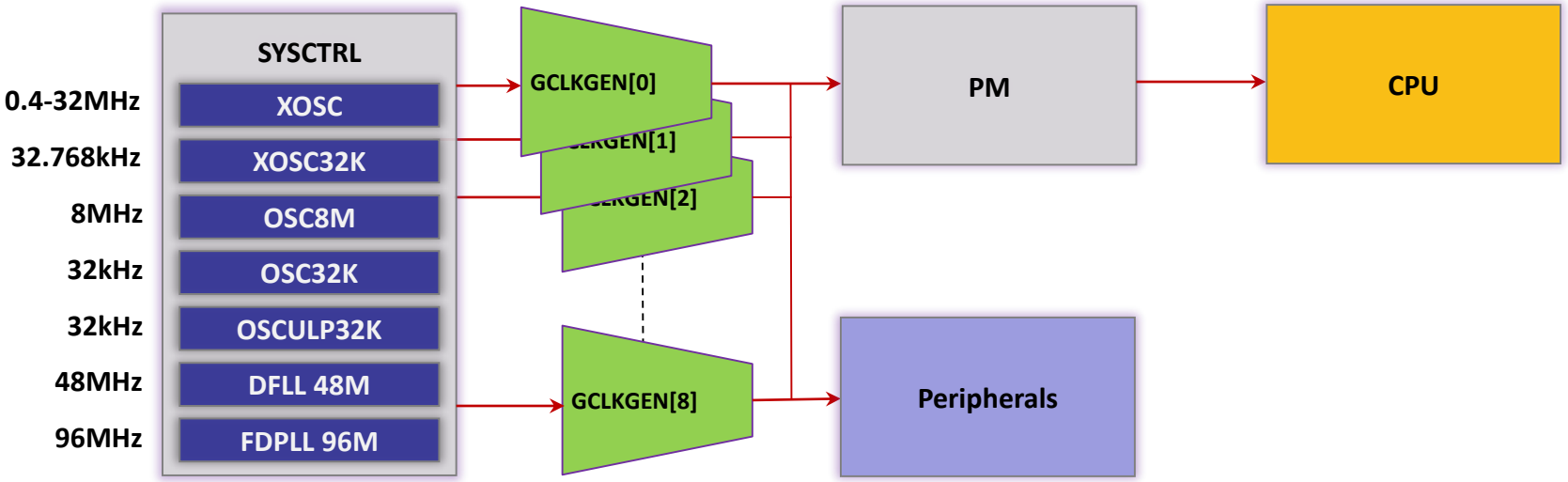
MICROCHIP

Regional Training Centers

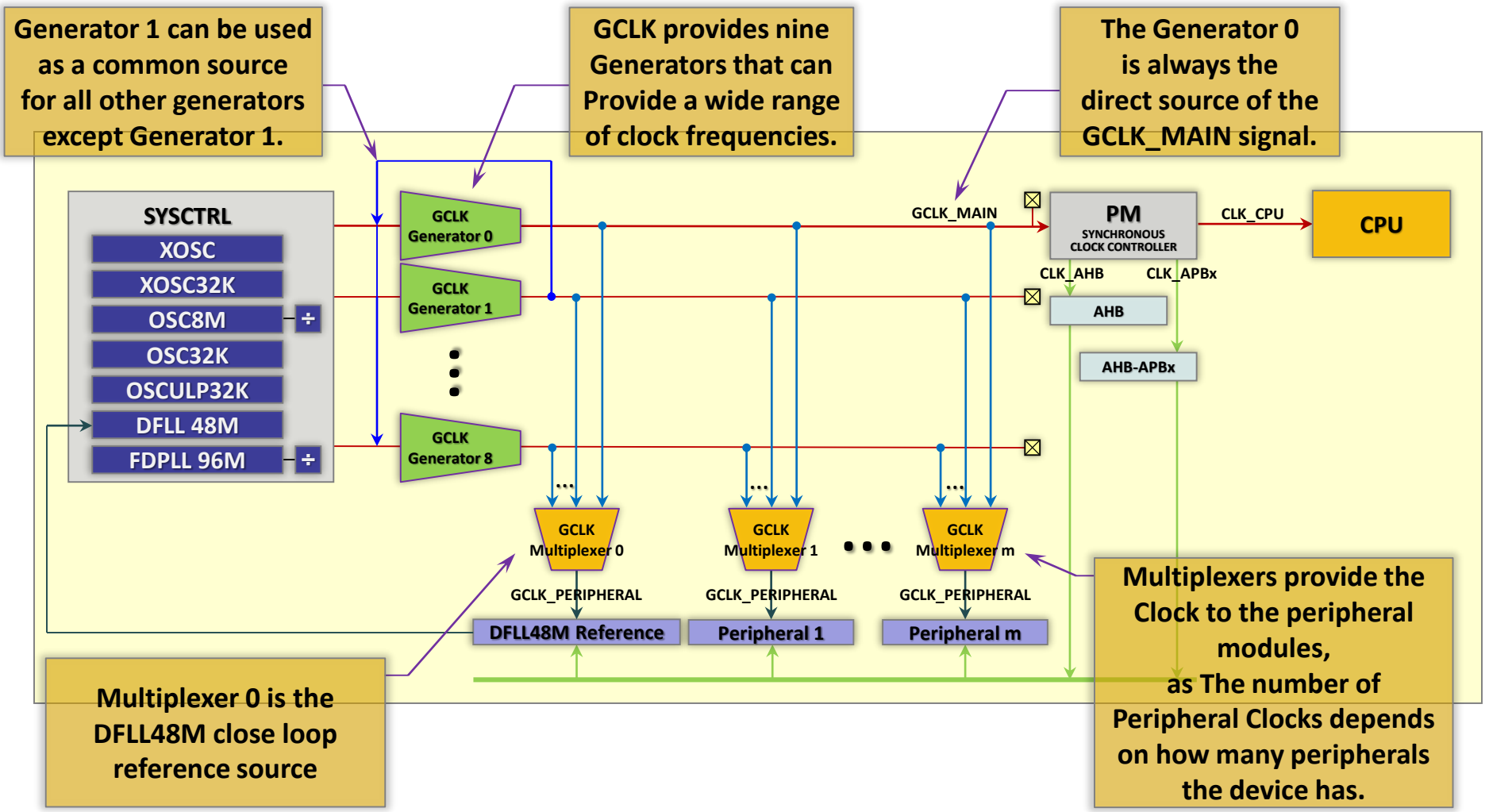
Section 8
Clock System Architecture

Clock System

- Review previously section, SAMD21's clock system provides clock sources to the Generic Clock Controller。The available clock sources are XOSC, XOSC32K, OSC32K, OSCULP32K, OSC8M,DFLL48M and FDPLL96M。
- The bus clock can be enabled and disabled in the Power Manager(PM) include CPU Clock(GCLK_MAIN)。

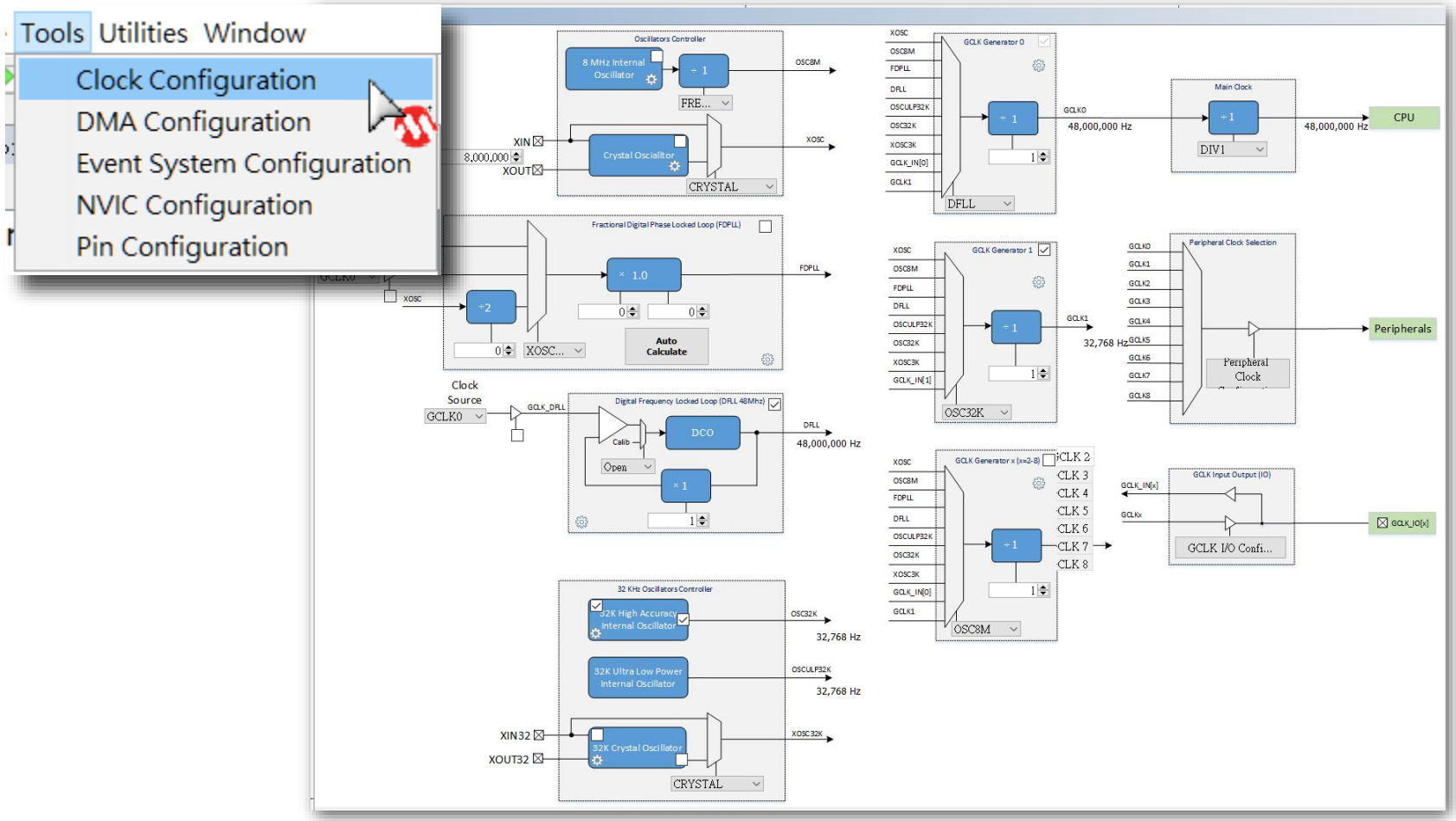


Clock System Block Diagram



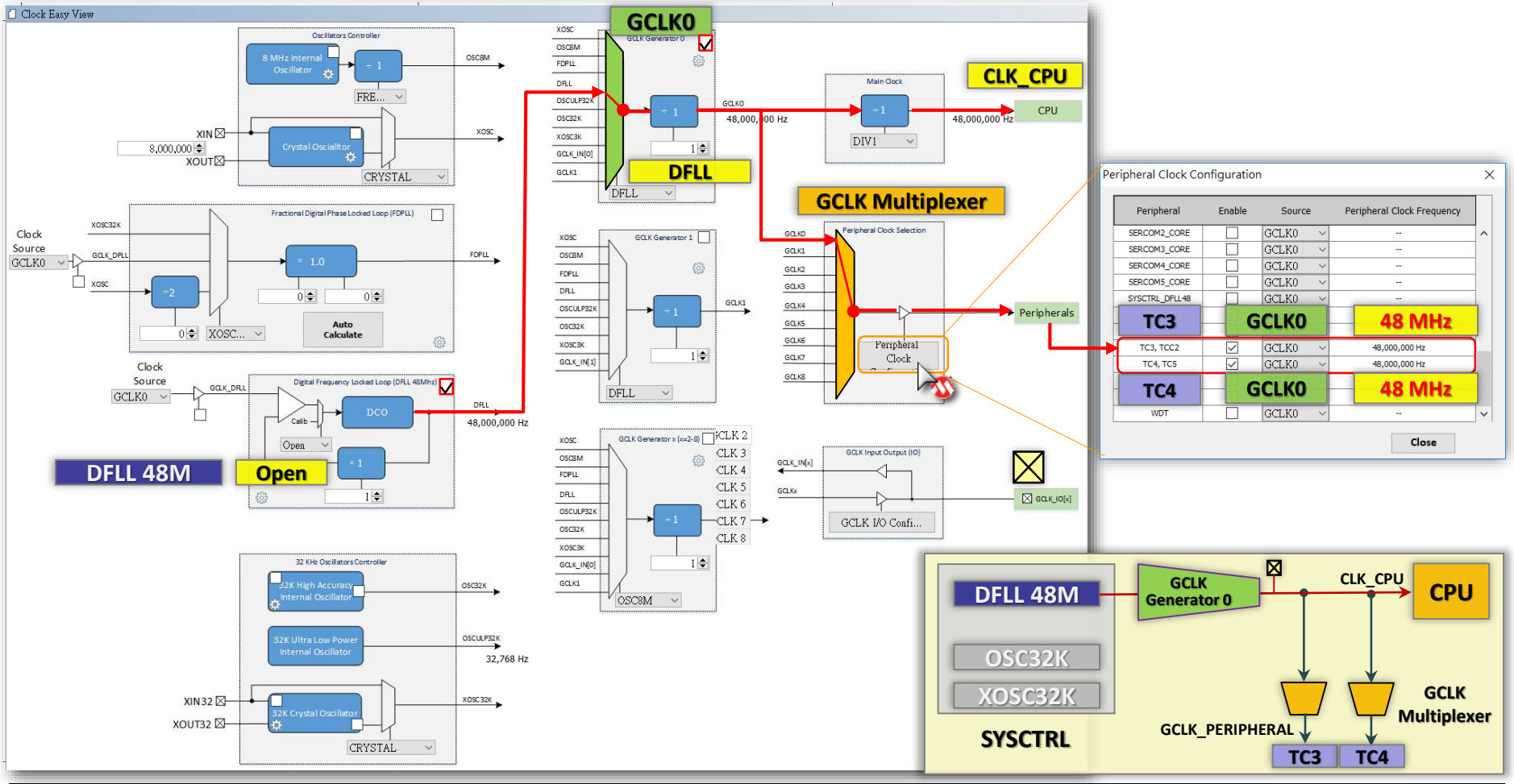
Clock Setting Use MH3

- Open Clock Easy View windows in **Tools ▸ Clock Configuration**.



