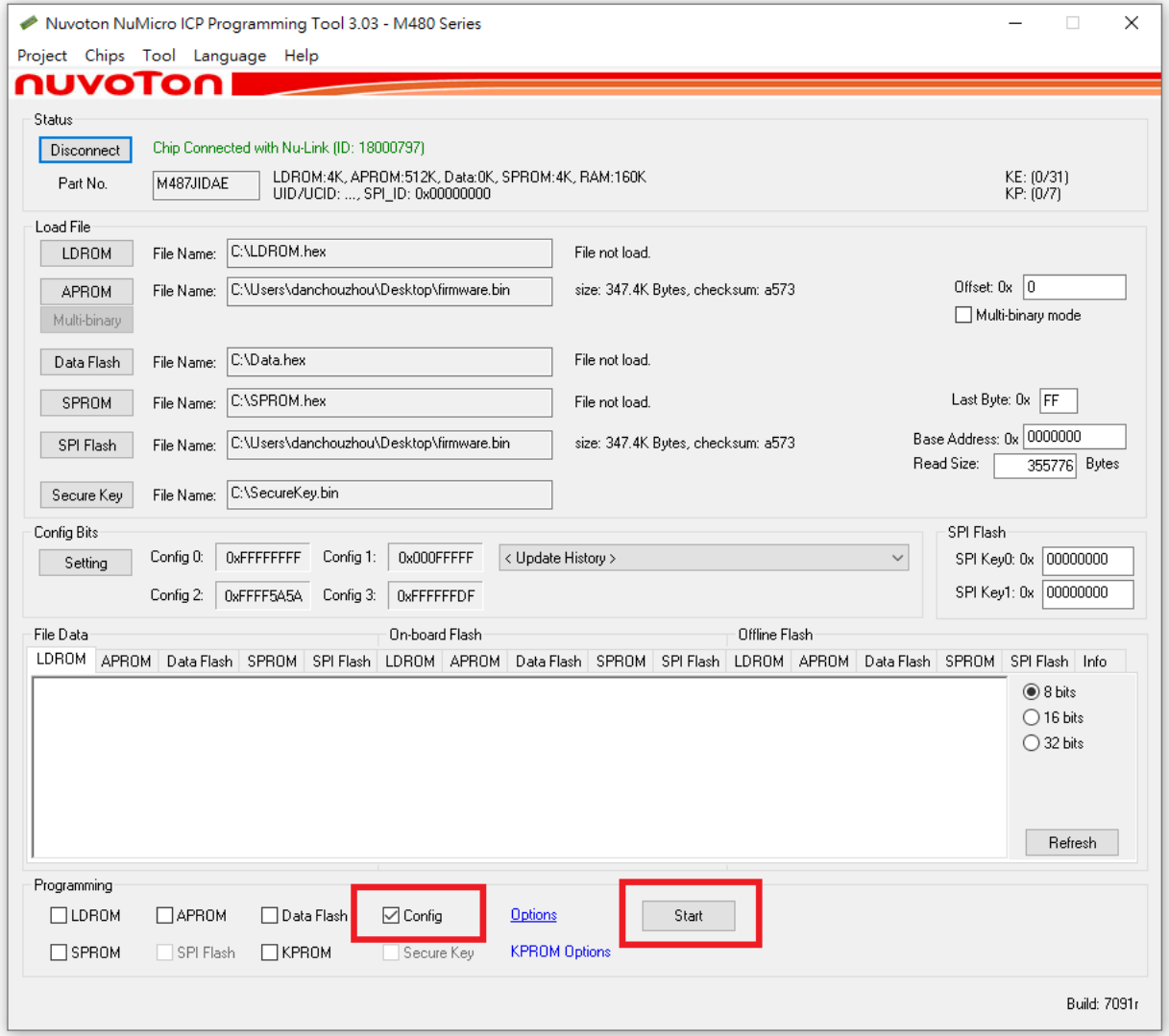
Factory reset



