CodeGenerator v1.0 For Development Testing

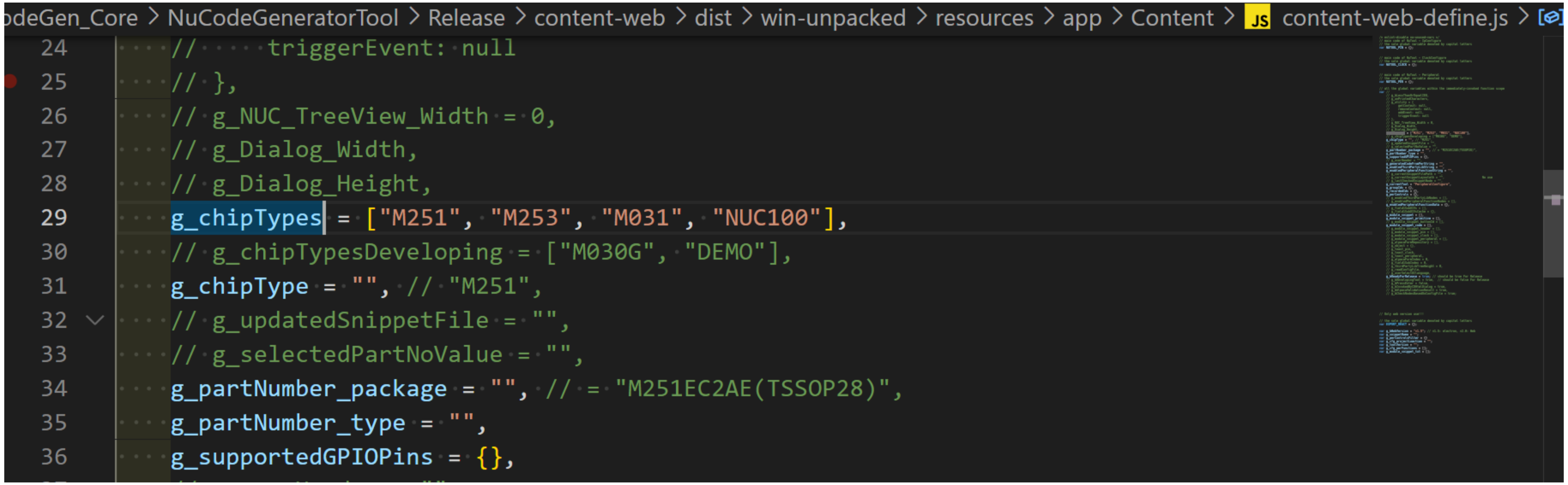
CM Wu

2023/11/10

| Photo by Jeffery Cheng

How to add a new Chip Series?

- Chip Name should use uppercase, EX: M2003C
  - Step 1: 在content-web-define.js的g_chipTypes新增 “M2003C”
  - Step 2: 在PeripheralConfigure資料夾底下新增 “NUC_M2003C_Content.js”

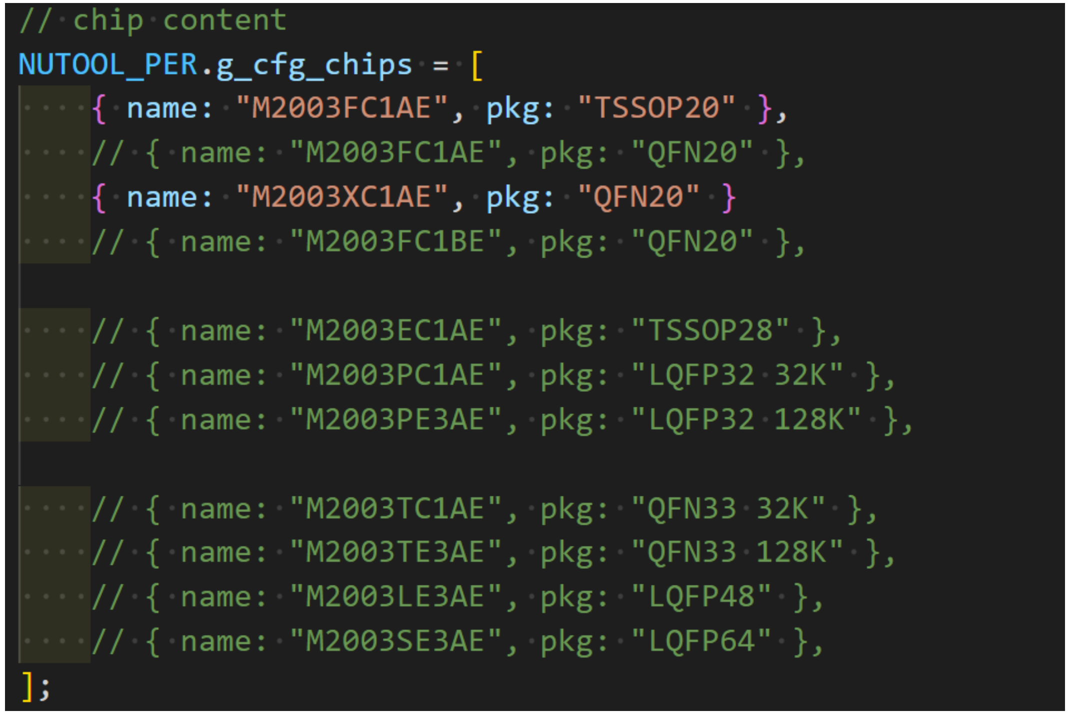


odeGen_Core > NuCodeGeneratorTool > Release > content-web > dist > win-unpacked > resources > app > Content > content-web-define.js >

```
    //     triggerEvent: null
    // },
    // g_NUC_TreeView_Width = 0,
    // g_Dialog_Width,
    // g_Dialog_Height,
    g_chipTypes = ["M251", "M253", "M031", "NUC100"],
    // g_chipTypesDeveloping = ["M030G", "DEMO"],
    g_chipType = "", // "M251",
    // g_updatedSnippetFile = "",
    // g_selectedPartNoValue = "",
    g_partNumber_package = "", // = "M251EC2AE(TSSOP28)",
    g_partNumber_type = "",
    g_supportedGPIOPins = {},
```

How to add a new Part No.?

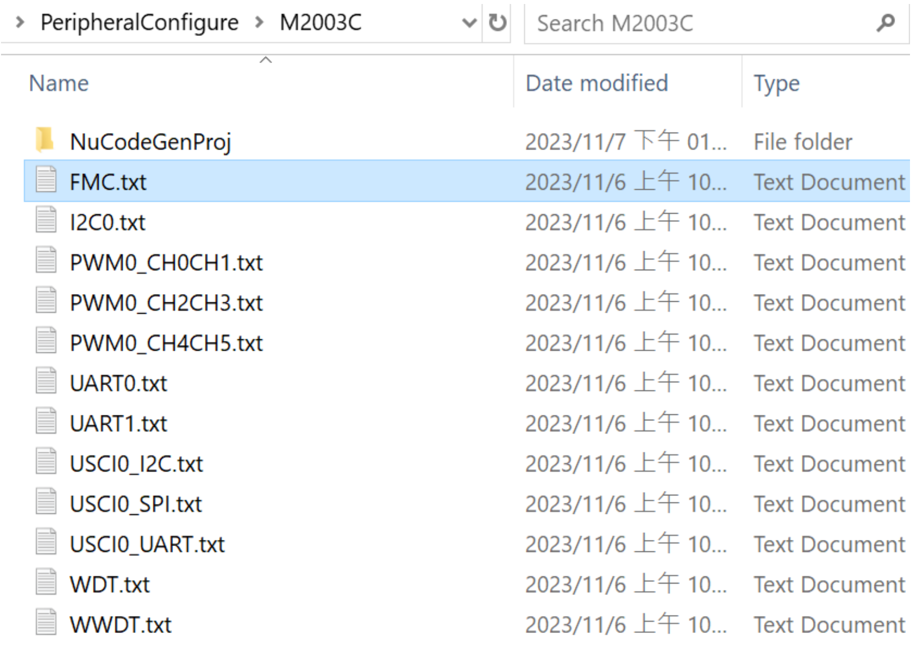
- 在對應的NUC_XXX_Content.js 新增, EX: M2003XC1AE
    - Step 1: 在NUC_M2003C_Content.js的 NUTOOL_PER.g_cfg_chips 加入M2003XC1AE



```
// chip content
NUTOOL_PER.g_cfg_chips = [
    { name: "M2003FC1AE", pkg: "TSSOP20" },
    // { name: "M2003FC1AE", pkg: "QFN20" },
    { name: "M2003XC1AE", pkg: "QFN20" }
    // { name: "M2003FC1BE", pkg: "QFN20" },

    // { name: "M2003EC1AE", pkg: "TSSOP28" },
    // { name: "M2003PC1AE", pkg: "LQFP32 32K" },
    // { name: "M2003PE3AE", pkg: "LQFP32 128K" },

    // { name: "M2003TC1AE", pkg: "QFN33 32K" },
    // { name: "M2003TE3AE", pkg: "QFN33 128K" },
    // { name: "M2003LE3AE", pkg: "LQFP48" },
    // { name: "M2003SE3AE", pkg: "LQFP64" },
];
```

How to add a new Peripheral Function? (1/2)

- 新增對應的txt檔. EX: M2003C FMC
  - Step 1: 新增PeripheralConfigure/M2003C 資料夾.
  - Step 2: 在M2003C 裡面新增 FMC.txt
  - Step 3: 在NUC_M2003C_Content.js的NUTOOL_PER.g_cfg_perFunctions新增 FMC.txt