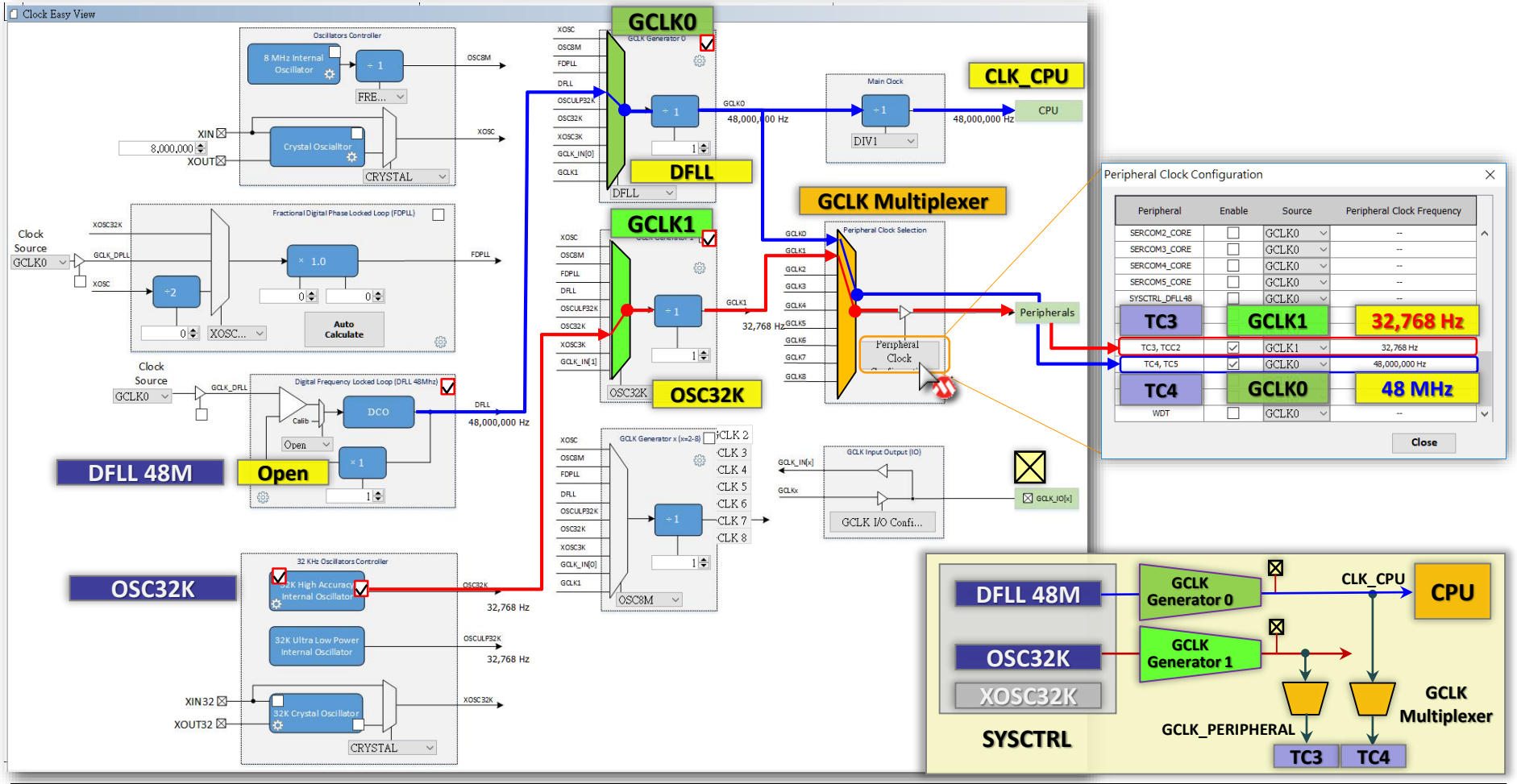
Clock Setting Use MH3

- Default clock setting of new project (DFLL48M open loop).



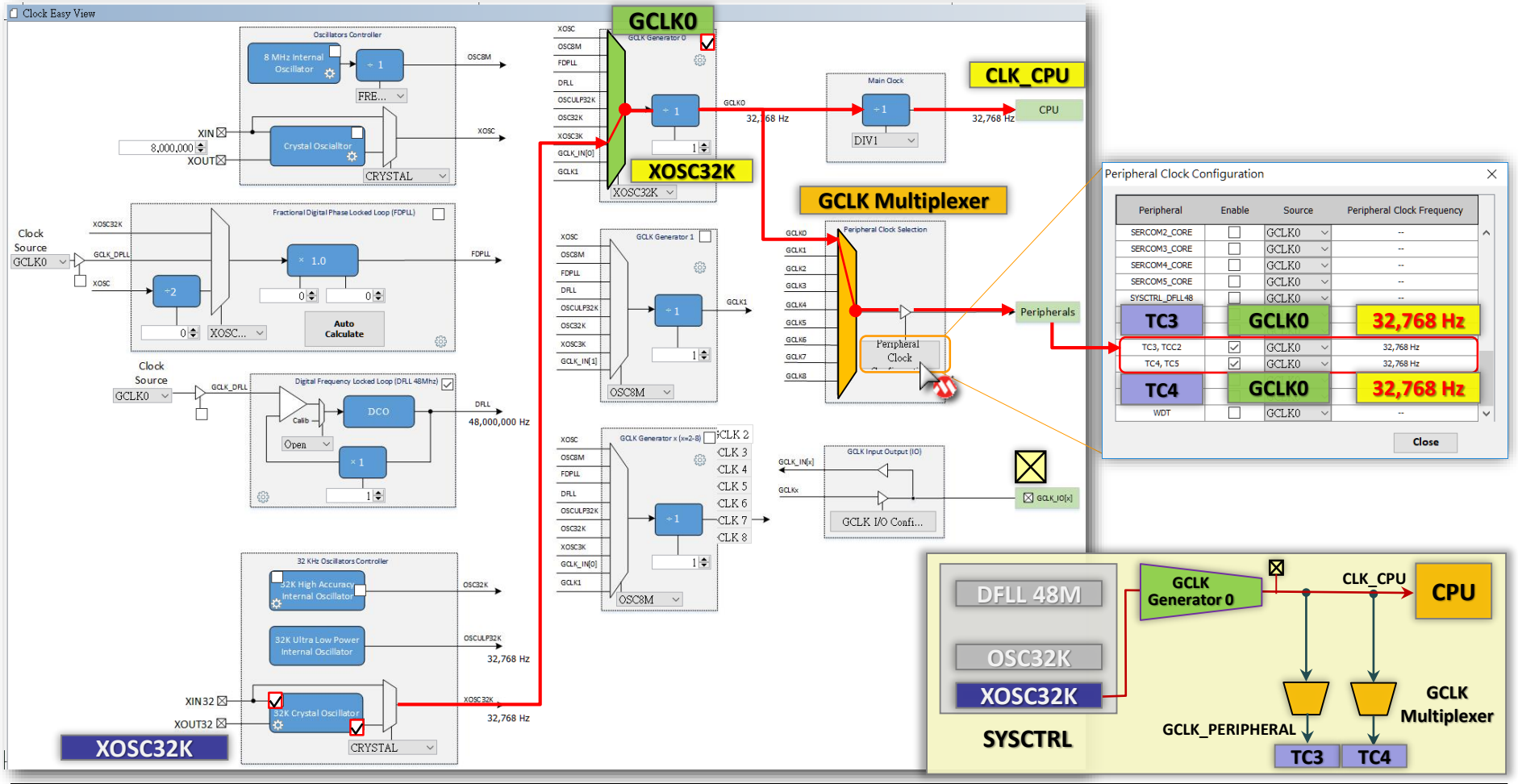
Clock Setting Use MH3

- For Example : Internal 32k RC -> GCLK Generator 1 -> TC3



Clock Setting Use MH3

- For Example : External 32.768K -> Generator 0 -> CPU & TC3



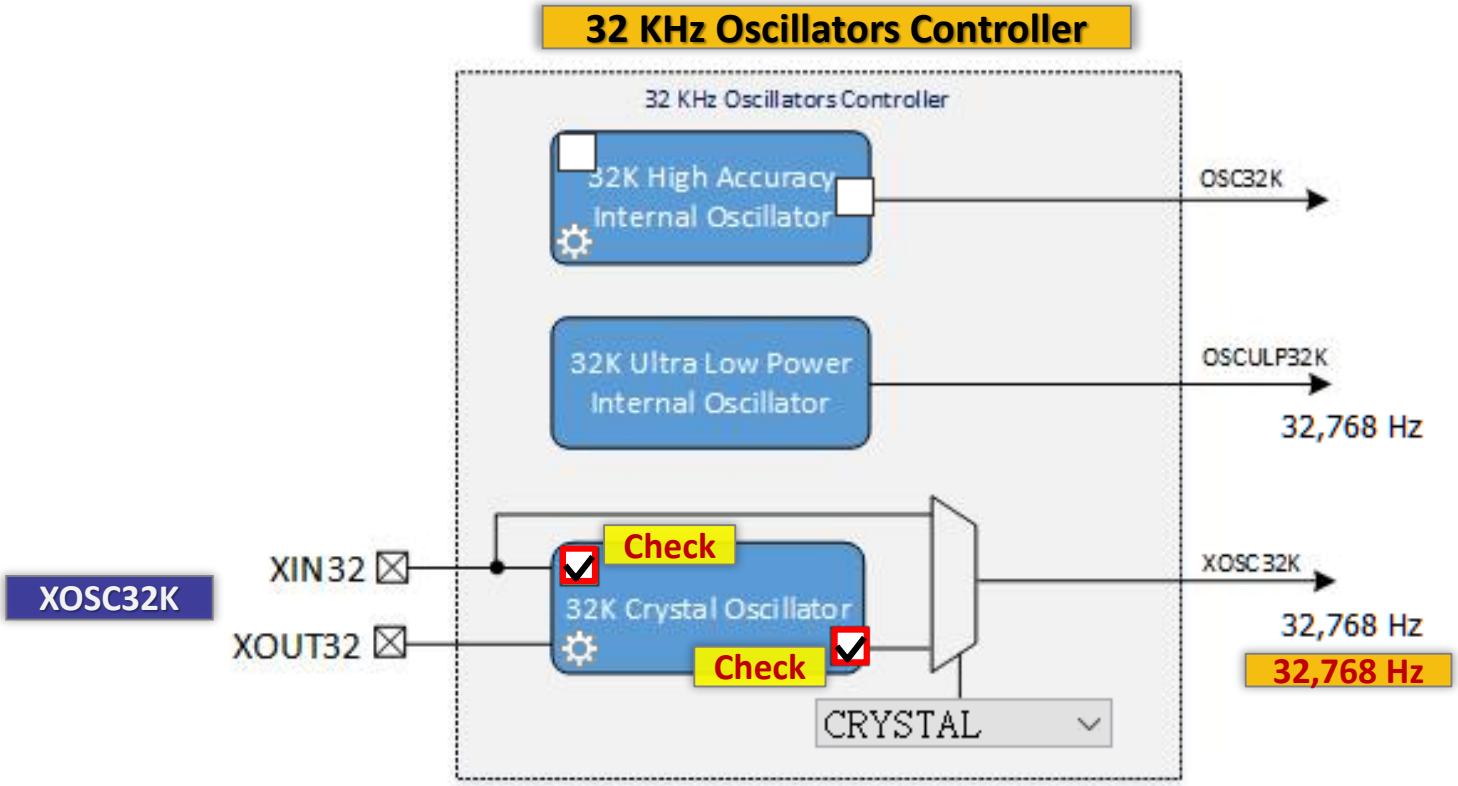
Lab7 System Clock XOSC32K

- Try to change main clock from default source to external 32.768K.
- Setting XOSC32K then enable it, first. Then Connect XOSC32K to Generator 0. Connect TC3, TC4 to Generator 0.