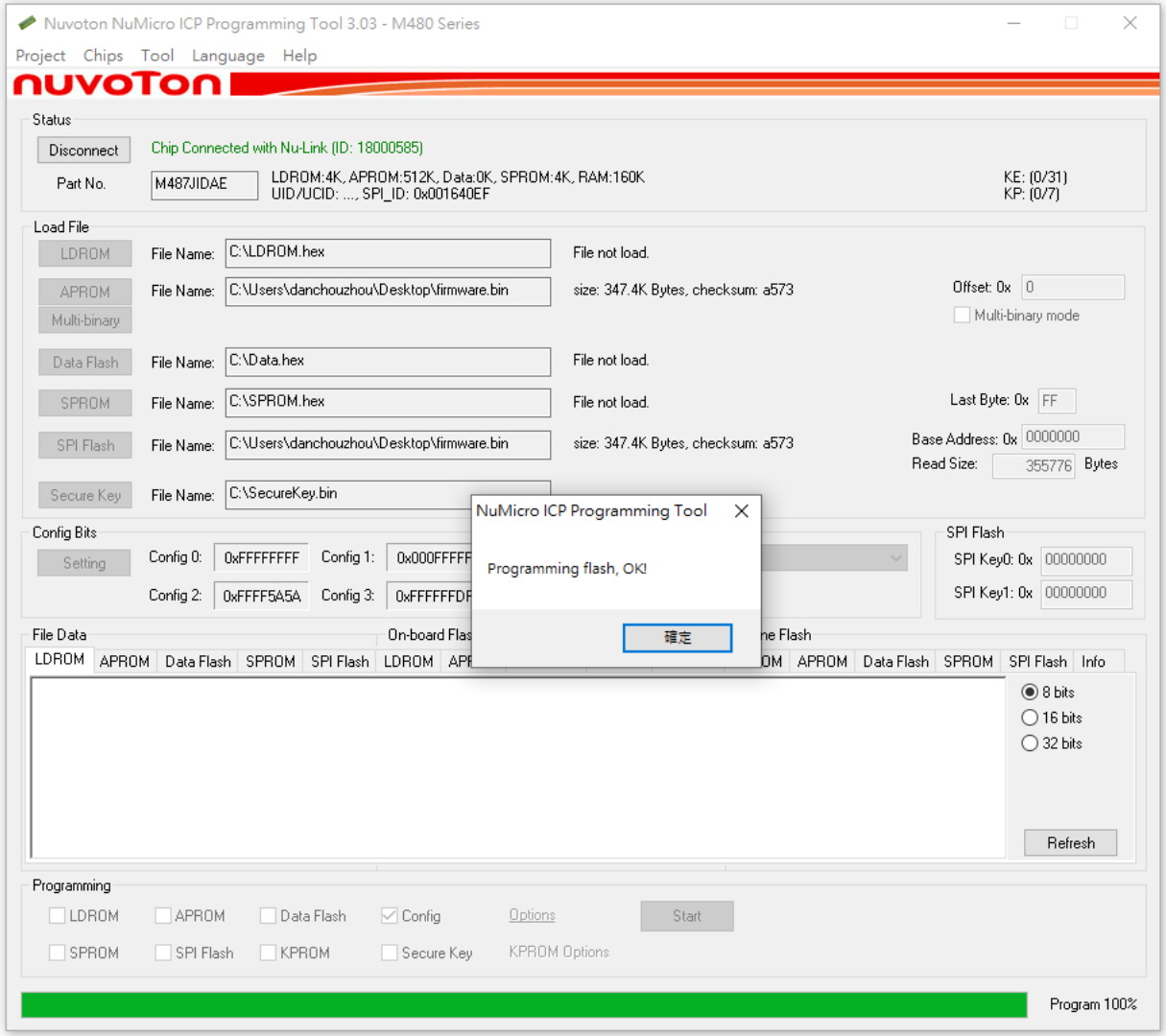
Factory reset



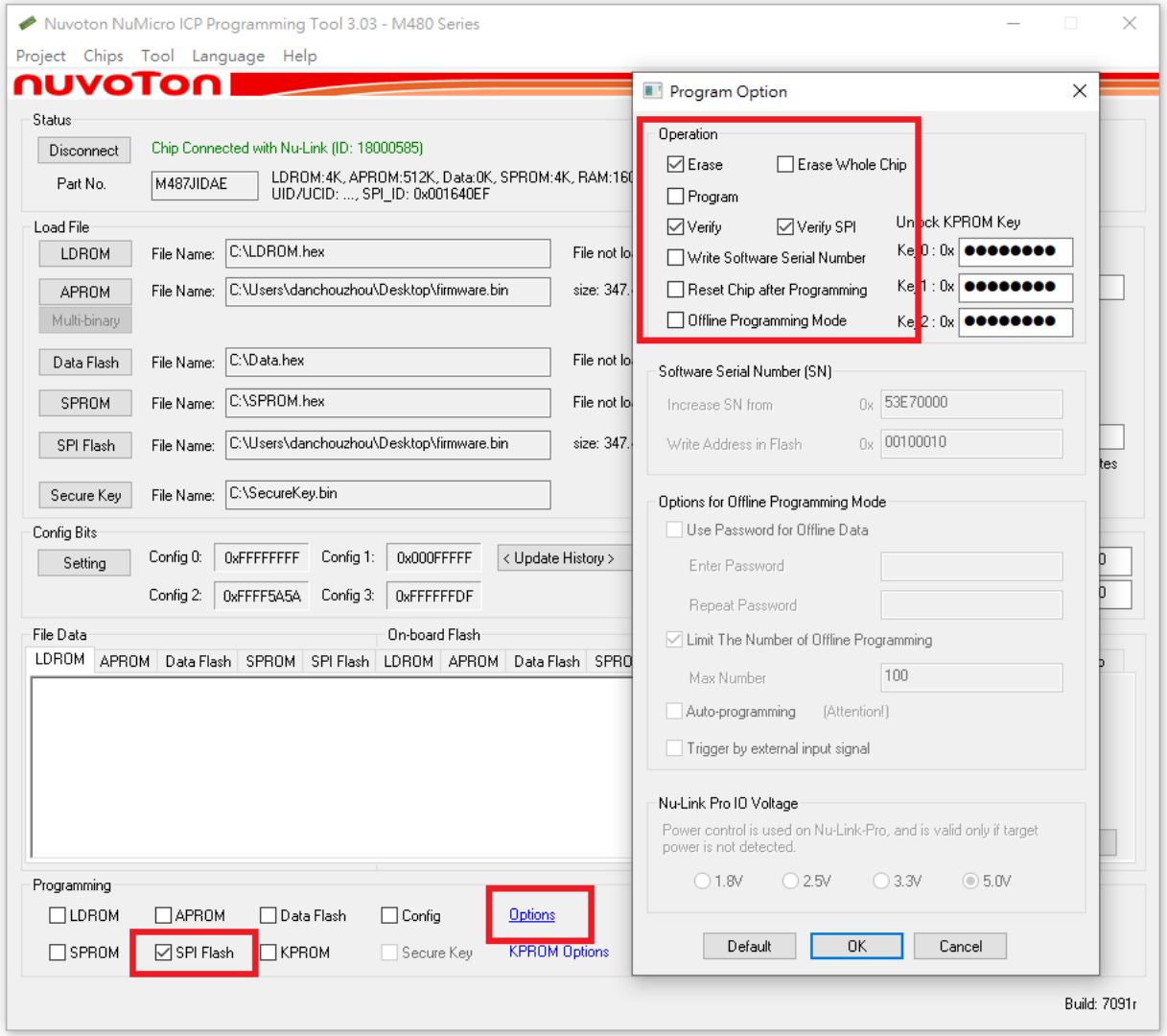