› PeripheralConfigure › M2003C

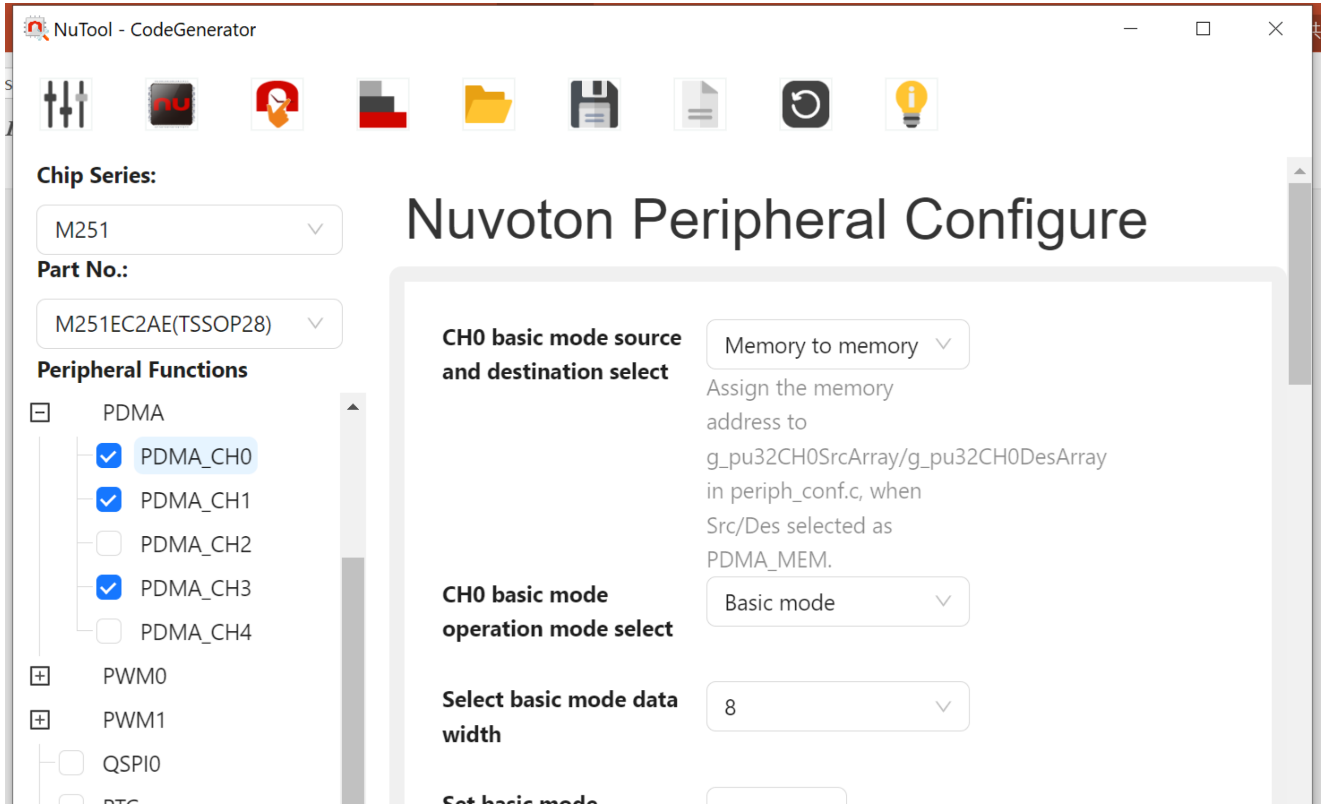
Search M2003C

Name	Date modified	Type
NuCodeGenProj	2023/11/7 下午 01...	File folder
FMC.txt	2023/11/6 上午 10...	Text Document
I2C0.txt	2023/11/6 上午 10...	Text Document
PWM0_CH0CH1.txt	2023/11/6 上午 10...	Text Document
PWM0_CH2CH3.txt	2023/11/6 上午 10...	Text Document
PWM0_CH4CH5.txt	2023/11/6 上午 10...	Text Document
UART0.txt	2023/11/6 上午 10...	Text Document
UART1.txt	2023/11/6 上午 10...	Text Document
USCI0_I2C.txt	2023/11/6 上午 10...	Text Document
USCI0_SPI.txt	2023/11/6 上午 10...	Text Document
USCI0_UART.txt	2023/11/6 上午 10...	Text Document
WDT.txt	2023/11/6 上午 10...	Text Document
WWDT.txt	2023/11/6 上午 10...	Text Document

How to add a new Peripheral Function? (2/2)

- 若txt的檔名有底線(_)，會將底線前的名字視作第一層，然後底線後的會顯示在第二層。目前僅支援兩層樹狀圖結構。

```
NUTOOL_PER.g_cfg_perFunctions = [
    ...
    'PDMA_CH0.txt',
    'PDMA_CH1.txt',
    'PDMA_CH2.txt',
    'PDMA_CH3.txt',
    'PDMA_CH4.txt',
    'PDMA_CH5.txt',
    'PDMA_CH6.txt',
    'PDMA_CH7.txt',
    ...
];
```



NuTool - CodeGenerator

Chip Series: M251

Part No.: M251EC2AE(TSSOP28)

Peripheral Functions

⊟ PDMA
- ☑ PDMA_CH0
- ☑ PDMA_CH1
- ☐ PDMA_CH2
- ☑ PDMA_CH3
- ☐ PDMA_CH4

⊞ PWM0

⊞ PWM1

☐ QSPI0

☐ RTC

Nuvoton Peripheral Configure

CH0 basic mode source and destination select: Memory to memory

Assign the memory address to g_pu32CH0SrcArray/g_pu32CH0DesArray in periph_conf.c, when Src/Des selected as PDMA_MEM.

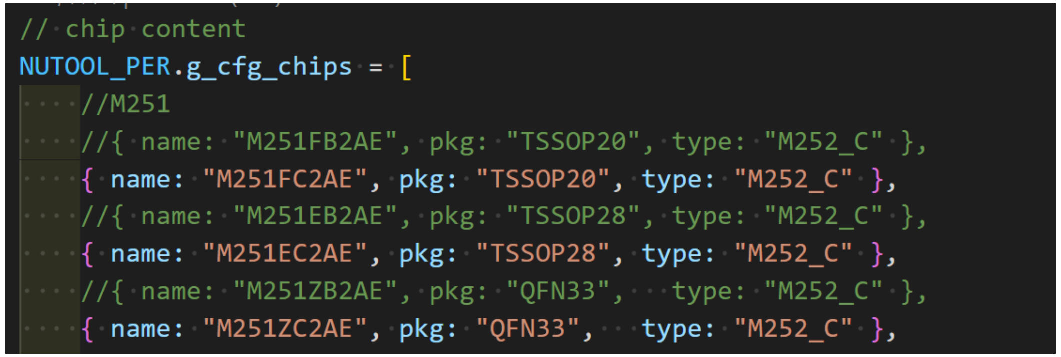
CH0 basic mode operation mode select: Basic mode

Select basic mode data width: 8

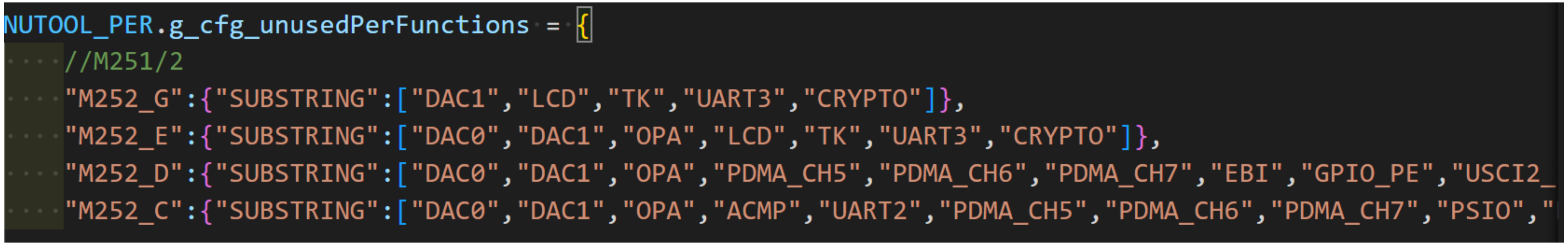
Set basic mode

Advanced Control of Peripheral Function(1/3)

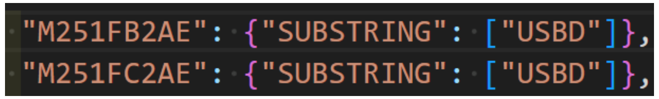
- 定義在對應的NUC_XXX_Content.js
  - NUTOOL_PER.g_cfg_unusedPerFunctions: {“type or name”: [Unused peripheral]}



```
// chip content
NUTOOL_PER.g_cfg_chips = [
    //M251
    //{ name: "M251FB2AE", pkg: "TSSOP20", type: "M252_C" },
    { name: "M251FC2AE", pkg: "TSSOP20", type: "M252_C" },
    //{ name: "M251EB2AE", pkg: "TSSOP28", type: "M252_C" },
    { name: "M251EC2AE", pkg: "TSSOP28", type: "M252_C" },
    //{ name: "M251ZB2AE", pkg: "QFN33",   type: "M252_C" },
    { name: "M251ZC2AE", pkg: "QFN33",   type: "M252_C" },
```



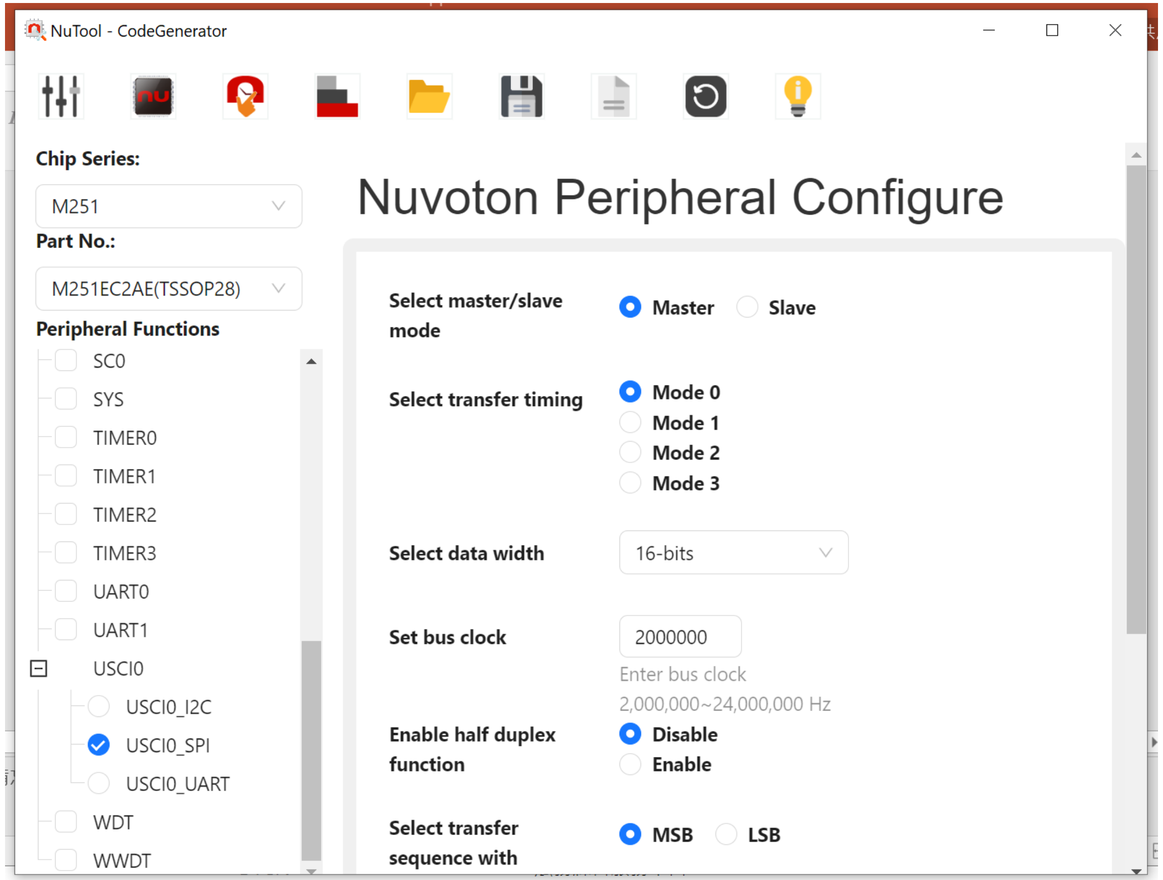
```
NUTOOL_PER.g_cfg_unusedPerFunctions = {
    //M251/2
    "M252_G":{"SUBSTRING":["DAC1","LCD","TK","UART3","CRYPTO"]},
    "M252_E":{"SUBSTRING":["DAC0","DAC1","OPA","LCD","TK","UART3","CRYPTO"]},
    "M252_D":{"SUBSTRING":["DAC0","DAC1","OPA","PDMA_CH5","PDMA_CH6","PDMA_CH7","EBI","GPIO_PE","USCI2_
    "M252_C":{"SUBSTRING":["DAC0","DAC1","OPA","ACMP","UART2","PDMA_CH5","PDMA_CH6","PDMA_CH7","PSIO","
```



```
"M251FB2AE": {"SUBSTRING": ["USBD"]},
"M251FC2AE": {"SUBSTRING": ["USBD"]},
```

Advanced Control of Peripheral Function(2/3)

- NUTOOL_PER.g_cfg_perFuncMutex: 互斥, 將多選的方框變成單選的原框.
  EX: `NUTOOL_PER.g_cfg_perFuncMutex = ["USCI0", "USCI1", "USCI2"];`



Chip Series: M251

Part No.: M251EC2AE(TSSOP28)