- Let's go!

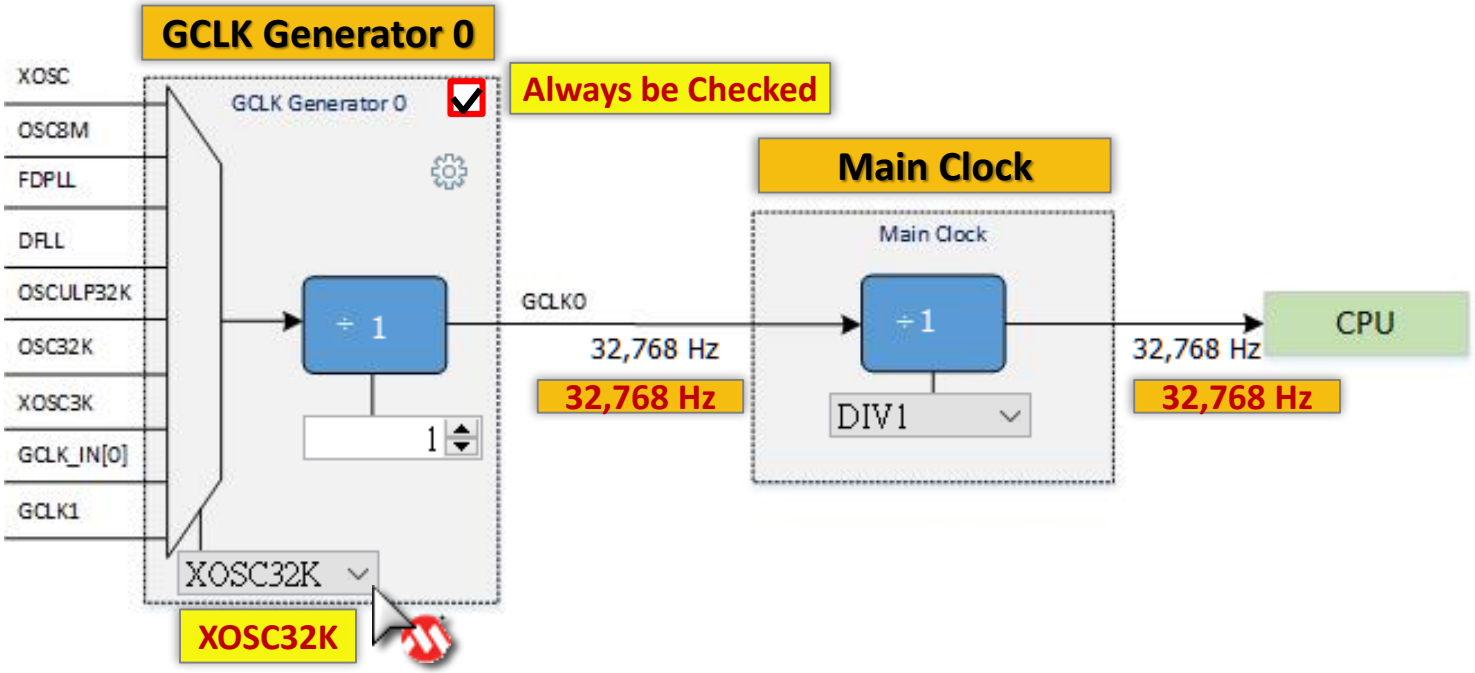
Lab7 System Clock XOSC32K

Step 1



Lab7 System Clock XOSC32K

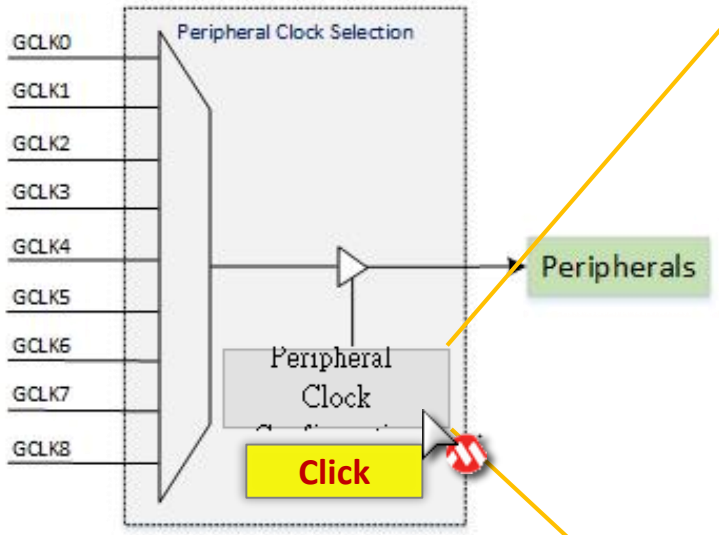
Step 2



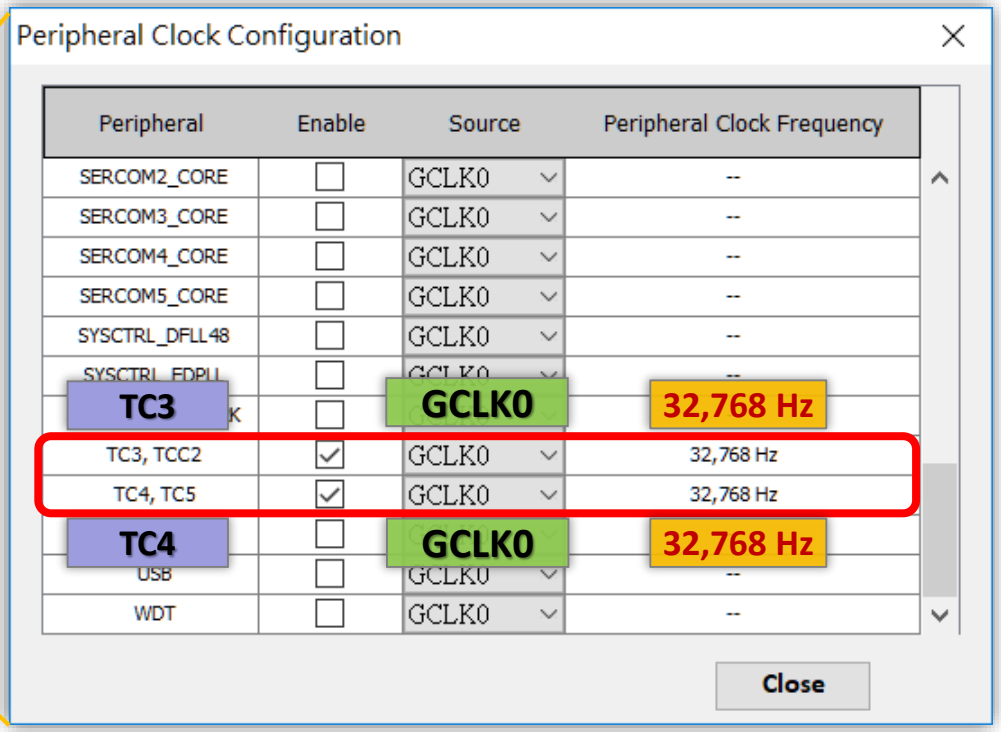
Lab7 System Clock XOSC32K

Step 3

Peripheral Clock Selection



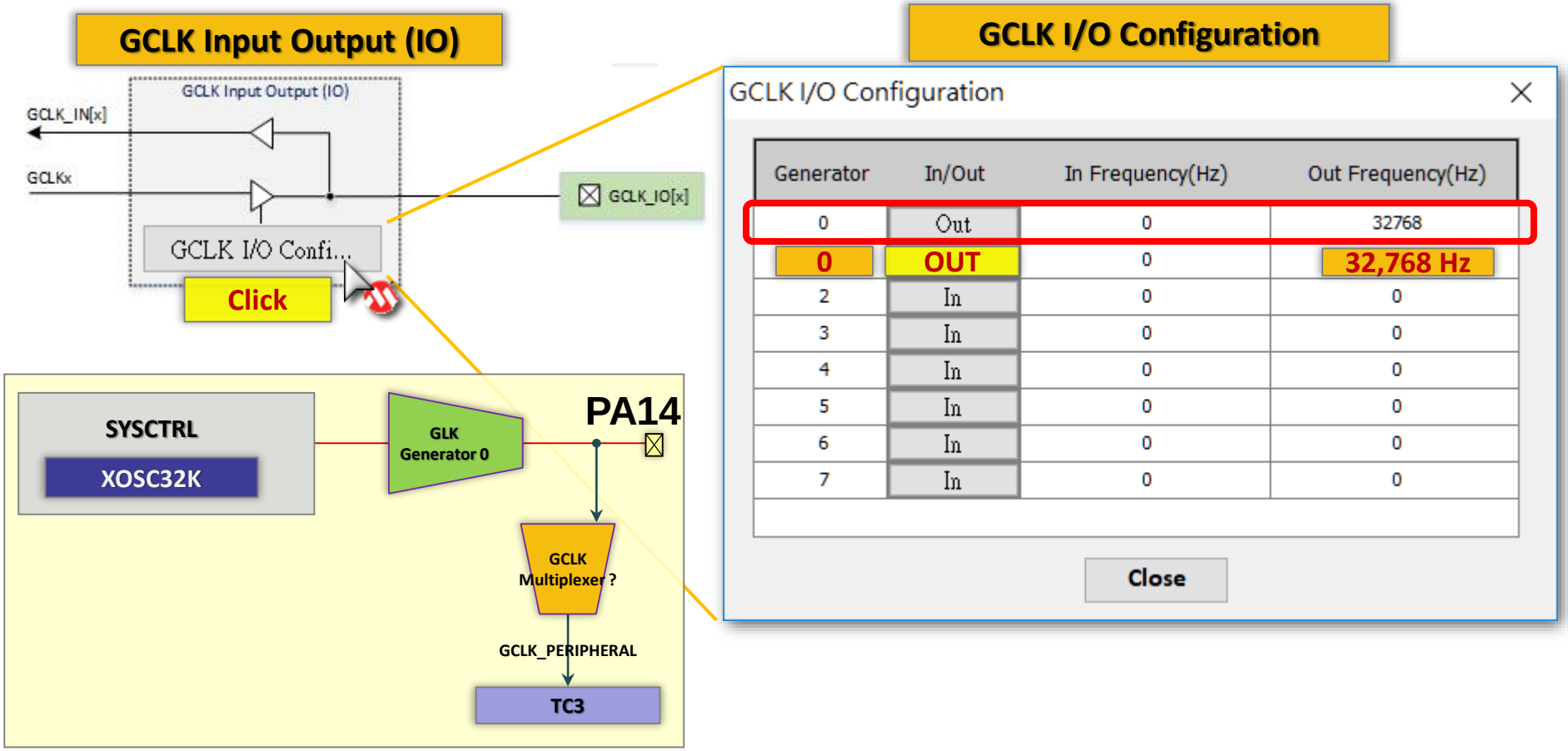
Peripheral Clock Configuration



Lab7 System Clock XOSC32K

Step 4

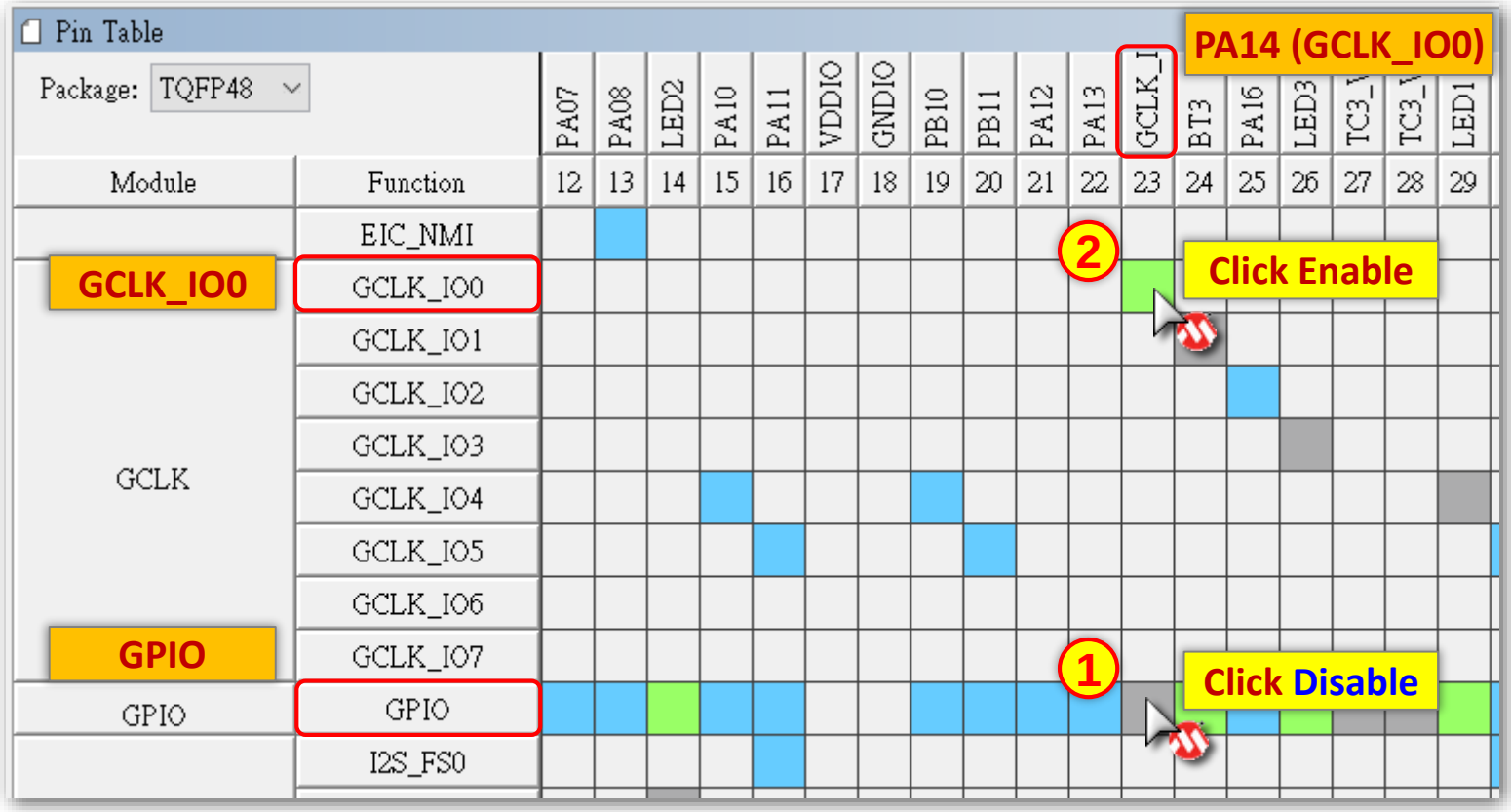
- Configure GCLK Generator 0 to output clock.



Lab7 System Clock XOSC32K

Step 5

- Please disable BT2(PA14) GPIO pin function firstly and then reassign PA14 as GCLK_IO0 output pin.

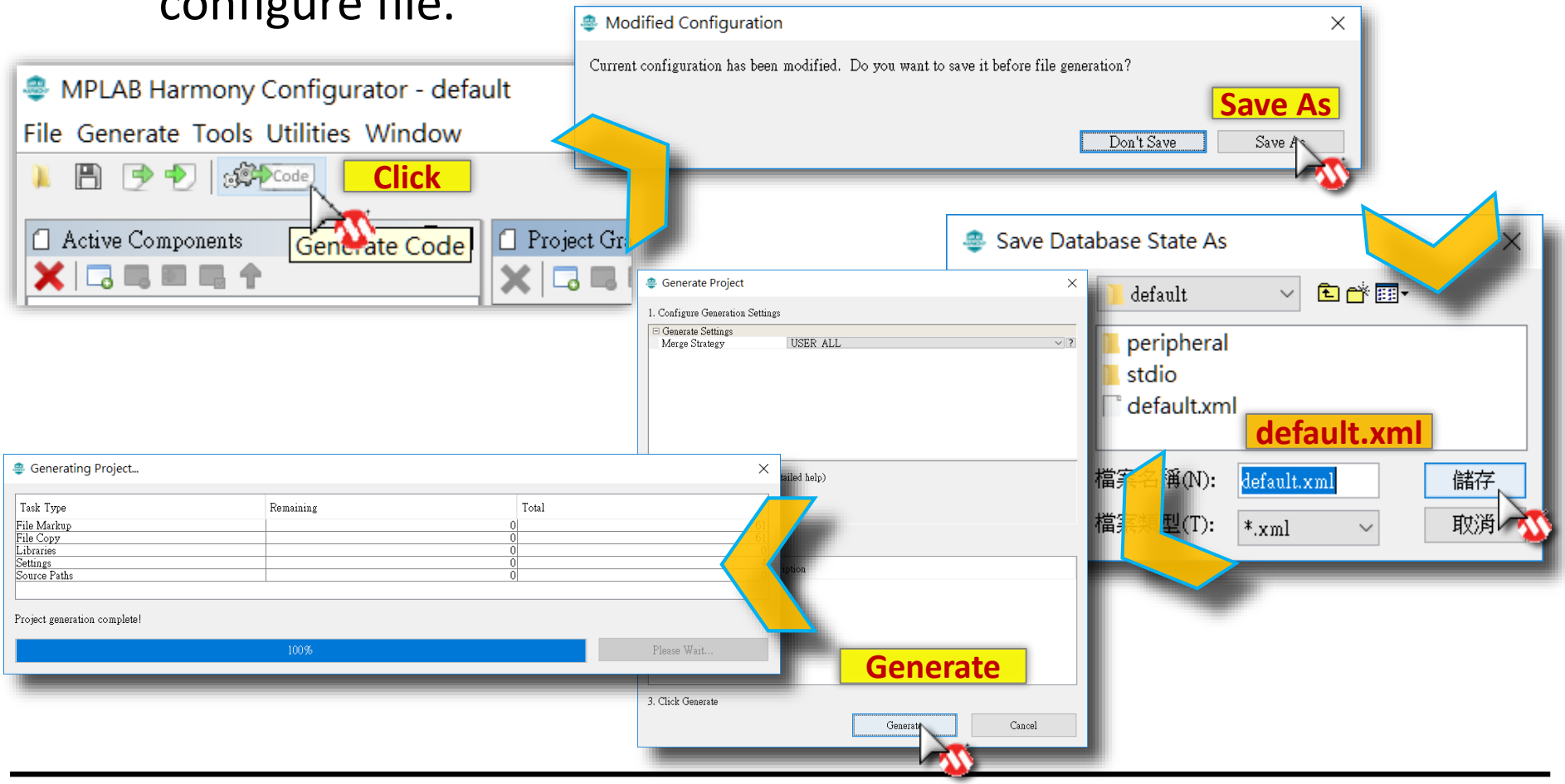


Lab7 System Clock XOSC32K

Step 6

- Click 

to Generate Code and save changes to MHC configure file.



Lab7 System Clock XOSC32K

Step 7

a. **Comment out BT2(PA14) related GPIO code segment.**

```
…

int main (void)
{
   SYS_Initialize ( NULL );

   while(1)