Factory reset



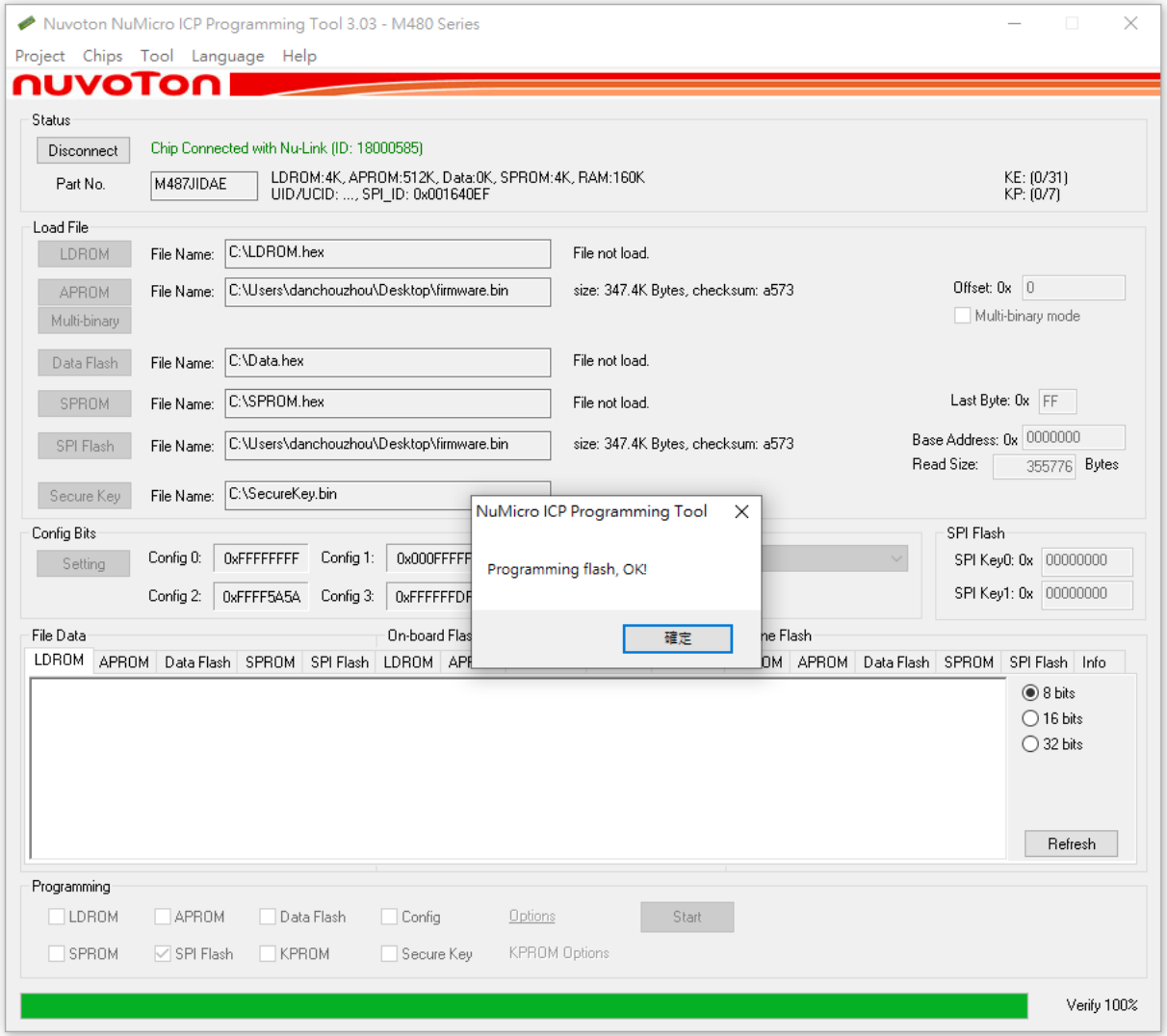
Factory reset



Factory reset



Factory reset



Factory reset