Peripheral Functions

- □ SC0
- □ SYS
- □ TIMER0
- □ TIMER1
- □ TIMER2
- □ TIMER3
- □ UART0
- □ UART1
- ⊟ USCI0
  - ○ USCI0_I2C
  - ● USCI0_SPI
  - ○ USCI0_UART
- □ WDT
- □ WWDT

Nuvoton Peripheral Configure

Select master/slave mode: ● Master ○ Slave

Select transfer timing: ● Mode 0 ○ Mode 1 ○ Mode 2 ○ Mode 3

Select data width: 16-bits

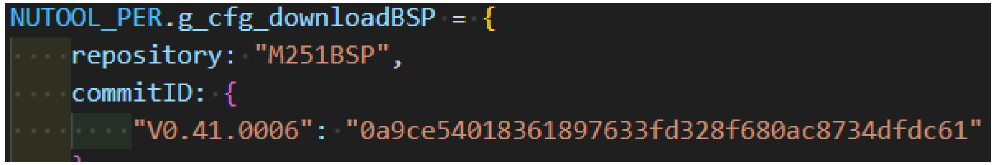
Set bus clock: 2000000
Enter bus clock
2,000,000~24,000,000 Hz

Enable half duplex function: ● Disable ○ Enable

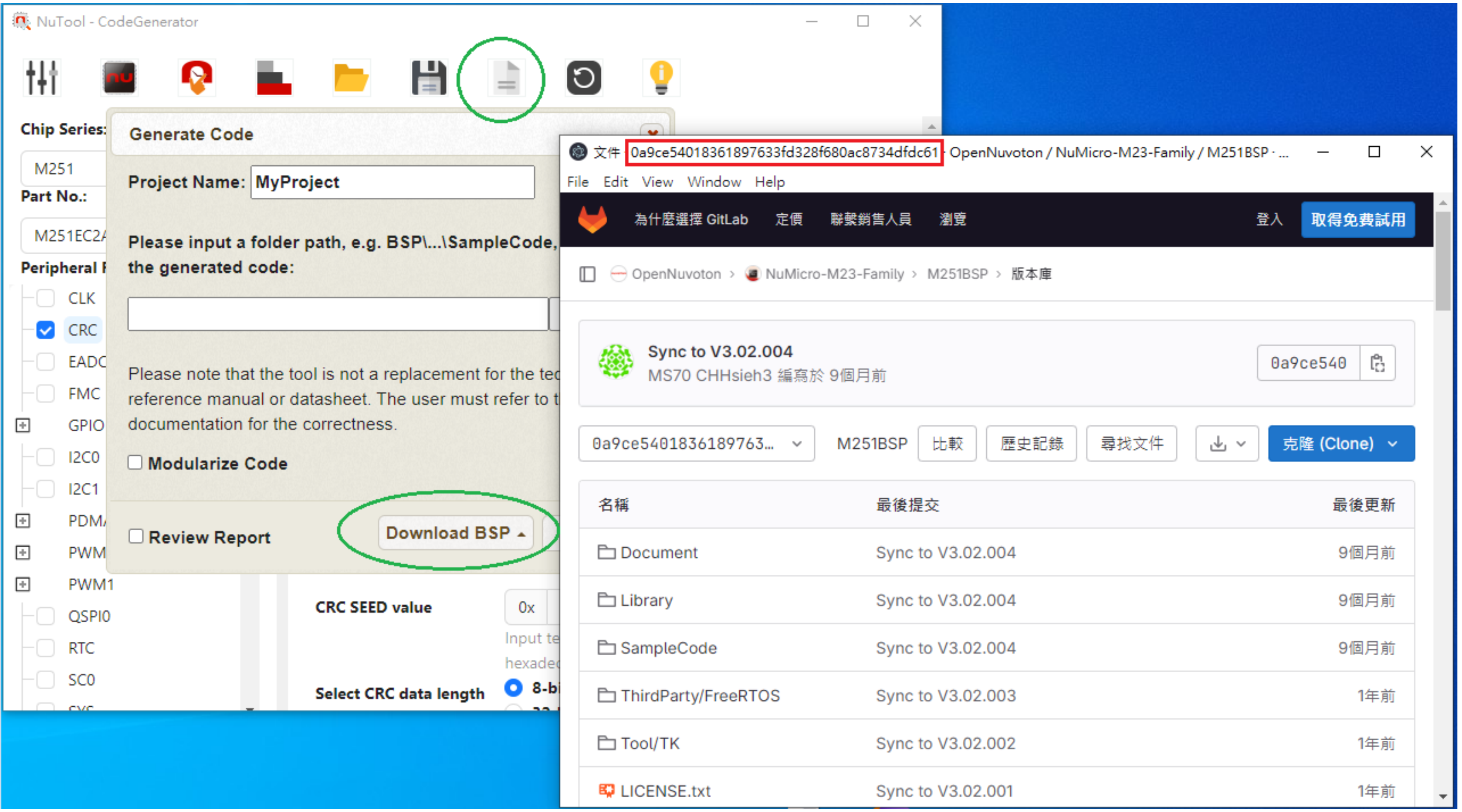
Select transfer sequence with: ● MSB ○ LSB

Advanced Control of Peripheral Function(3/3)

- NUTOOL_PER.g_cfg_downloadBSP: Download BSP information as below



```
NUTOOL_PER.g_cfg_downloadBSP = {
    repository: "M251BSP",
    commitID: {
        "V0.41.0006": "0a9ce54018361897633fd328f680ac8734dfdc61"
```



NuTool - CodeGenerator
Chip Series:
M251
Part No.:
M251EC2A
Peripheral F
CLK
☑ CRC
EADC
FMC
GPIO
I2C0
I2C1
PDMA
PWM
PWM1
QSPI0
RTC
SC0
SYS

Generate Code

Project Name: MyProject

Please input a folder path, e.g. BSP\...\SampleCode, the generated code:

Please note that the tool is not a replacement for the tec reference manual or datasheet. The user must refer to t documentation for the correctness.

☐ Modularize Code

☐ Review Report

Download BSP

CRC SEED value 0x

Input te hexadec

Select CRC data length ● 8-bi

文件 0a9ce54018361897633fd328f680ac8734dfdc61 · OpenNuvoton / NuMicro-M23-Family / M251BSP · ...
File Edit View Window Help
為什麼選擇 GitLab 定價 聯繫銷售人員 瀏覽 登入 取得免費試用
OpenNuvoton › NuMicro-M23-Family › M251BSP › 版本庫

Sync to V3.02.004
MS70 CHHsieh3 編寫於 9個月前 0a9ce540

0a9ce5401836189763... M251BSP 比較 歷史記錄 尋找文件 克隆 (Clone)

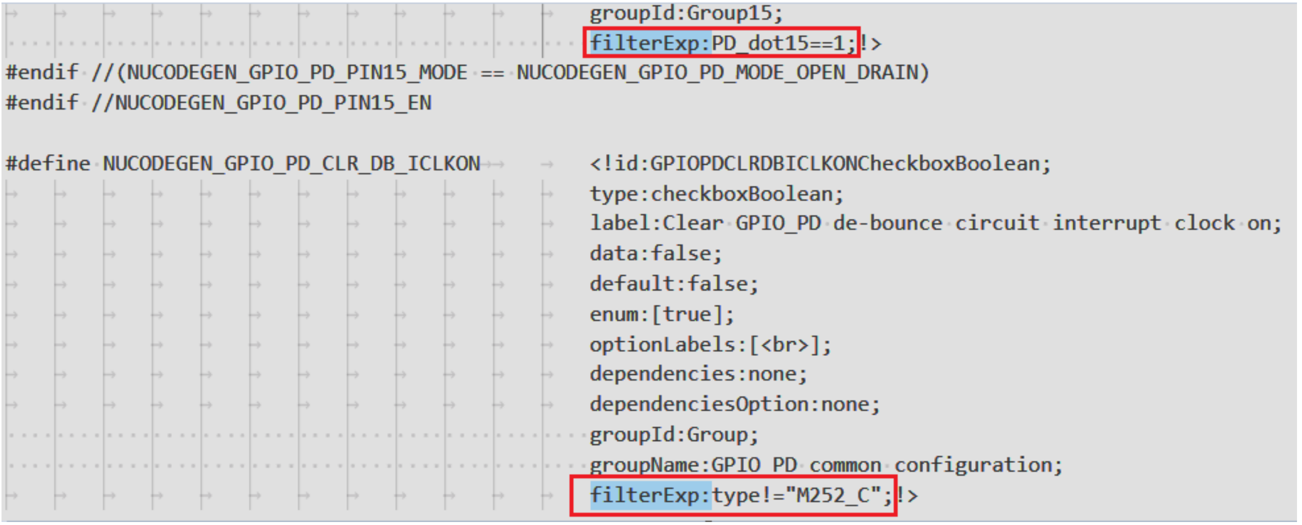
名稱	最後提交	最後更新
Document	Sync to V3.02.004	9個月前
Library	Sync to V3.02.004	9個月前
SampleCode	Sync to V3.02.004	9個月前
ThirdParty/FreeRTOS	Sync to V3.02.003	1年前
Tool/TK	Sync to V3.02.002	1年前
LICENSE.txt	Sync to V3.02.001	1年前

Txt Tag Syntax(1/7)

1. id: must be unique.
2. type: determines the type of UI.
3. label: determines the description text shown around the UI.
4. data: stores the current value of UI.
5. default: stores the default value of UI.
6. helper: determines the helper text following UI.
7. enum :determines the option data array.
8. optionLabels: determines the option label array which should corresponds to enum.

Txt Tag Syntax(2/7)

9. maximum: determines the maximum input value.
10. minimum: determines the minimum input value.
11. group: determines the title text following UI.
12. filterExp: built-in conditions.
    - EX: GPIO pins have built-in condition called Px_dotx.
    - EX: In NUC_MXXX_Content.js, type is defined in NUTOOL_PER.g_cfg_chips.