   {
      …

      // TODO 7.01
      // if ( BT2_Get() ) LED3_Clear();
      // else             LED3_Set();
   }
}
```

Flash Access Time and Wait State

- The SAMD21 main flash Random Read time around 40nS. That means you must adjust Wait State to proper value if main clock over than 24MHz. The actual value shows in below, it will depend on system supply voltage.

V_{DD} range	NVM Wait States	Maximum Operating Frequency	Units
1.62V to 2.7V	0	14	MHz
	1	28	
	2	42	
	3	48	
2.7V to 3.63V	0	24	
	1	48	

CPU_CLK = 32.768KHz

VDD = 3.3V

MHC will auto adjust proper NVM Wait State.
(In case of 3.3V voltage supply only).

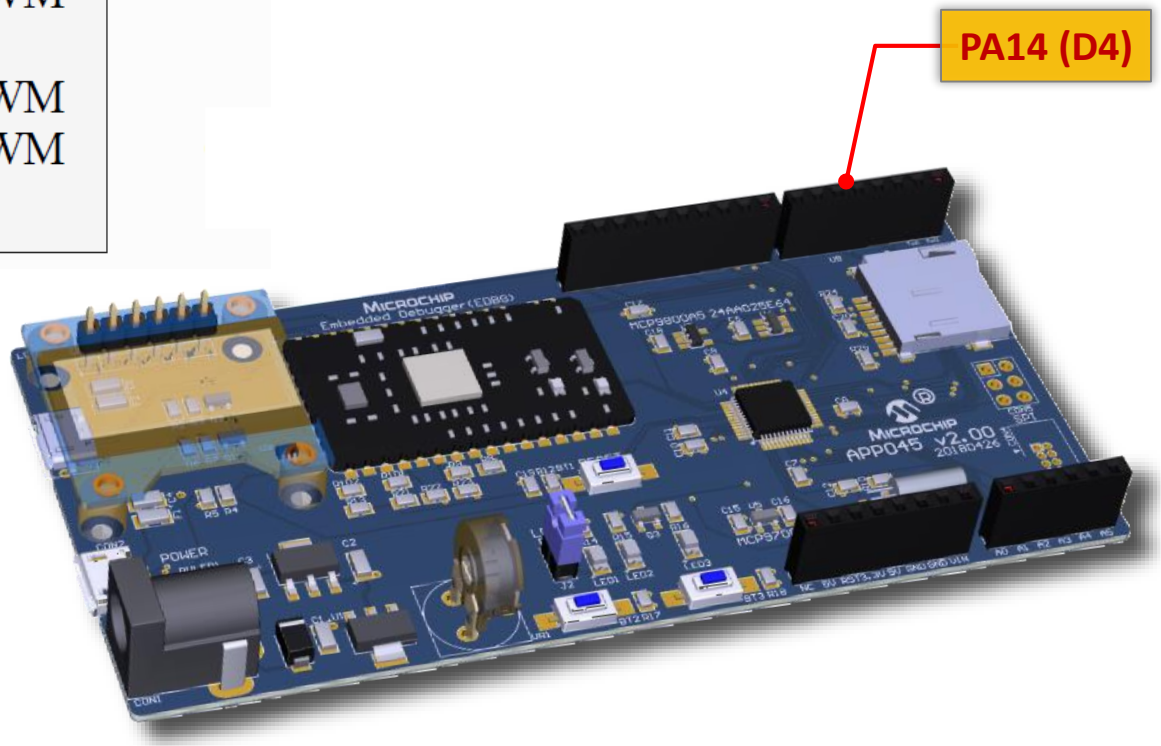
plib_nvmctrl.c

```
void NVMCTRL_Initialize(void)
{
    NVMCTRL_REGS->NVMCTRL_CTRLB =
        NVMCTRL_CTRLB_READMODE_NO_MISS_PENALTY |
        NVMCTRL_CTRLB_SLEEPPRM_WAKEONACCESS |
        NVMCTRL_CTRLB_RWS(0) |
        NVMCTRL_CTRLB_MANW_Msk;
}
```