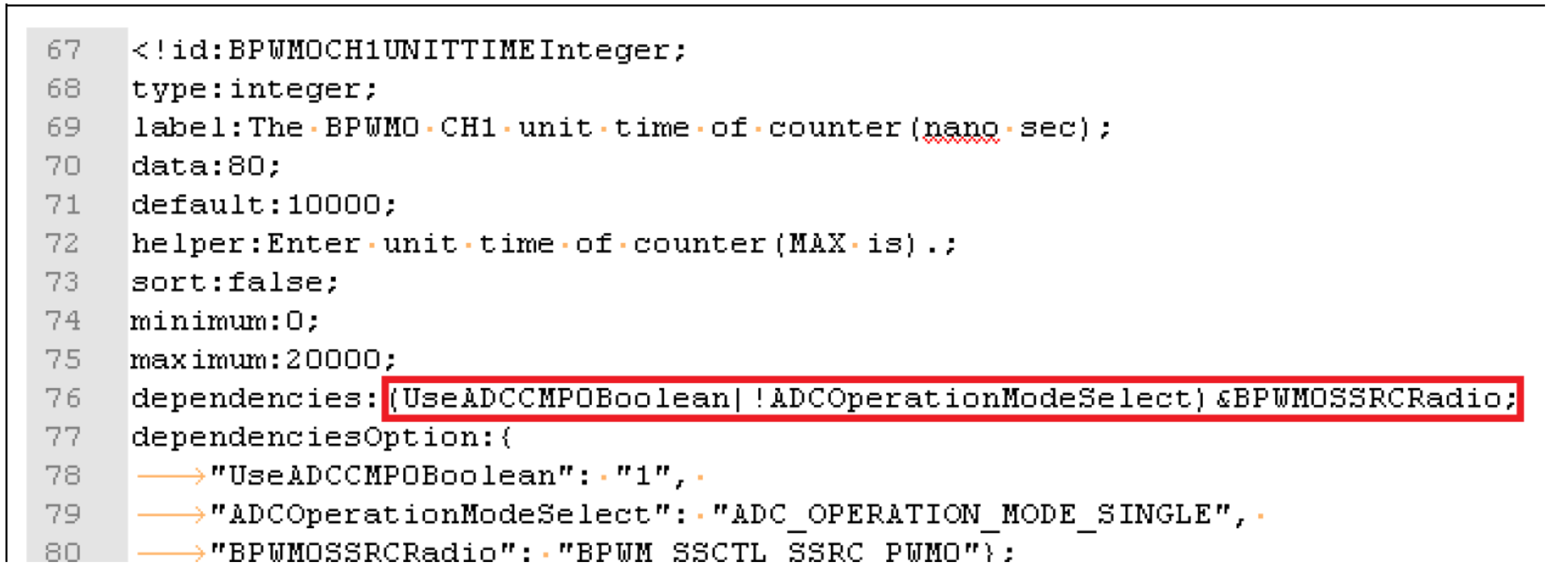


```
                                                groupId:Group15;
                                                filterExp:PD_dot15==1;!>
#endif //(NUCODEGEN_GPIO_PD_PIN15_MODE == NUCODEGEN_GPIO_PD_MODE_OPEN_DRAIN)
#endif //NUCODEGEN_GPIO_PD_PIN15_EN

#define NUCODEGEN_GPIO_PD_CLR_DB_ICLKON                 <!id:GPIOPDCLRDBICLKONCheckboxBoolean;
                                                type:checkboxBoolean;
                                                label:Clear GPIO_PD de-bounce circuit interrupt clock on;
                                                data:false;
                                                default:false;
                                                enum:[true];
                                                optionLabels:[<br>];
                                                dependencies:none;
                                                dependenciesOption:none;
                                                groupId:Group;
                                                groupName:GPIO_PD common configuration;
                                                filterExp:type!="M252_C";!>
```

Txt Tag Syntax(3/7)

(UI renders only when data of another UI meets some kind of conditions.)

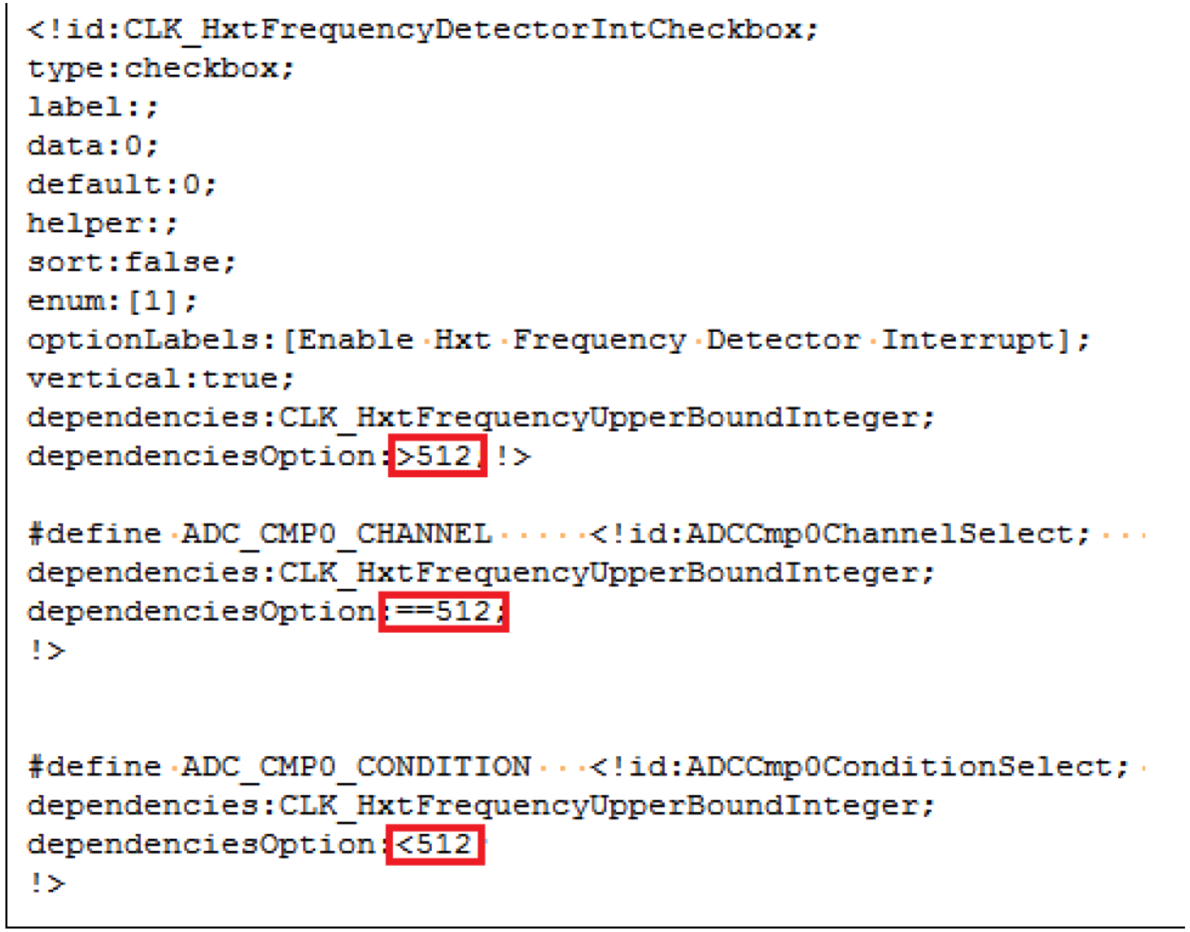
13. dependencies: determines the id of dependent UI.
14. dependenciesOption: determines the expression.
    EX: dependencies supports bitwise operation has Boolean or (|), Boolean and (&) and Boolean not (!).



```
<!id:BPWM0CH1UNITTIMEInteger;
type:integer;
label:The BPWM0 CH1 unit time of counter(nano sec);
data:80;
default:10000;
helper:Enter unit time of counter(MAX is).;
sort:false;
minimum:0;
maximum:20000;
dependencies:(UseADCCMP0Boolean|!ADCOperationModeSelect)&BPWM0SSRCRadio;
dependenciesOption:{
    "UseADCCMP0Boolean": "1", 
    "ADCOperationModeSelect": "ADC_OPERATION_MODE_SINGLE", 
    "BPWM0SSRCRadio": "BPWM_SSCTL_SSRC_PWM0"};
```

Txt Tag Syntax(4/7)

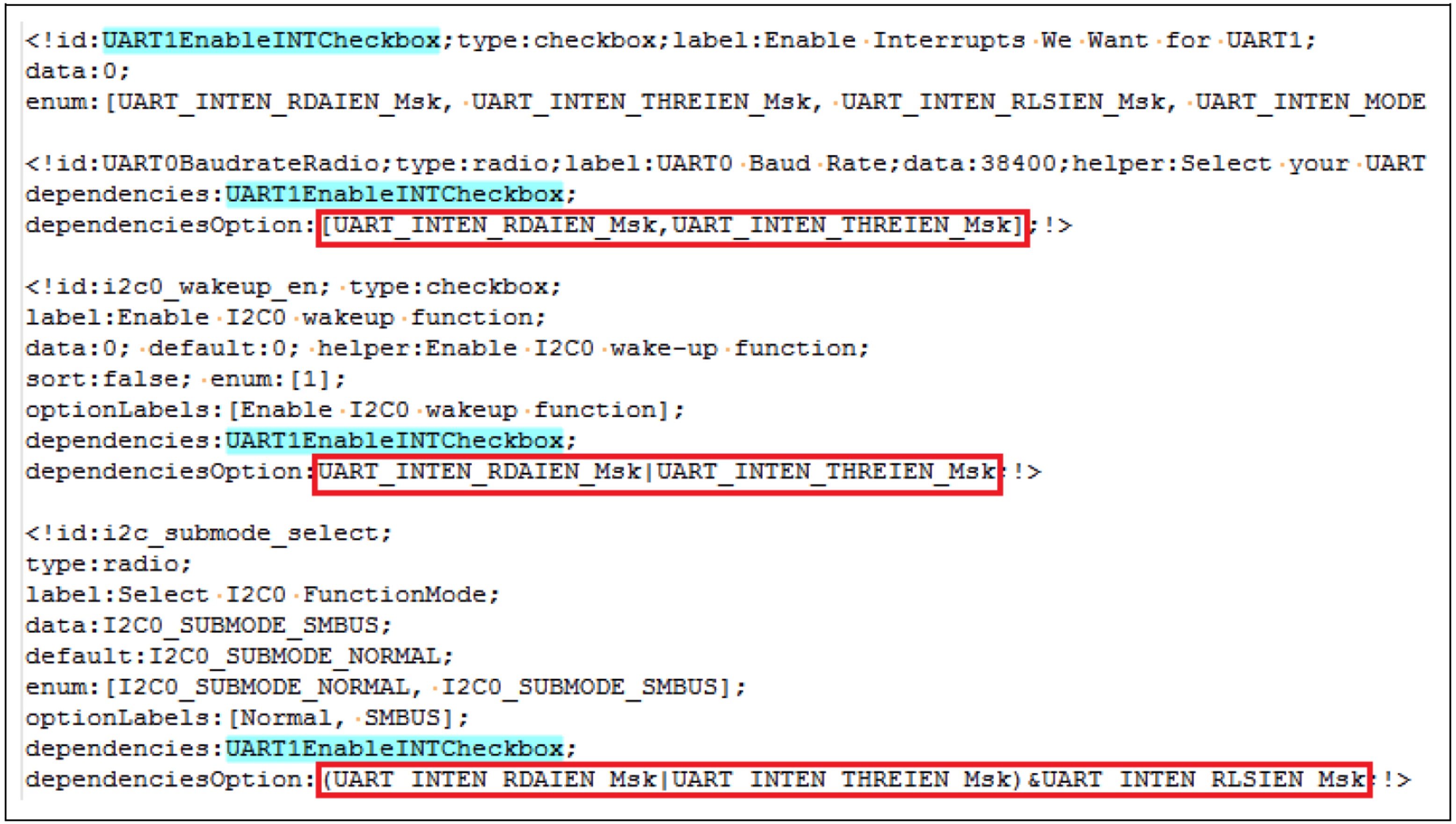
Example: dependenciesOption supports Greater than, Equal to, and Less than



```
<!id:CLK_HxtFrequencyDetectorIntCheckbox;
type:checkbox;
label:;
data:0;
default:0;
helper:;
sort:false;
enum:[1];
optionLabels:[Enable Hxt Frequency Detector Interrupt];
vertical:true;
dependencies:CLK_HxtFrequencyUpperBoundInteger;
dependenciesOption:>512; !>

#define ADC_CMP0_CHANNEL     <!id:ADCCmp0ChannelSelect;   
dependencies:CLK_HxtFrequencyUpperBoundInteger;
dependenciesOption:==512;
!>


#define ADC_CMP0_CONDITION   <!id:ADCCmp0ConditionSelect; 
dependencies:CLK_HxtFrequencyUpperBoundInteger;
dependenciesOption:<512;
!>
```

Txt Tag Syntax(5/7)

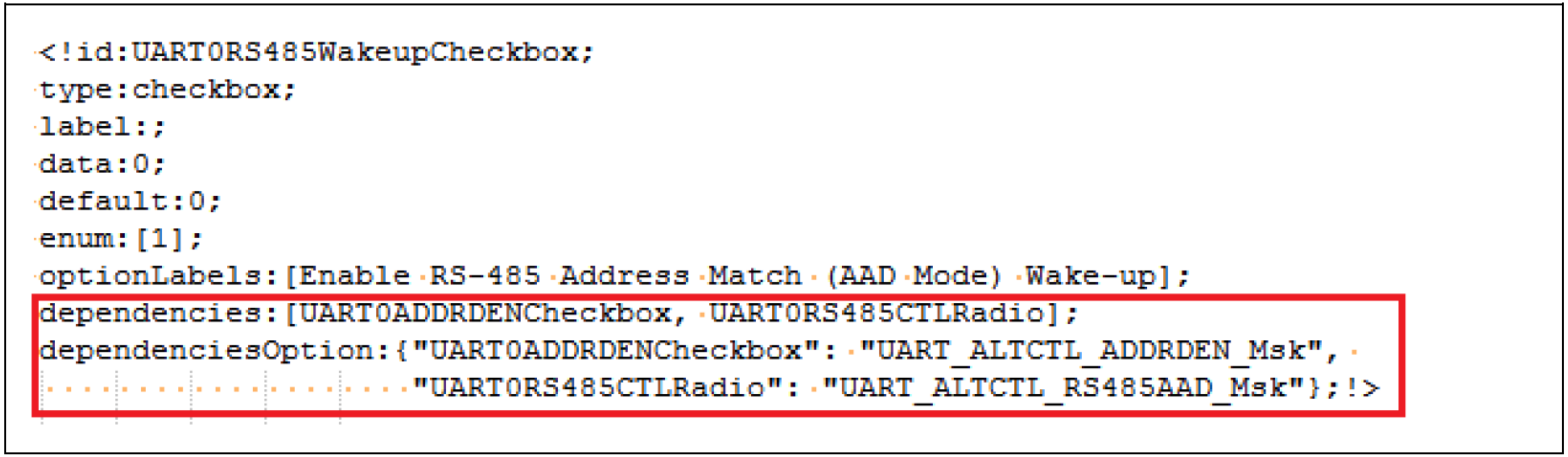
Example: dependenciesOption supports bitwise operation (|) (&) (!) and.



```
<!id:UART1EnableINTCheckbox;type:checkbox;label:Enable Interrupts We Want for UART1;
data:0;
enum:[UART_INTEN_RDAIEN_Msk, UART_INTEN_THREIEN_Msk, UART_INTEN_RLSIEN_Msk, UART_INTEN_MODE

<!id:UART0BaudrateRadio;type:radio;label:UART0 Baud Rate;data:38400;helper:Select your UART
dependencies:UART1EnableINTCheckbox;
dependenciesOption:[UART_INTEN_RDAIEN_Msk,UART_INTEN_THREIEN_Msk];!>

<!id:i2c0_wakeup_en; type:checkbox;
label:Enable I2C0 wakeup function;
data:0; default:0; helper:Enable I2C0 wake-up function;
sort:false; enum:[1];
optionLabels:[Enable I2C0 wakeup function];
dependencies:UART1EnableINTCheckbox;
dependenciesOption:UART_INTEN_RDAIEN_Msk|UART_INTEN_THREIEN_Msk;!>

<!id:i2c_submode_select;
type:radio;
label:Select I2C0 FunctionMode;
data:I2C0_SUBMODE_SMBUS;
default:I2C0_SUBMODE_NORMAL;
enum:[I2C0_SUBMODE_NORMAL, I2C0_SUBMODE_SMBUS];
optionLabels:[Normal, SMBUS];
dependencies:UART1EnableINTCheckbox;
dependenciesOption:(UART_INTEN_RDAIEN_Msk|UART_INTEN_THREIEN_Msk)&UART_INTEN_RLSIEN_Msk;!>
```

Txt Tag Syntax(6/7)

Example: Support multi conditions.



```
 <!id:UART0RS485WakeupCheckbox;
 type:checkbox;
 label:;
 data:0;
 default:0;
 enum:[1];
 optionLabels:[Enable RS-485 Address Match (AAD Mode) Wake-up];
dependencies:[UART0ADDRDENCheckbox, UART0RS485CTLRadio];
dependenciesOption:{"UART0ADDRDENCheckbox": "UART_ALTCTL_ADDRDEN_Msk", 
                    "UART0RS485CTLRadio": "UART_ALTCTL_RS485AAD_Msk"};!>
```

Txt Tag Syntax(7/7)

- Each field name (e.g. optionLabels) of tag is fixed and case sensitive
- Each field should end with semicolon
- If the value of field is array, it should be enclosed in brackets.
- If field is unused, the input should be none.
- The start and end symbols of tags are <! and !>, and the content in the middle is free to edit.

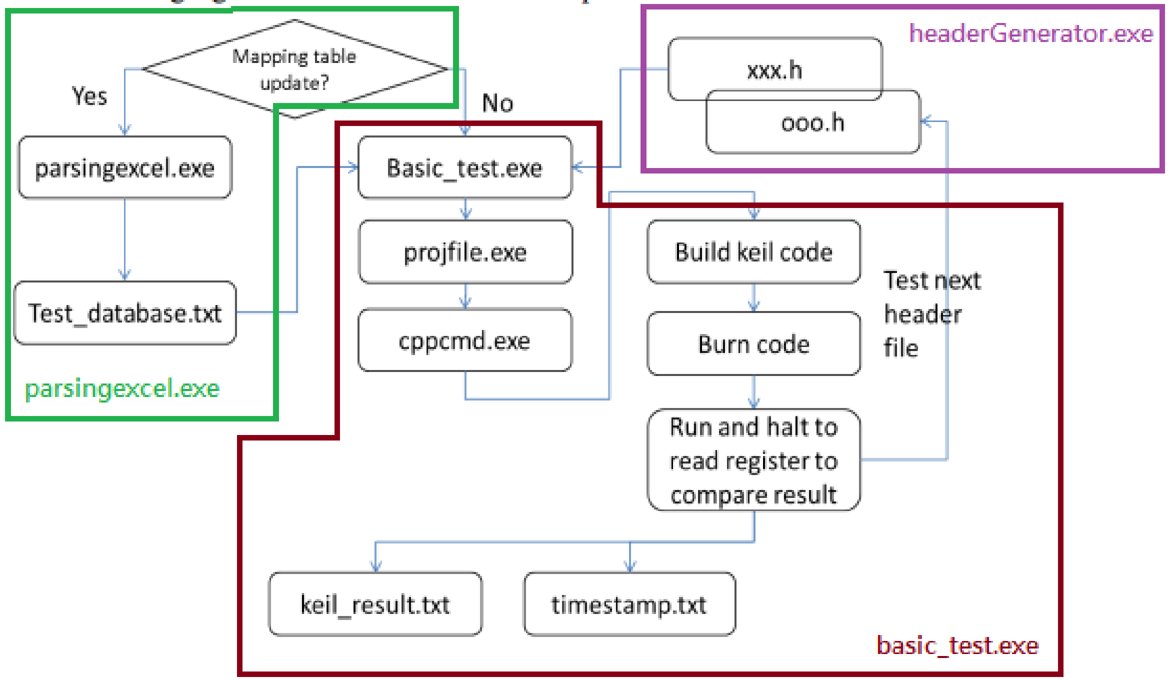
Functional Tests

- Purpose

  Run functional tests to ensure the correctness of .txt files and generated code.

- Flow

  The following figure shows the automatic test process and the final test result file.

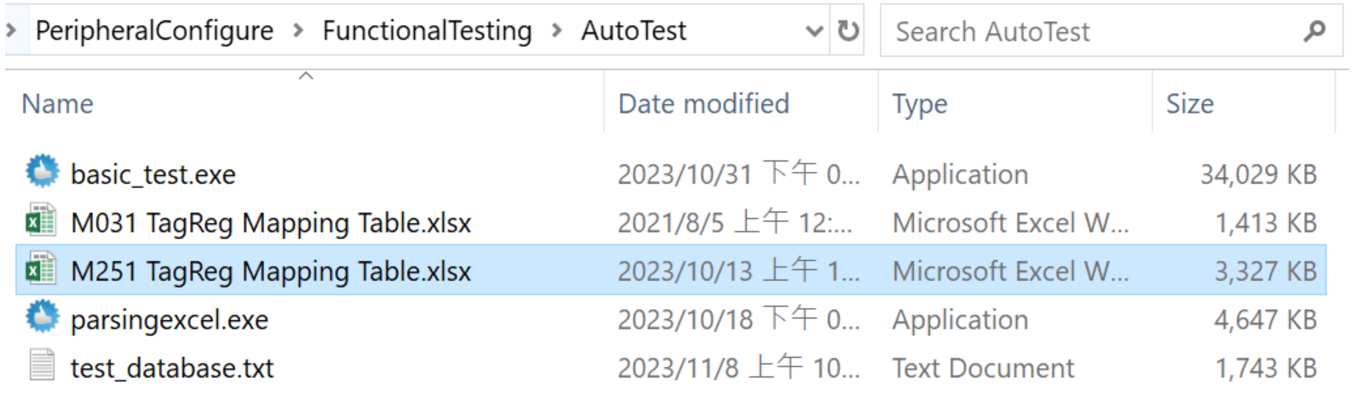


Functional Tests - parsingexcel.exe

- 解析 TagReg Mapping Table.xlsx, 產生 test_database.txt
- external.parsingexcel(chipType, peripherals)
  - chipType : <String> “M251”/”M031”/…
    => Determines chipType, default is GUI selected.
  - peripherals: <String> “all”/”CRC”/”CRC FMC”/…

    => Determines peripherals, default is “all”.

PS: This function require “XXX TagReg Mapping Table.xlsx”



Name	Date modified	Type	Size
basic_test.exe	2023/10/31 下午 0...	Application	34,029 KB
M031 TagReg Mapping Table.xlsx	2021/8/5 上午 12:...	Microsoft Excel W...	1,413 KB
M251 TagReg Mapping Table.xlsx	2023/10/13 上午 1...	Microsoft Excel W...	3,327 KB
parsingexcel.exe	2023/10/18 下午 0...	Application	4,647 KB
test_database.txt	2023/11/8 上午 10...	Text Document	1,743 KB

Functional Tests – headerGenerator.exe

- 根據選擇的條件, 產生對應的header files
- external.headerGenerator(partNo, peripherals, tagId)
  - partNo: <String> “M251EC2AE”/…
    => Determines part No., default is GUI selected.
  - Peripherals: <String> “CRC”/“FMC+WWDT”/”All”/…
    => Determines peripherals, default is GUI selected.
  - tagId: <String> "CLK_HXT_FailDetectorRadio"/” WWDT_PrescalerSelect“/…
    => Determines id in related txt file, default is null.

Functional Tests – basic_test.exe

- 測試result底下的header file, 是否與test_database.txt 一致
- external.basic_test(mode, chipType)
  - mode: <String> “-keil”/”-keil+download”/”-iar”/”-gcc”/”-all”
    => Run basic_test.exe with this this value, default=“-keil”.
  - chipType : <String> “M251”/”M031”/…
    => Determines chipType, default is GUI selected.

PS: This function require related application(Keil / Iar / Gcc).

Functional Tests – Execute All

- Execute parsingexcel.exe + headerGenerator.exe + basic_test.exe
- external.execAllFunctionalTests(chipType, partNo, peripherals, mode)
  - chipType : <String> “M251”/”M031”/...
    => Determines chipType, default is GUI selected.
  - partNo: <String> “M251EC2AE”/...
    => Determines part No., default is GUI selected.
  - Peripherals: <String> “CRC”/“FMC+WWDT”/”All”/...
    => Determines peripherals, default is GUI selected.
  - mode: <String> “-keil”/”-keil+download”/”-iar”/”-gcc”/”-all”
    => Run basic_test.exe with this this value, default=“-keil”.