Lab7 System Clock XOSC32K

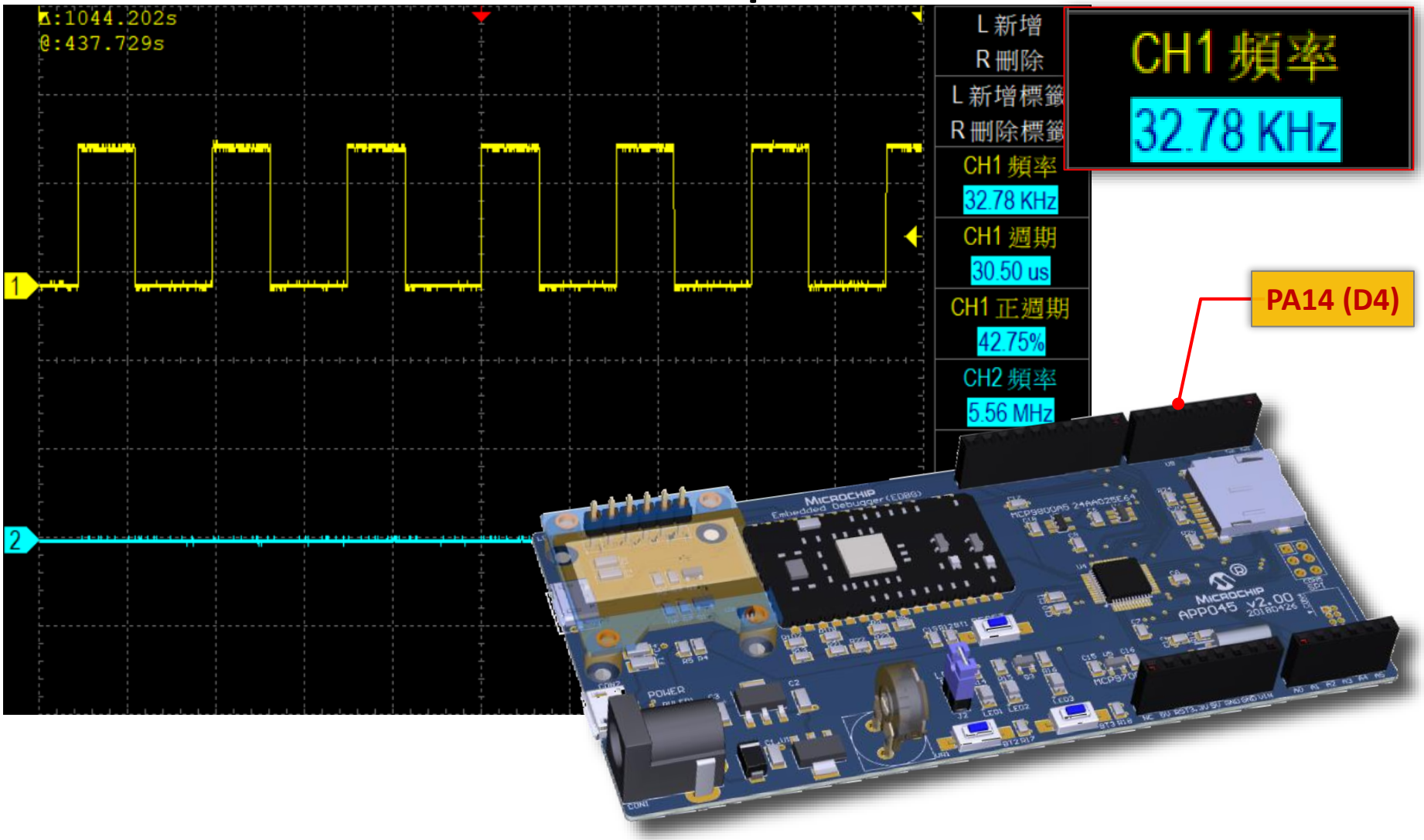
Clock Output

		CON9B Arduino UNO Socket
PA11	7	D0/RX
PA10	8	D1/TX
PA08	9	D2
PA09	10	D3/PWM
PA14	11	D4
PA15	12	D5/PWM
PA20	13	D6/PWM
EDBG GPIO	14	D7



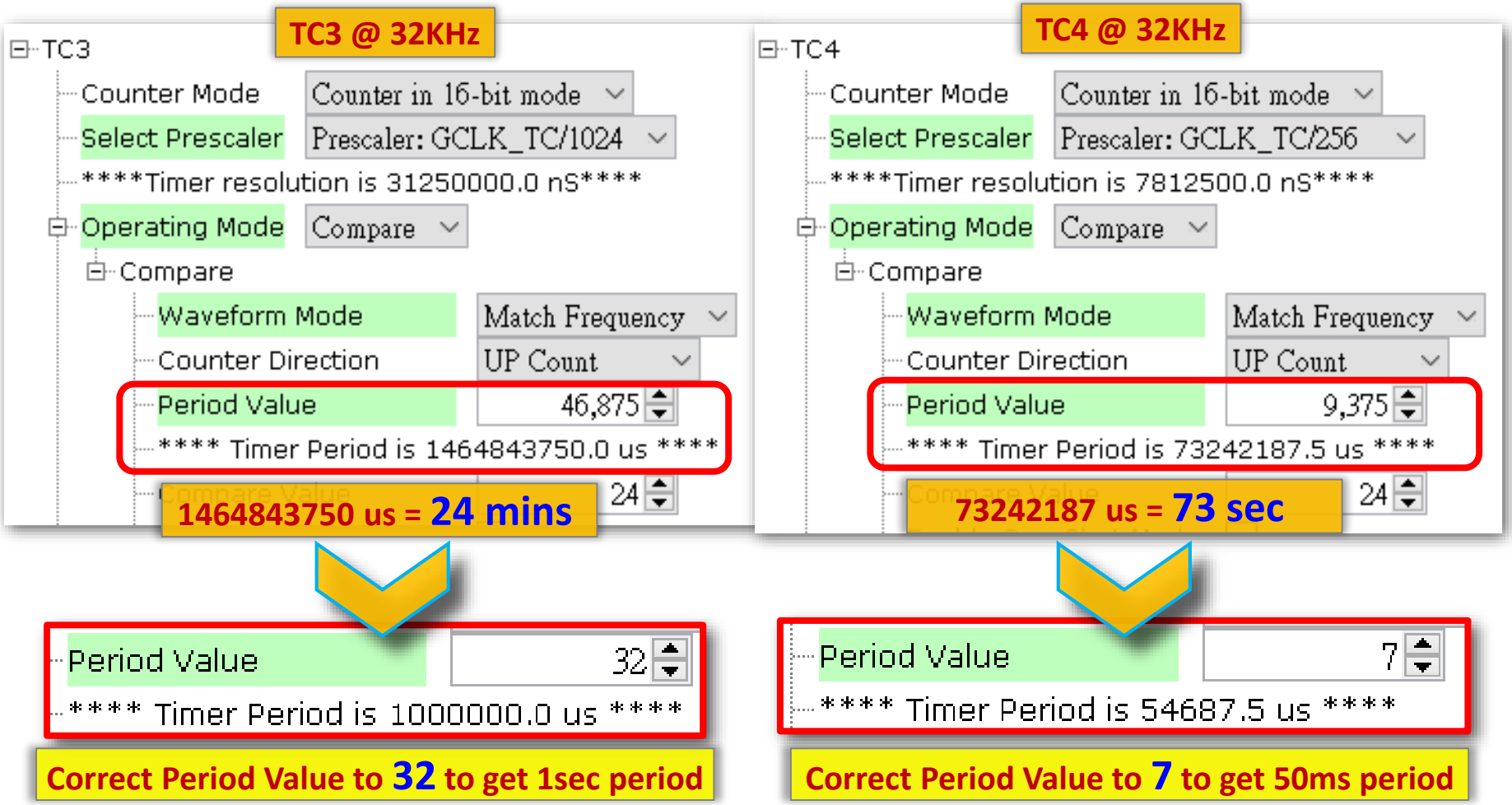
Lab7 System Clock XOSC32K

Clock Output



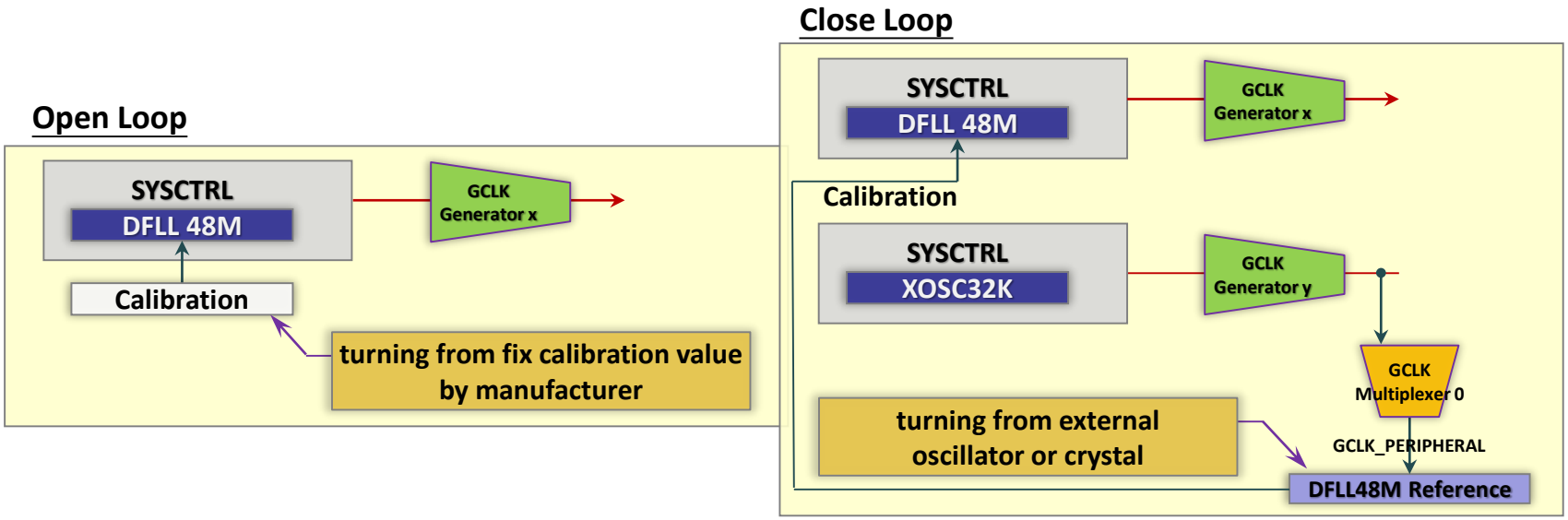
Lab7 System Clock XOSC32K

Step 8 LED toggle period correct



Clock source from DFLL48M

- There are 2 different mode for DFLL48M clock source
  - Open Loop : **Project Default Use**
    Clock turning from fix calibration value by manufacturer.
  - Close Loop :
    Clock turning from external oscillator or crystal.



Lab8 System Clock DFLL48M

Close Loop

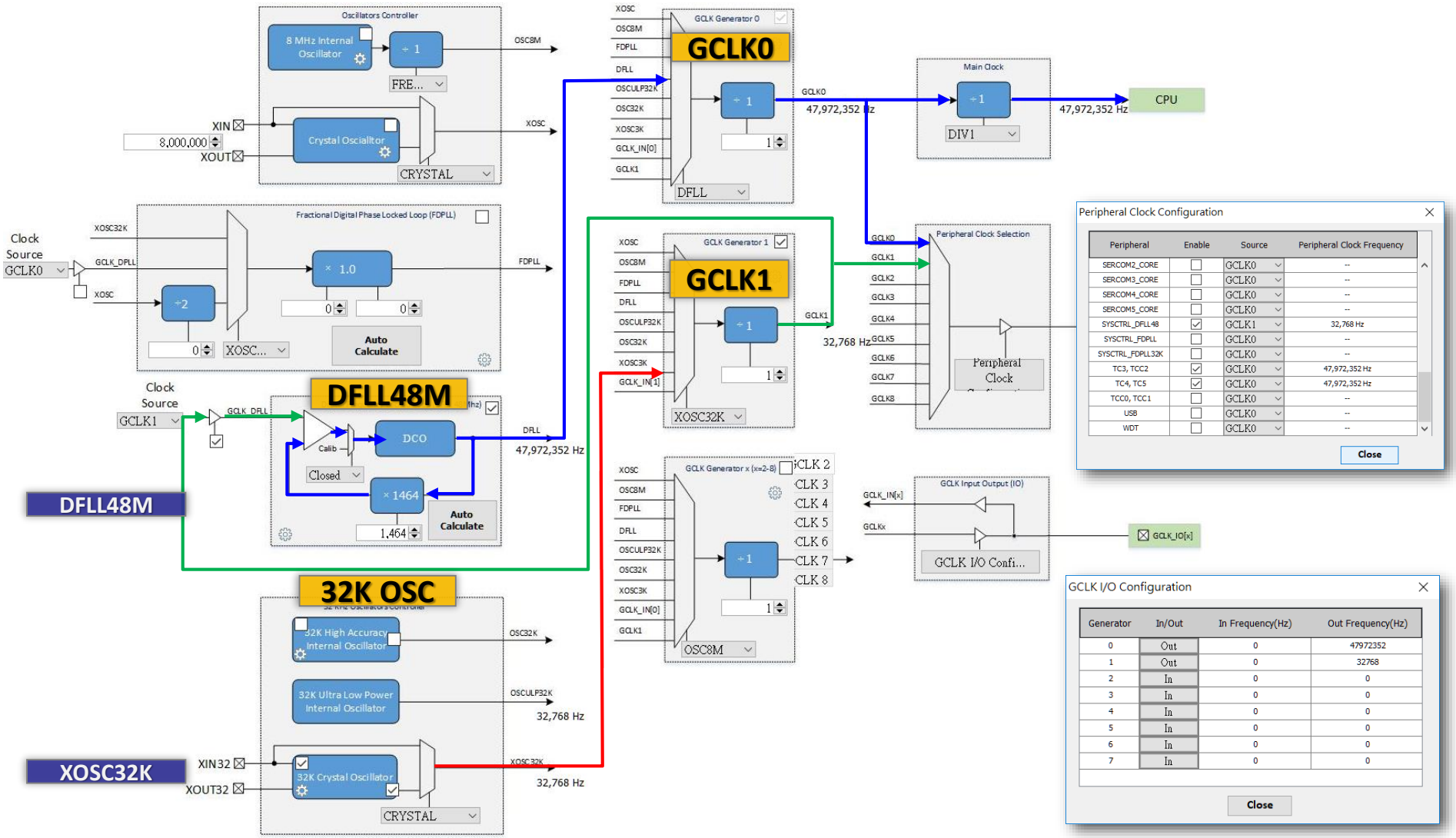
Configure DFLL48M(Close loop) as source of main clock(GCLK0).