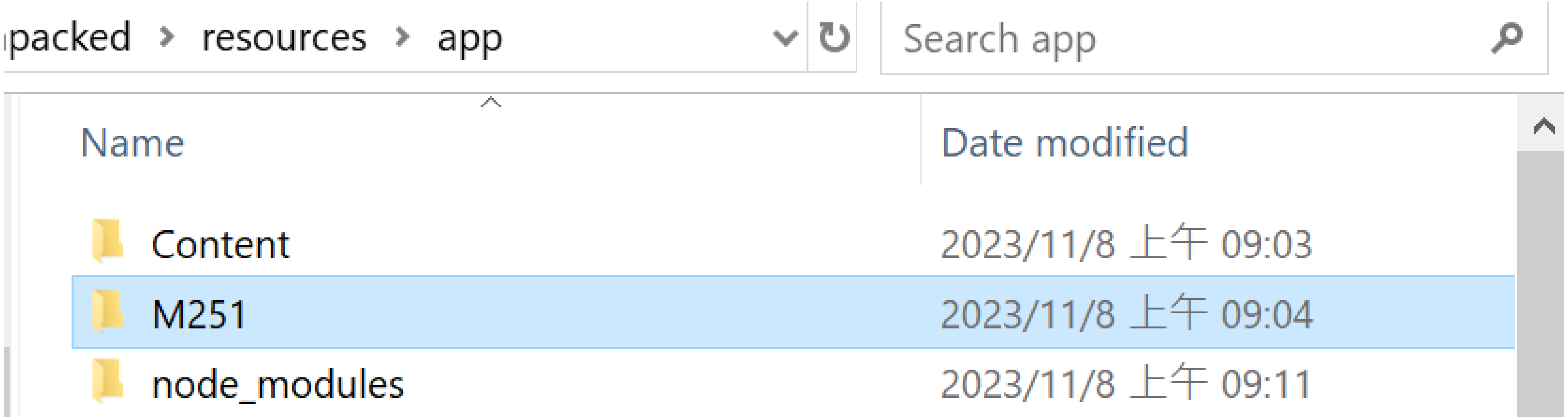
Functional Tests

- Usage

Step 1: Open CodeGenerator v1.0.

Step 2: Select items which want to test. EX: M251 -> M251EC2AE -> CRC+FMC

Step 3: Download related BSP, move to correct path and rename it.



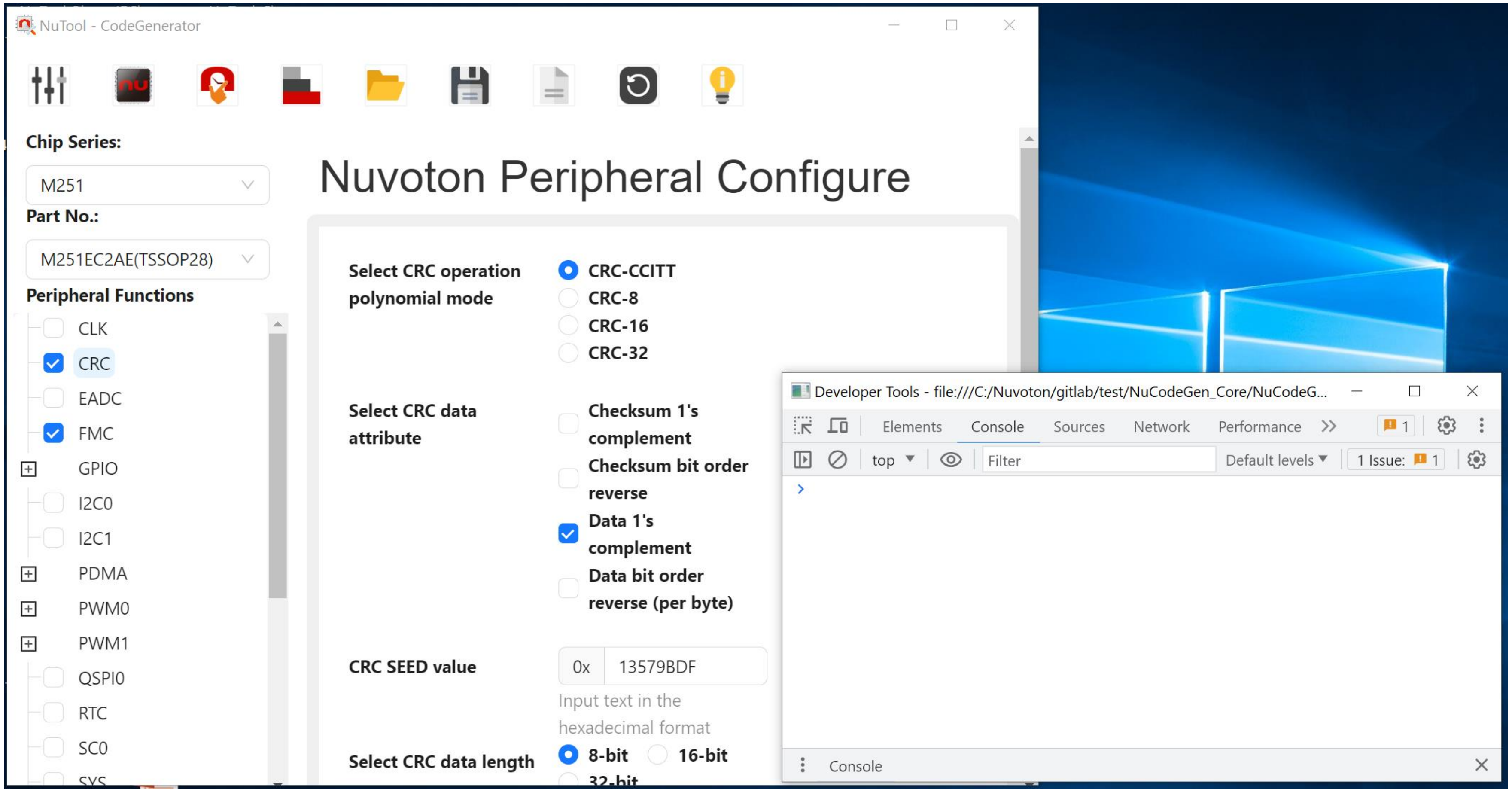
packed › resources › app

Search app

Name	Date modified
Content	2023/11/8 上午 09:03
M251	2023/11/8 上午 09:04
node_modules	2023/11/8 上午 09:11

Functional Tests

Step 4: Press keys Ctrl + Shift + I, launch Developer Tools, click Console tab.



NuTool - CodeGenerator

Chip Series: M251

Part No.: M251EC2AE(TSSOP28)

Peripheral Functions

- □ CLK
- ☑ CRC
- □ EADC
- ☑ FMC
- ⊞ GPIO
- □ I2C0
- □ I2C1
- ⊞ PDMA
- ⊞ PWM0
- ⊞ PWM1
- □ QSPI0
- □ RTC
- □ SC0
- □ SYS

Nuvoton Peripheral Configure

Select CRC operation polynomial mode: ● CRC-CCITT ○ CRC-8 ○ CRC-16 ○ CRC-32

Select CRC data attribute: □ Checksum 1's complement □ Checksum bit order reverse ☑ Data 1's complement □ Data bit order reverse (per byte)

CRC SEED value: 0x 13579BDF

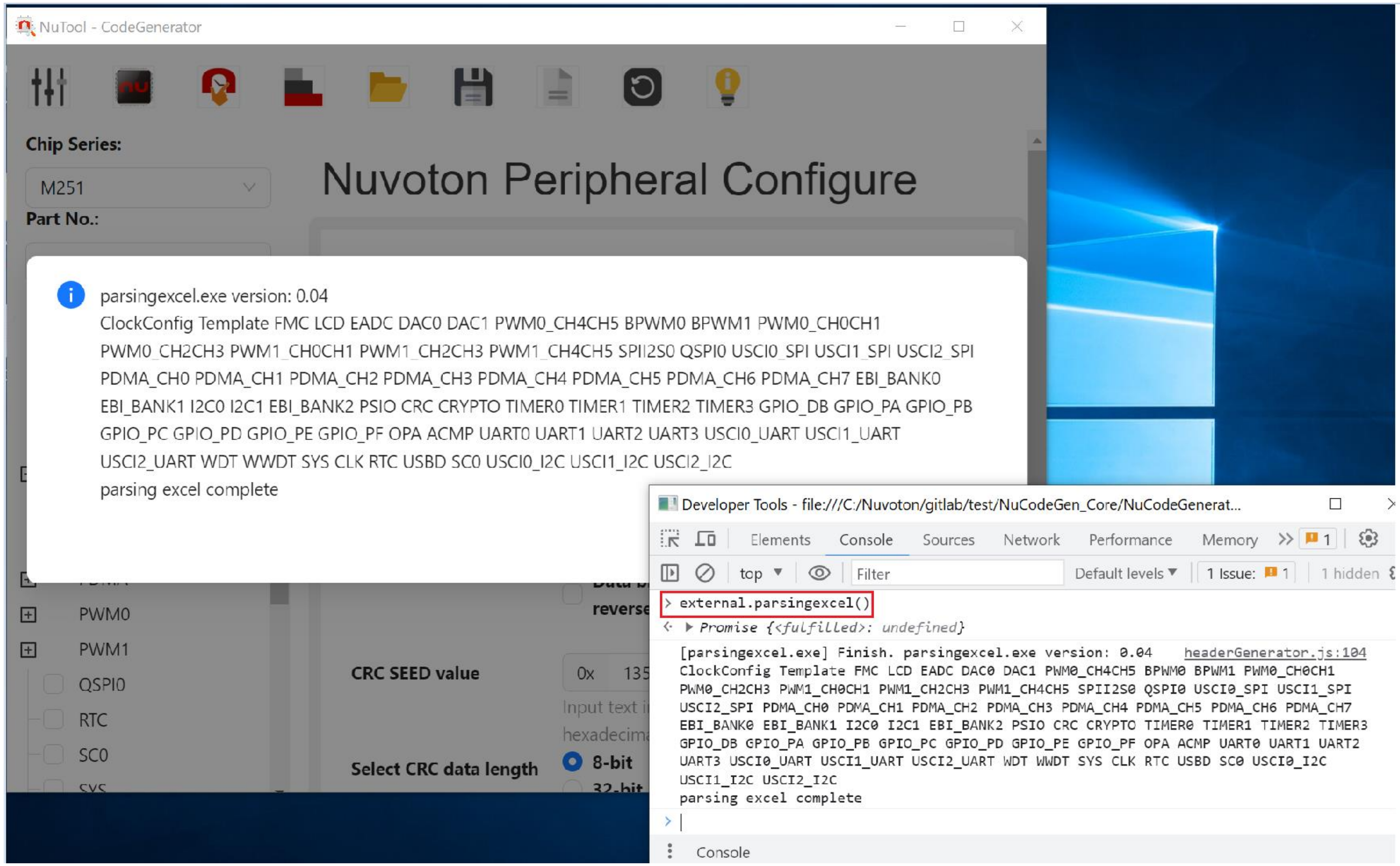
Input text in the hexadecimal format

Select CRC data length: ● 8-bit ○ 16-bit ○ 32-bit

Developer Tools - file:///C:/Nuvoton/gitlab/test/NuCodeGen_Core/NuCodeG...

Functional Tests

Step 5: Input external.parsingexcel().

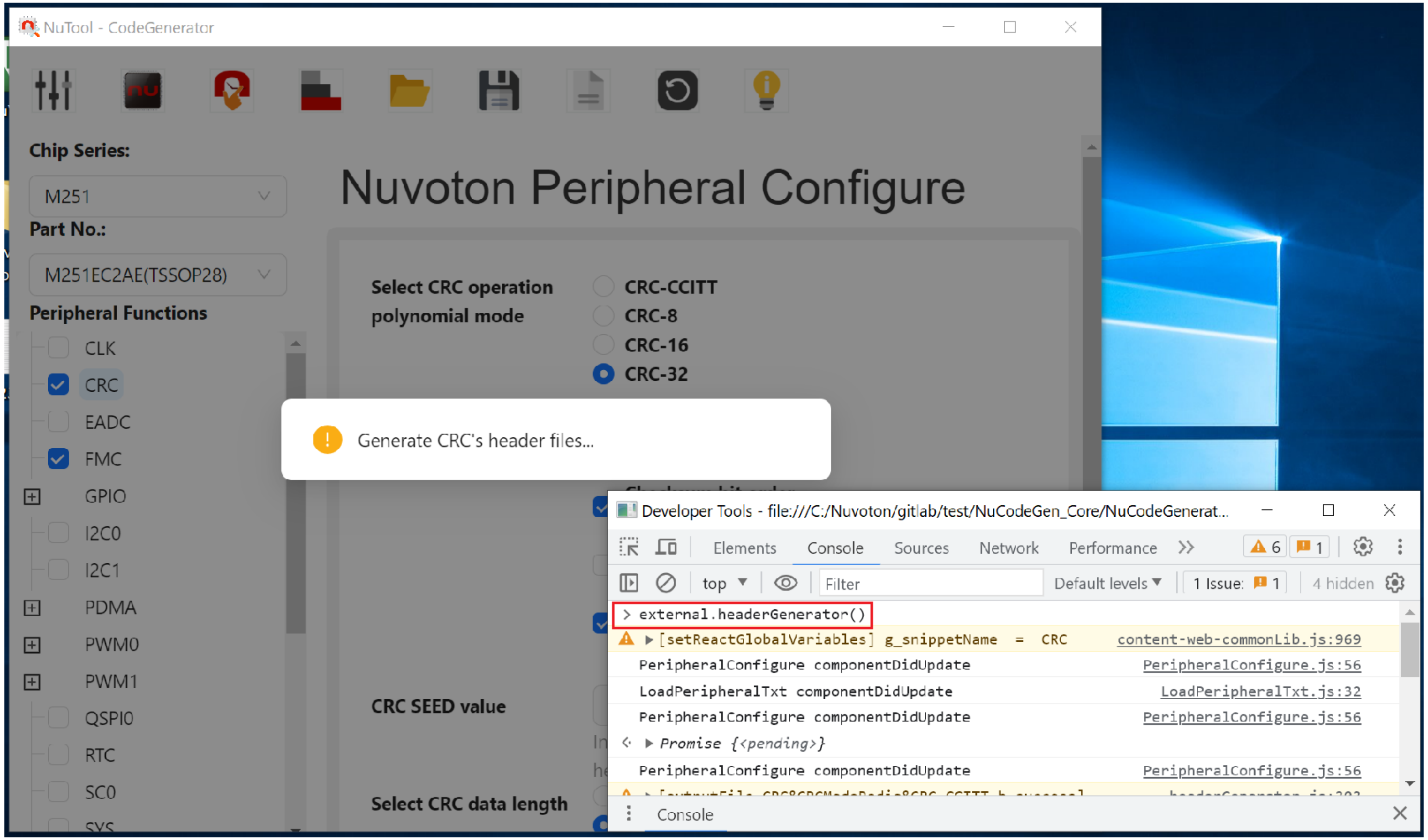


```
parsingexcel.exe version: 0.04
ClockConfig Template FMC LCD EADC DAC0 DAC1 PWM0_CH4CH5 BPWM0 BPWM1 PWM0_CH0CH1
PWM0_CH2CH3 PWM1_CH0CH1 PWM1_CH2CH3 PWM1_CH4CH5 SPII2S0 QSPI0 USCI0_SPI USCI1_SPI USCI2_SPI
PDMA_CH0 PDMA_CH1 PDMA_CH2 PDMA_CH3 PDMA_CH4 PDMA_CH5 PDMA_CH6 PDMA_CH7 EBI_BANK0
EBI_BANK1 I2C0 I2C1 EBI_BANK2 PSIO CRC CRYPTO TIMER0 TIMER1 TIMER2 TIMER3 GPIO_DB GPIO_PA GPIO_PB
GPIO_PC GPIO_PD GPIO_PE GPIO_PF OPA ACMP UART0 UART1 UART2 UART3 USCI0_UART USCI1_UART
USCI2_UART WDT WWDT SYS CLK RTC USBD SC0 USCI0_I2C USCI1_I2C USCI2_I2C
parsing excel complete
```

```
> external.parsingexcel()
<· ▸ Promise {<fulfilled>: undefined}
  [parsingexcel.exe] Finish. parsingexcel.exe version: 0.04     headerGenerator.js:104
  ClockConfig Template FMC LCD EADC DAC0 DAC1 PWM0_CH4CH5 BPWM0 BPWM1 PWM0_CH0CH1
  PWM0_CH2CH3 PWM1_CH0CH1 PWM1_CH2CH3 PWM1_CH4CH5 SPII2S0 QSPI0 USCI0_SPI USCI1_SPI
  USCI2_SPI PDMA_CH0 PDMA_CH1 PDMA_CH2 PDMA_CH3 PDMA_CH4 PDMA_CH5 PDMA_CH6 PDMA_CH7
  EBI_BANK0 EBI_BANK1 I2C0 I2C1 EBI_BANK2 PSIO CRC CRYPTO TIMER0 TIMER1 TIMER2 TIMER3
  GPIO_DB GPIO_PA GPIO_PB GPIO_PC GPIO_PD GPIO_PE GPIO_PF OPA ACMP UART0 UART1 UART2
  UART3 USCI0_UART USCI1_UART USCI2_UART WDT WWDT SYS CLK RTC USBD SC0 USCI0_I2C
  USCI1_I2C USCI2_I2C
  parsing excel complete
>
```

Functional Tests

Step 6: Input external.headerGenerator().



Nuvoton Peripheral Configure

Select CRC operation polynomial mode
○ CRC-CCITT
○ CRC-8
○ CRC-16
● CRC-32

CRC SEED value

Select CRC data length

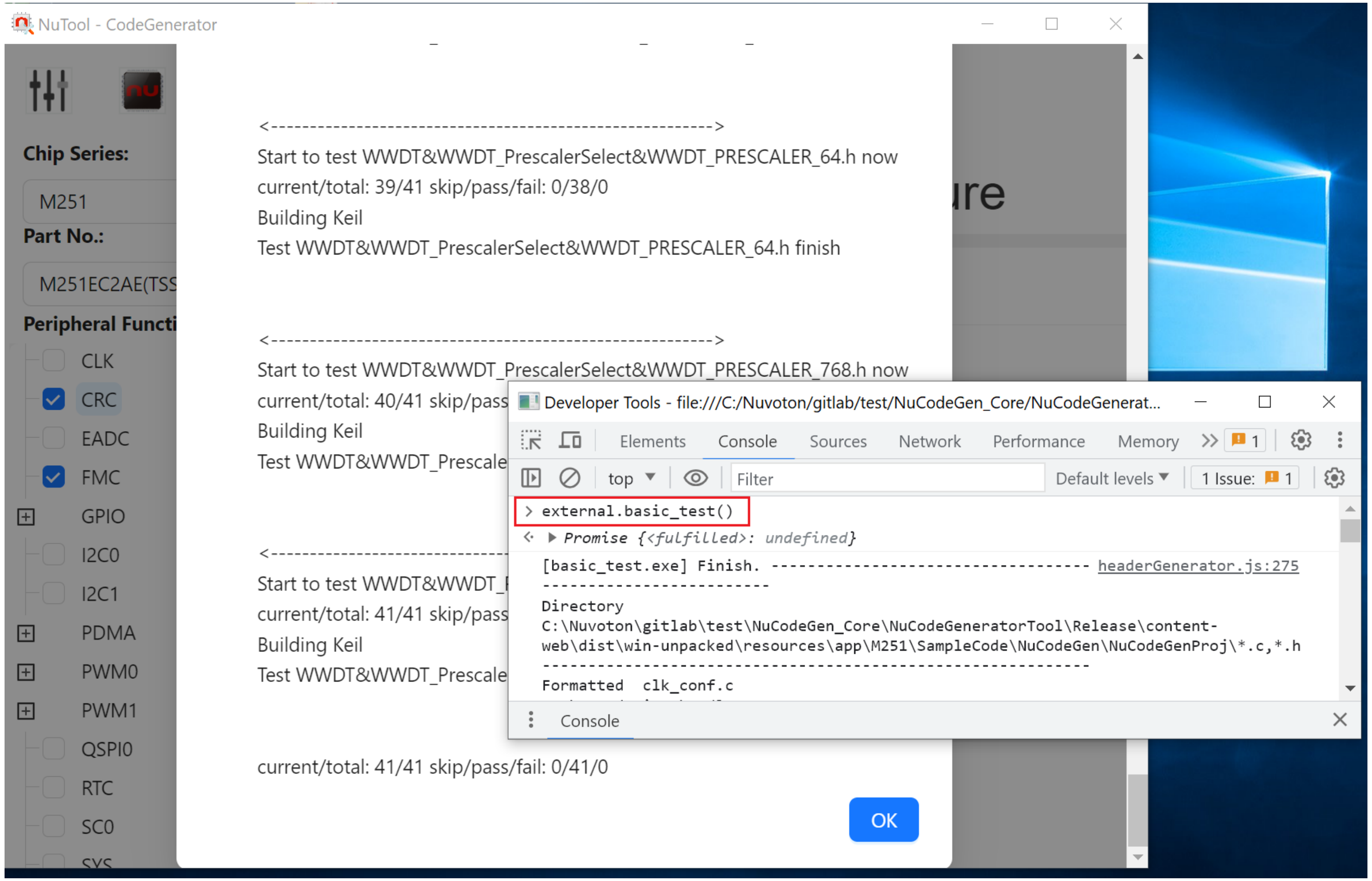
Generate CRC's header files...

Developer Tools - file:///C:/Nuvoton/gitlab/test/NuCodeGen_Core/NuCodeGenerat...

```
> external.headerGenerator()
⚠ ▶[setReactGlobalVariables] g_snippetName  =  CRC      content-web-commonLib.js:969
  PeripheralConfigure componentDidUpdate                PeripheralConfigure.js:56
  LoadPeripheralTxt componentDidUpdate                    LoadPeripheralTxt.js:32
  PeripheralConfigure componentDidUpdate                PeripheralConfigure.js:56
< ▶ Promise {<pending>}
  PeripheralConfigure componentDidUpdate                PeripheralConfigure.js:56
```

Functional Tests

Step 7: Input external.basic_test().