- Enable XOSC32K clock source.
- Enable GCLK Generator 1 and assign XOSC32K as source of it.
- Enable DFLL48M and assign GCLK1 as source or it.
- Select Close Loop calibration of DFLL48M.
- Click Auto Calculate to generate multiplier factor of DFLL48M.
- Assign DFLL48M as clock source of GCLK Generator 0.
- CPU wait state will auto judge by MHC.

- Let's go!

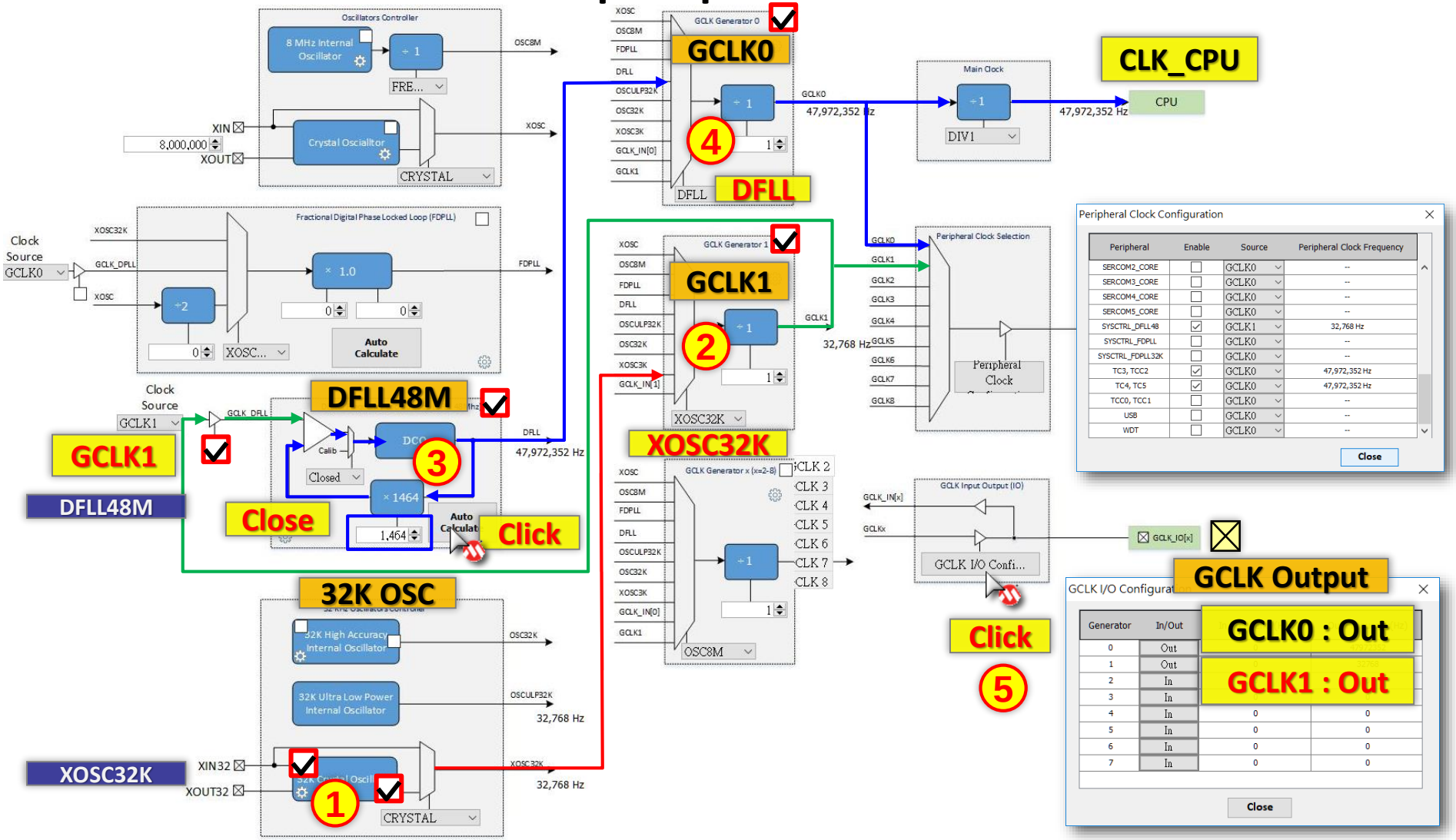
Lab8 System Clock DFLL48M

Overview



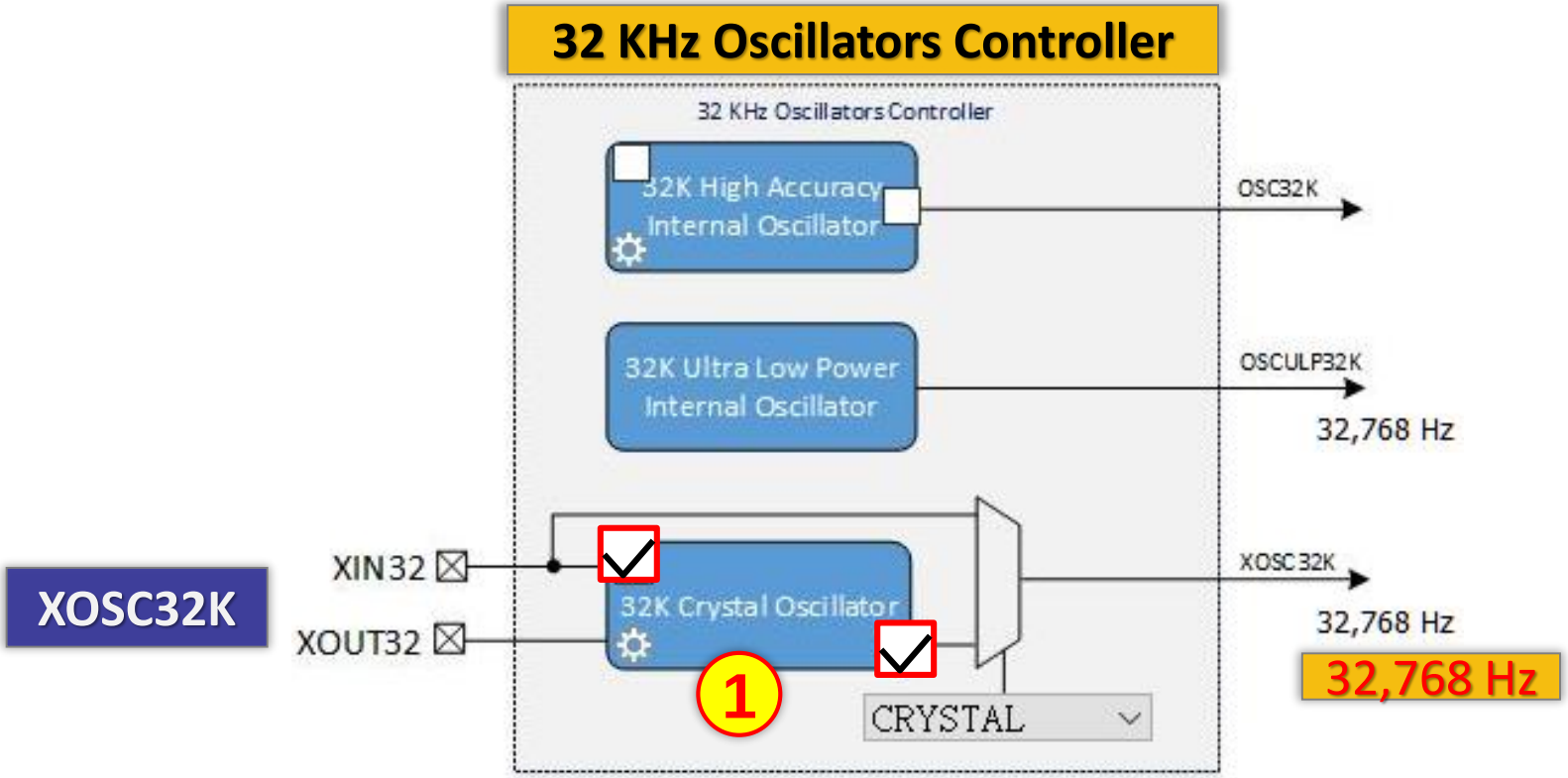
Lab8 System Clock DFLL48M

Setup Steps Overview



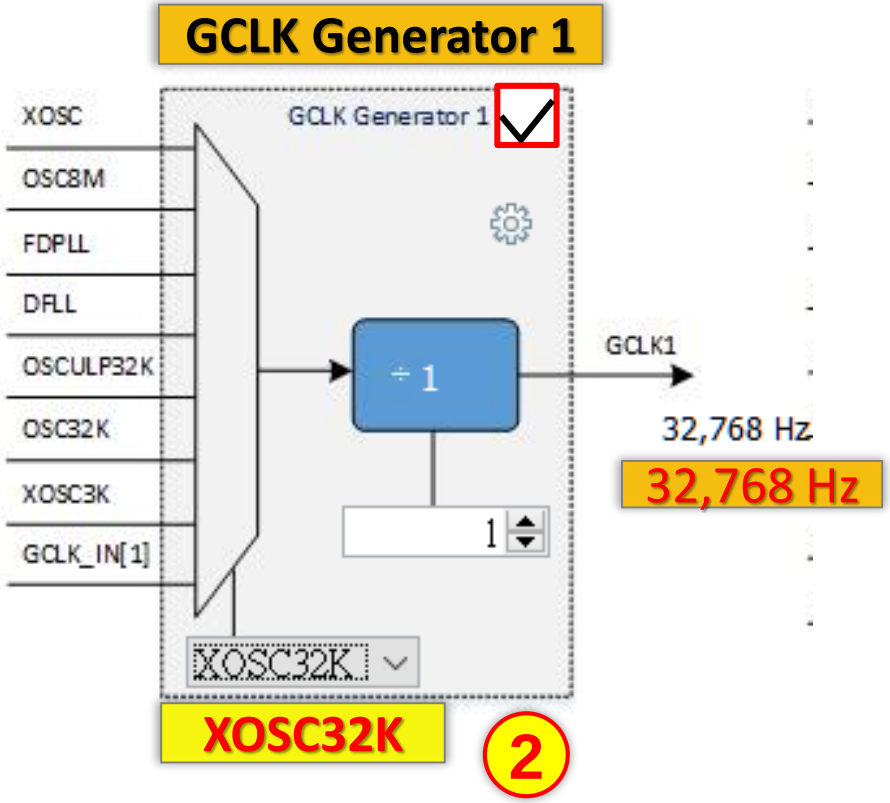
Lab8 System Clock DFLL48M

Step 1



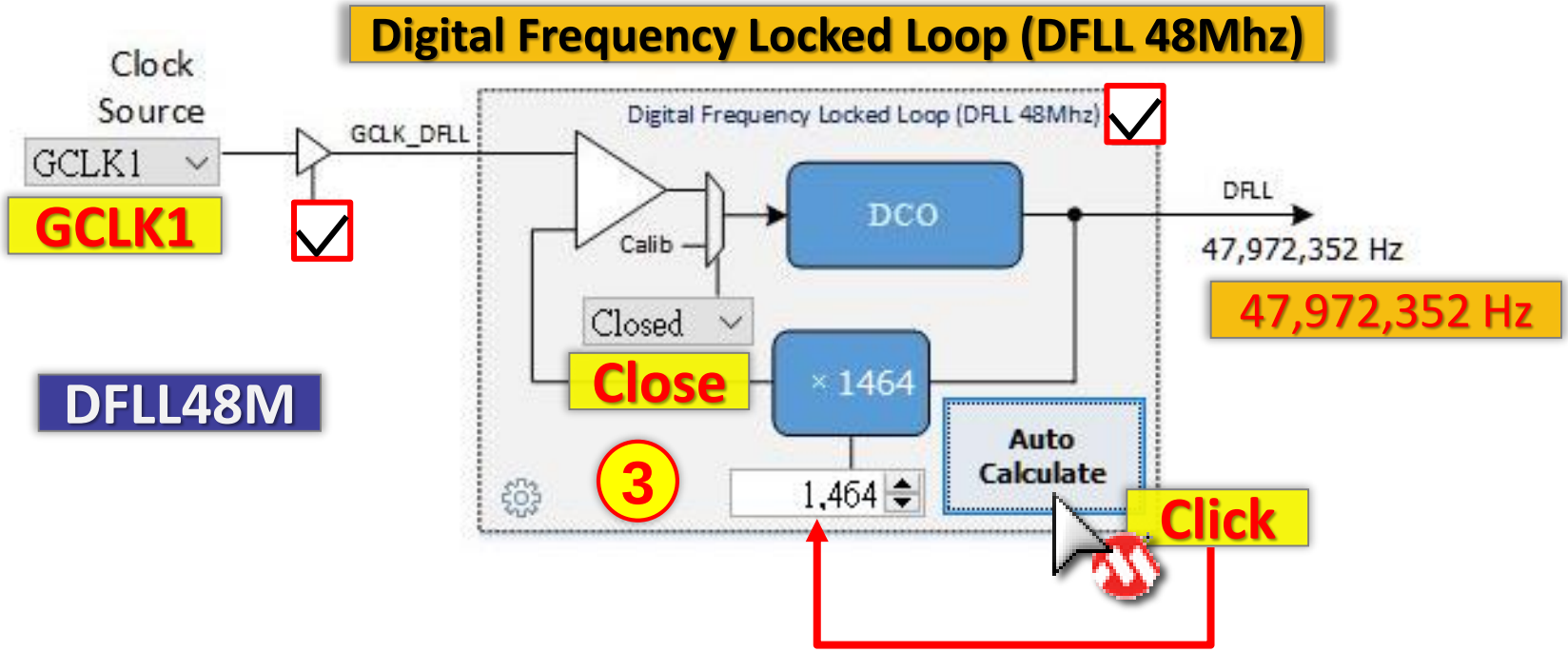
Lab8 System Clock DFLL48M

Step 2



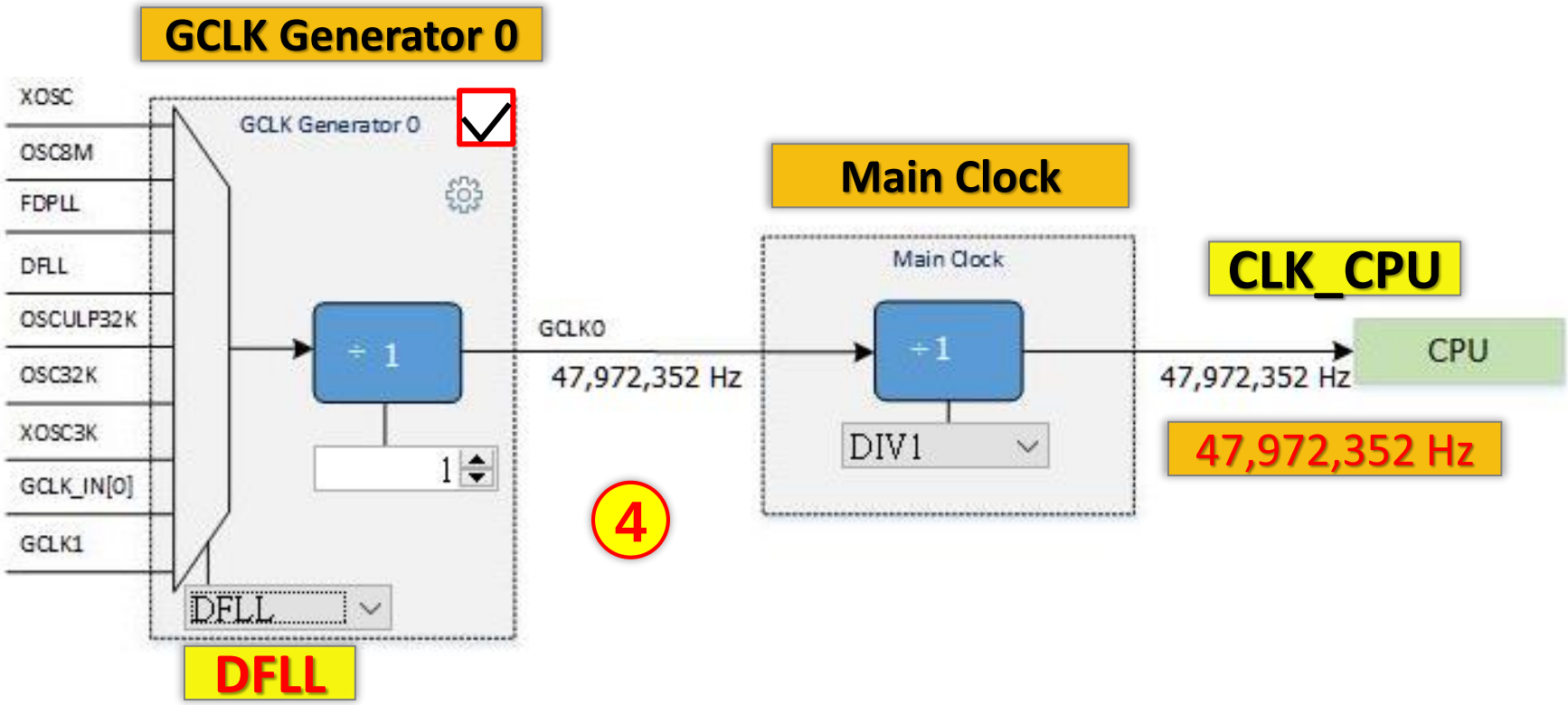
Lab8 System Clock DFLL48M

Step 3



Lab8 System Clock DFL48M

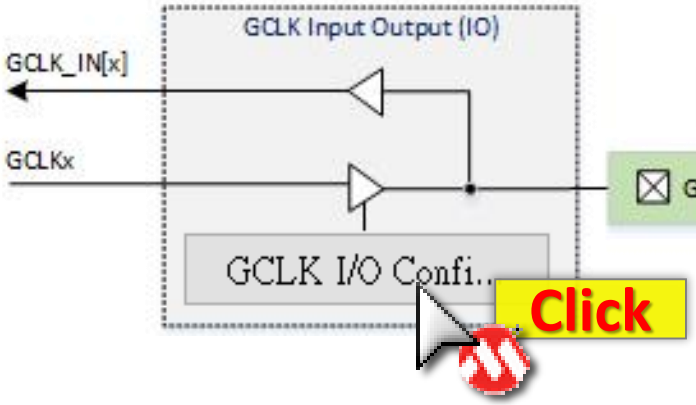
Step 4



Lab8 System Clock DFLL48M

Step 5

GCLK Input Output (IO)



GCLK I/O Configuration