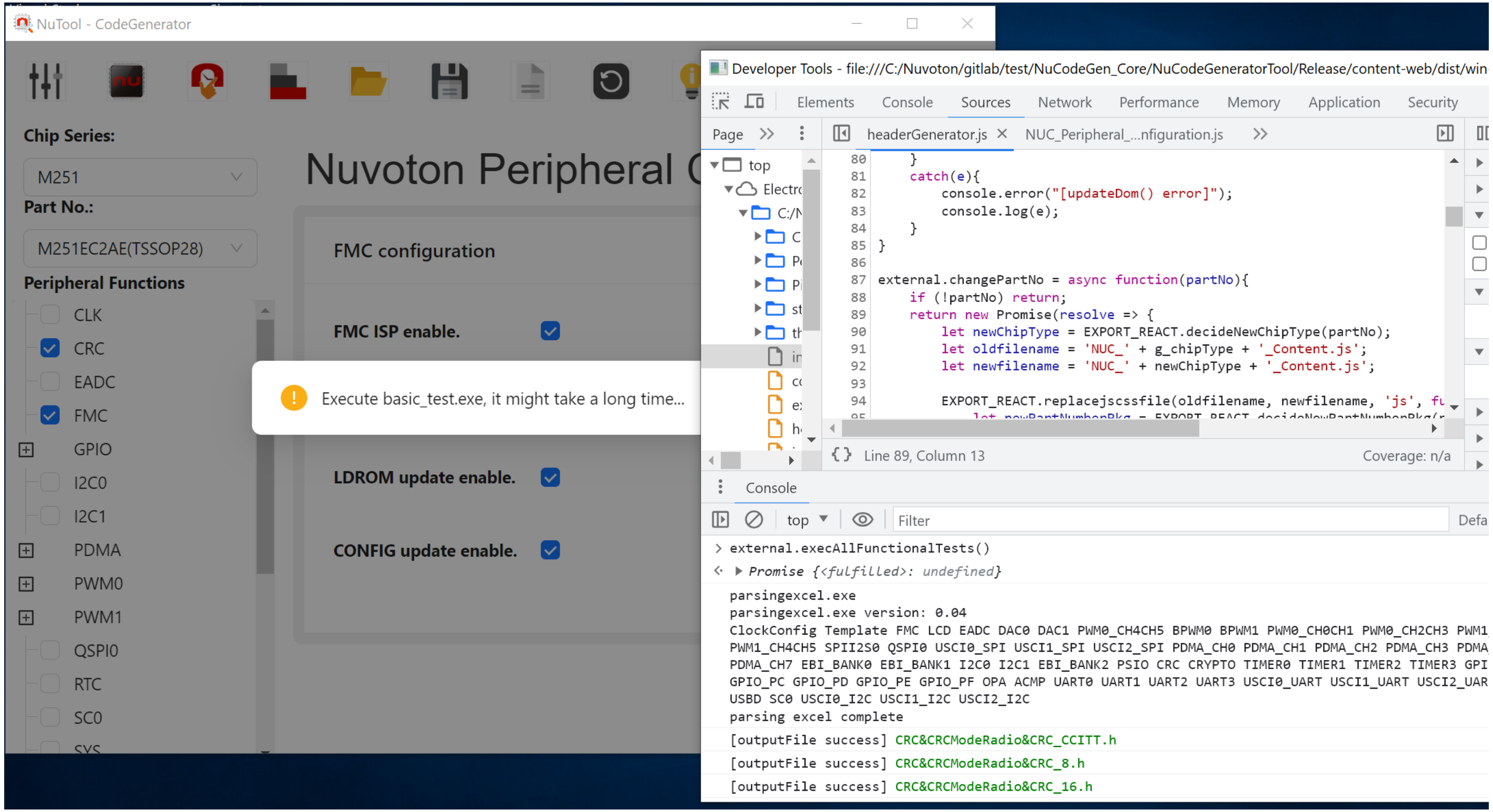


```
<------------------------------------------------------------->
Start to test WWDT&WWDT_PrescalerSelect&WWDT_PRESCALER_64.h now
current/total: 39/41 skip/pass/fail: 0/38/0
Building Keil
Test WWDT&WWDT_PrescalerSelect&WWDT_PRESCALER_64.h finish

<------------------------------------------------------------->
Start to test WWDT&WWDT_PrescalerSelect&WWDT_PRESCALER_768.h now
current/total: 40/41 skip/pass
Building Keil
Test WWDT&WWDT_Prescale

<------------------------------------
Start to test WWDT&WWDT_
current/total: 41/41 skip/pass
Building Keil
Test WWDT&WWDT_Prescale

current/total: 41/41 skip/pass/fail: 0/41/0
```

```
> external.basic_test()
<· ▶ Promise {<fulfilled>: undefined}
  [basic_test.exe] Finish. ---------------------------------- headerGenerator.js:275
  ------------------------
  Directory
  C:\Nuvoton\gitlab\test\NuCodeGen_Core\NuCodeGeneratorTool\Release\content-
  web\dist\win-unpacked\resources\app\M251\SampleCode\NuCodeGen\NuCodeGenProj\*.c,*.h
  ----------------------------------------------------------
  Formatted  clk_conf.c
```

Functional Tests

Note 1: Input external.execAllFunctionalTests() instead of Step 5 + Step 6 + Step 7



FMC configuration

FMC ISP enable. ☑

LDROM update enable. ☑

CONFIG update enable. ☑

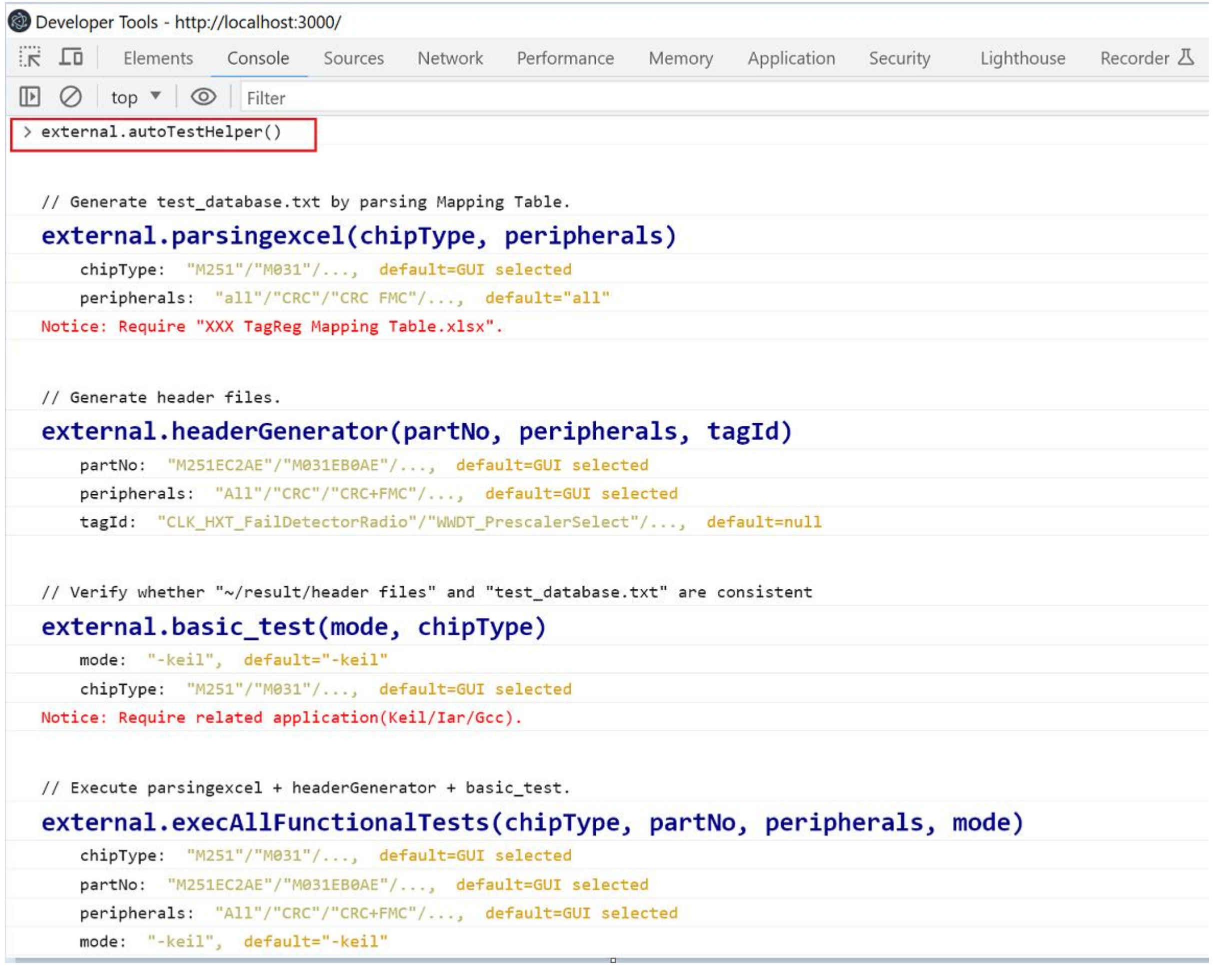
Execute basic_test.exe, it might take a long time...

```
    }
    catch(e){
        console.error("[updateDom() error]");
        console.log(e);
    }
}

external.changePartNo = async function(partNo){
    if (!partNo) return;
    return new Promise(resolve => {
        let newChipType = EXPORT_REACT.decideNewChipType(partNo);
        let oldfilename = 'NUC_' + g_chipType + '_Content.js';
        let newfilename = 'NUC_' + newChipType + '_Content.js';

        EXPORT_REACT.replacejscssfile(oldfilename, newfilename, 'js', fu
```

```
> external.execAllFunctionalTests()
<· ▶ Promise {<fulfilled>: undefined}
  parsingexcel.exe
  parsingexcel.exe version: 0.04
  ClockConfig Template FMC LCD EADC DAC0 DAC1 PWM0_CH4CH5 BPWM0 BPWM1 PWM0_CH0CH1 PWM0_CH2CH3 PWM1
  PWM1_CH4CH5 SPII2S0 QSPI0 USCI0_SPI USCI1_SPI USCI2_SPI PDMA_CH0 PDMA_CH1 PDMA_CH2 PDMA_CH3 PDMA
  PDMA_CH7 EBI_BANK0 EBI_BANK1 I2C0 I2C1 EBI_BANK2 PSIO CRC CRYPTO TIMER0 TIMER1 TIMER2 TIMER3 GPI
  GPIO_PC GPIO_PD GPIO_PE GPIO_PF OPA ACMP UART0 UART1 UART2 UART3 USCI0_UART USCI1_UART USCI2_UAR
  USBD SC0 USCI0_I2C USCI1_I2C USCI2_I2C
  parsing excel complete
  [outputFile success] CRC&CRCModeRadio&CRC_CCITT.h
  [outputFile success] CRC&CRCModeRadio&CRC_8.h
  [outputFile success] CRC&CRCModeRadio&CRC_16.h
```

Functional Tests

Note 2: Input external.autoTestHelper() can get **the latest usage**.



```
> external.autoTestHelper()


  // Generate test_database.txt by parsing Mapping Table.
  external.parsingexcel(chipType, peripherals)
      chipType:  "M251"/"M031"/...,  default=GUI selected
      peripherals:  "all"/"CRC"/"CRC FMC"/...,  default="all"
  Notice: Require "XXX TagReg Mapping Table.xlsx".


  // Generate header files.
  external.headerGenerator(partNo, peripherals, tagId)
      partNo:  "M251EC2AE"/"M031EB0AE"/...,  default=GUI selected
      peripherals:  "All"/"CRC"/"CRC+FMC"/...,  default=GUI selected
      tagId:  "CLK_HXT_FailDetectorRadio"/"WWDT_PrescalerSelect"/...,  default=null


  // Verify whether "~/result/header files" and "test_database.txt" are consistent
  external.basic_test(mode, chipType)
      mode:  "-keil",  default="-keil"
      chipType:  "M251"/"M031"/...,  default=GUI selected
  Notice: Require related application(Keil/Iar/Gcc).


  // Execute parsingexcel + headerGenerator + basic_test.
  external.execAllFunctionalTests(chipType, partNo, peripherals, mode)
      chipType:  "M251"/"M031"/...,  default=GUI selected
      partNo:  "M251EC2AE"/"M031EB0AE"/...,  default=GUI selected
      peripherals:  "All"/"CRC"/"CRC+FMC"/...,  default=GUI selected
      mode:  "-keil",  default="-keil"
```