GCLK I/O Configuration

Generator	In/Out	In Frequency(Hz)	
0	Out		47972352
1	Out		32768
2	In	0	0
3	In	0	0
4	In	0	0
5	In	0	0
6	In	0	0
7	In	0	0

Close

0 **Out** **47972352**

5

1 **Out** **32768**

Lab8 System Clock DFLL48M

Wrong Click Sequence in MHC Clock Easy View

plib_clock.c

```
…

void CLOCK_Initialize (void)
{
    /* Function to Initialize the Oscillators */
    SYSCTRL_Initialize();

    GCLK1_Initialize();
    DFLL_Initialize();
    GCLK0_Initialize();

    …
```

Correct Sequence

plib_clock.c

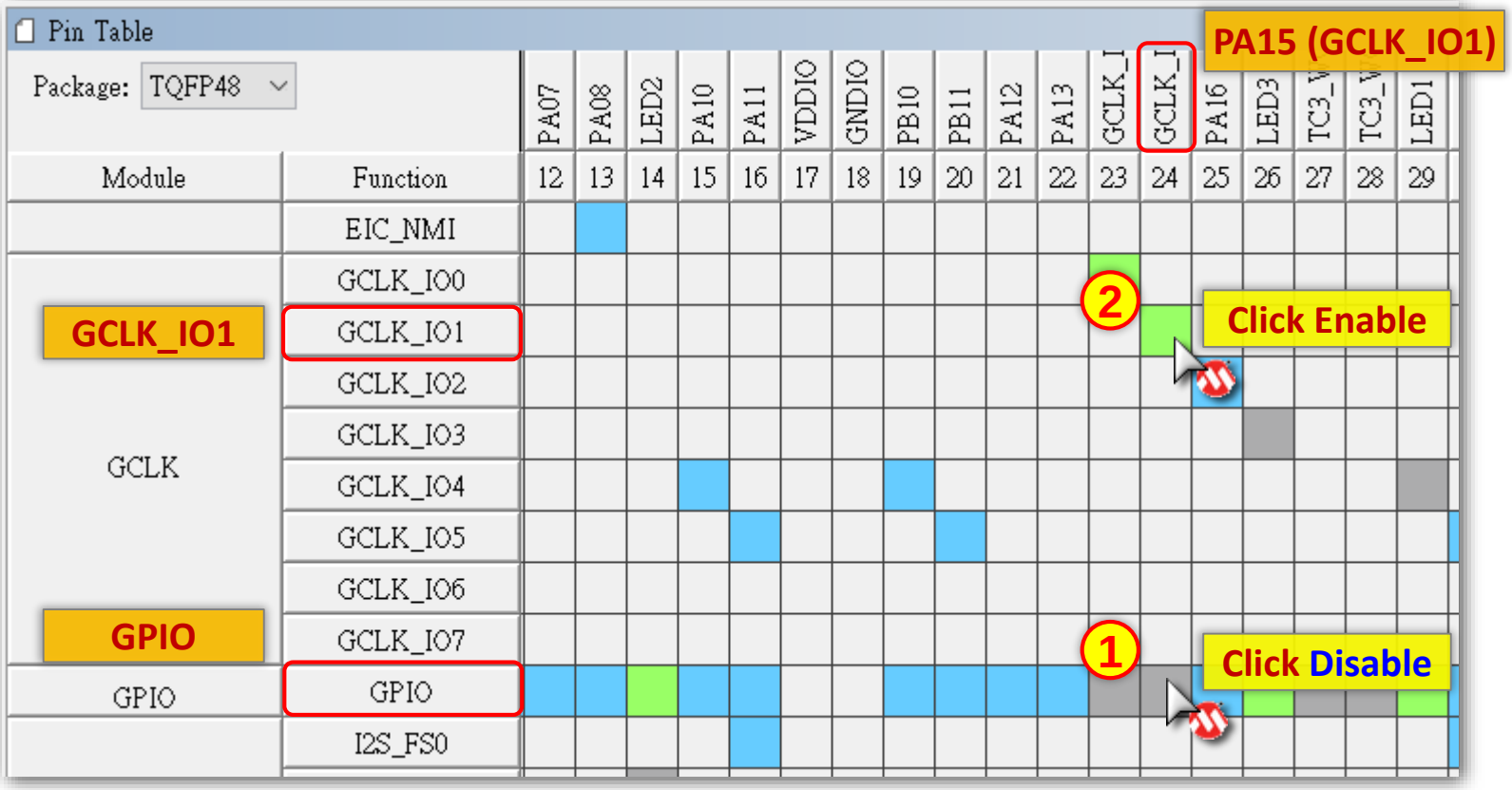
```
…

void CLOCK_Initialize (void)
{
    /* Function to Initialize the Oscillators */
    SYSCTRL_Initialize();

    DFLL_Initialize();
    GCLK1_Initialize();
    GCLK0_Initialize();

    …
```

Wrong Sequence

Lab8 System Clock DFL48M

Step 6

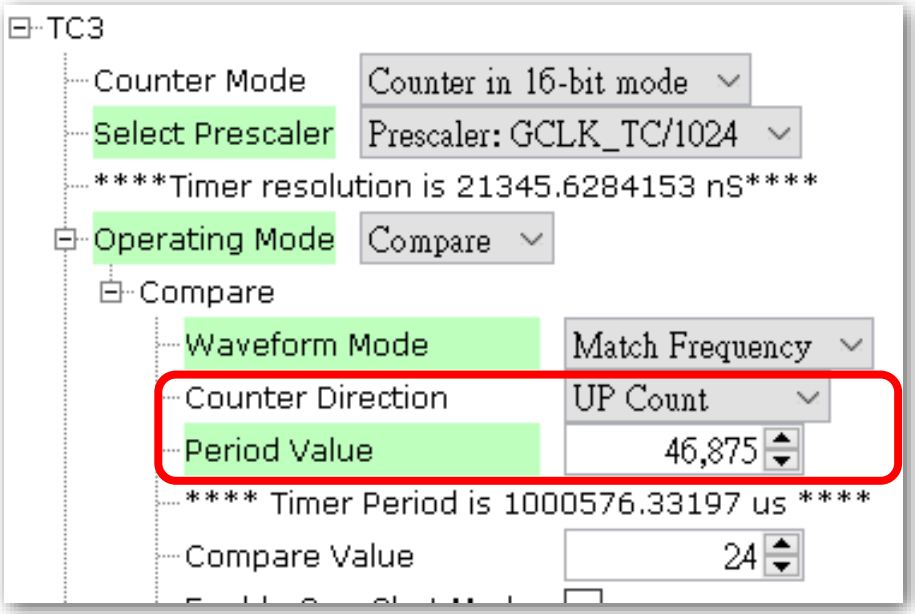
- Please disable BT3(PA15) GPIO pin function firstly and then reassign PA15 as GCLK_IO1 output pin.



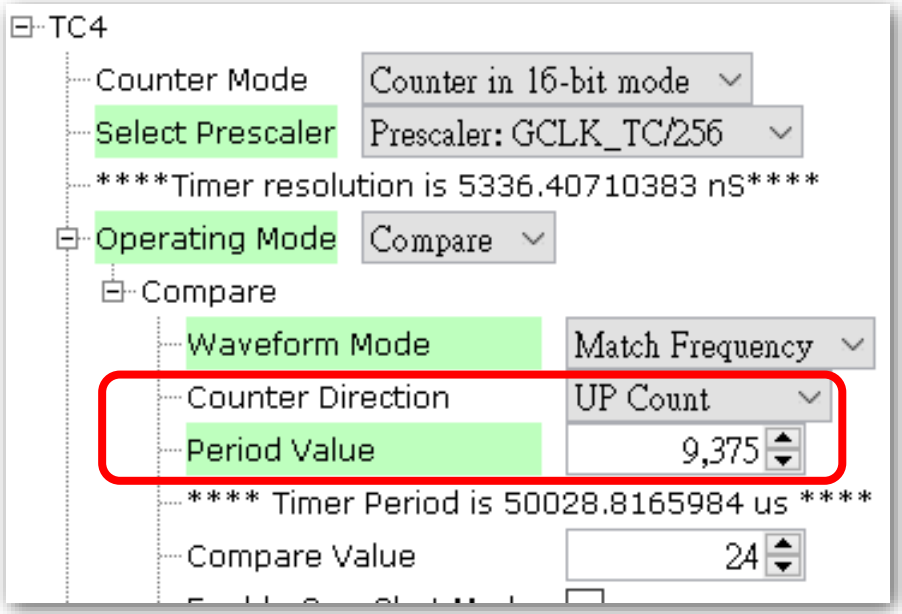
Lab8 System Clock DFL48M

Step 7 LED toggle period correct

TC3 @ 48MHz



TC4 @ 48MHz



Correct Period Value to 46875 to get 1sec period

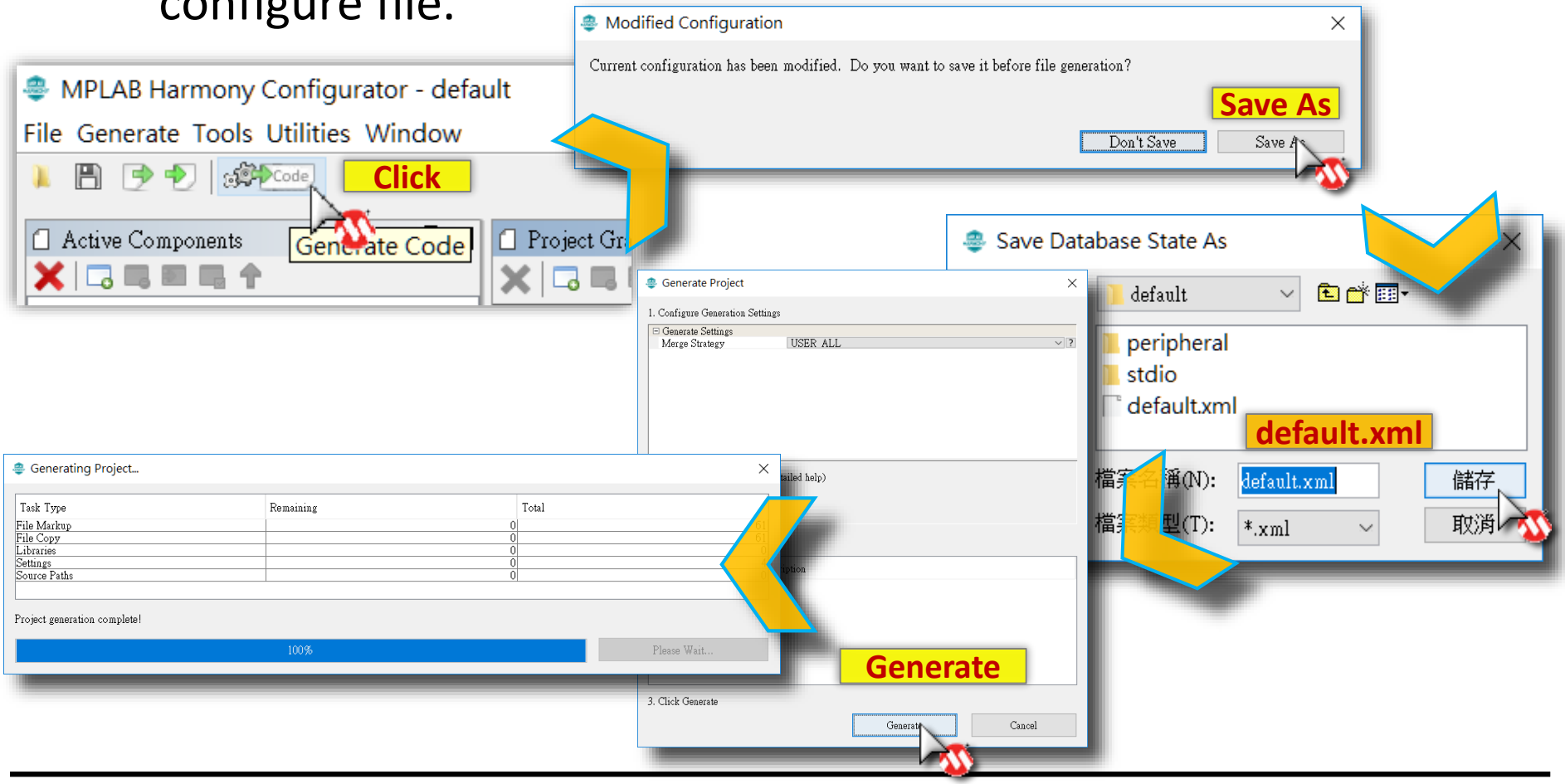
Correct Period Value to 9375 to get 50ms period

Lab8 System Clock DFLL48M

Step 8

- Click 

to Generate Code and save changes to MHC configure file.



Flash Access Time and Wait State

- MHC will auto adjust proper NVM Wait State, also.

plib_nvmctrl.c

```
void NVMCTRL_Initialize(void)
{
    NVMCTRL_REGS->NVMCTRL_CTRLB =
        NVMCTRL_CTRLB_READMODE_NO_MISS_PENALTY |
        NVMCTRL_CTRLB_SLEEPPRM_WAKEONACCESS |
        NVMCTRL_CTRLB_RWS(1) |
        NVMCTRL_CTRLB_MANW_Msk;
}
```

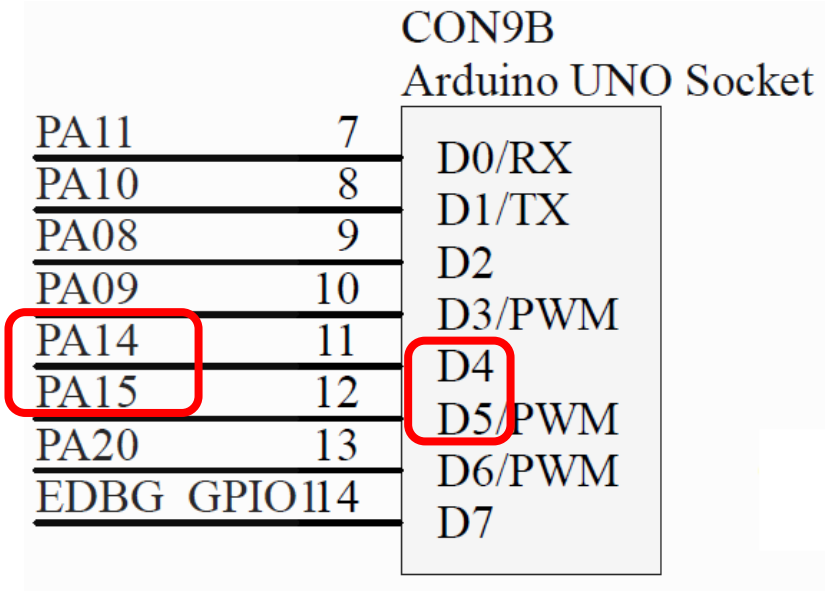
V_{DD} range	NVM Wait States	Maximum Operating Frequency	Units
1.62V to 2.7V	0	14	MHz
	1	28	
	2	42	
	3	48	
2.7V to 3.63V	0	24	
	1	48	

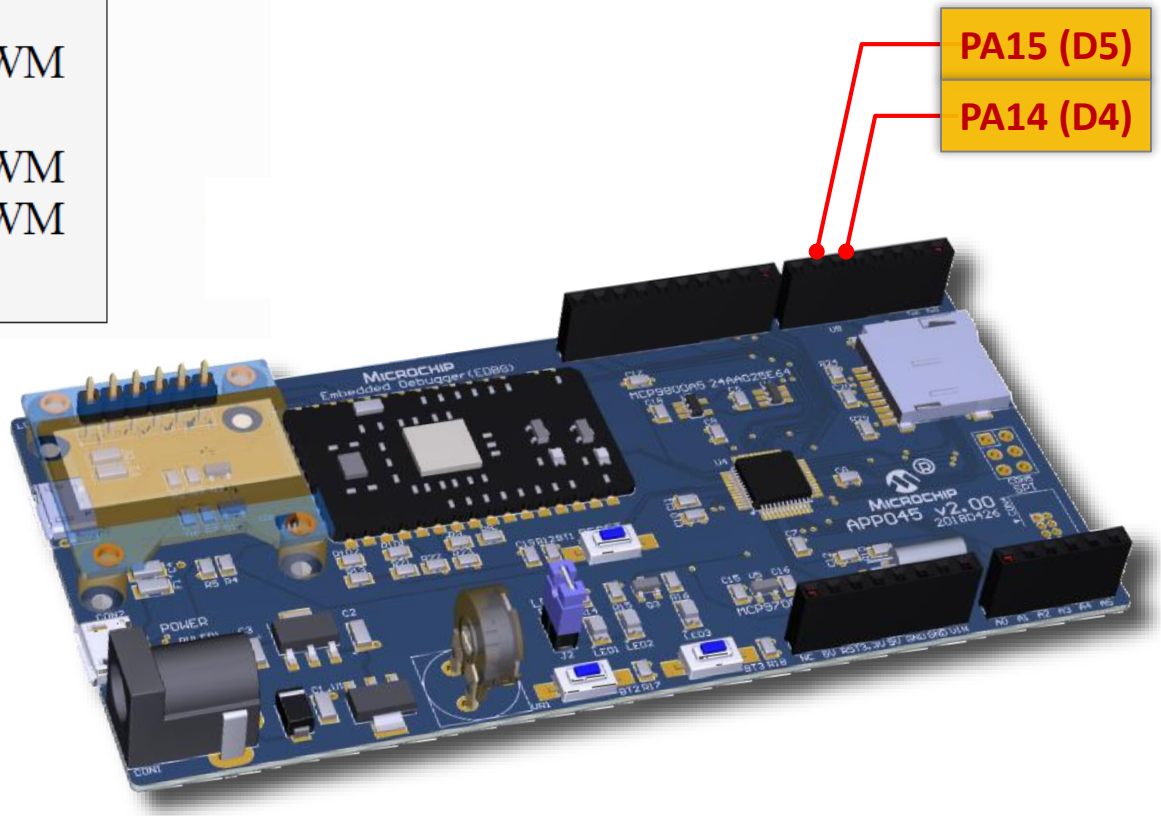
VDD = 3.3V

CPU_CLK = 48MHz

Lab8 System Clock DFL48M

Clock Output





Lab8 System Clock DFLL48M

Clock Output