PLL in LPC2148

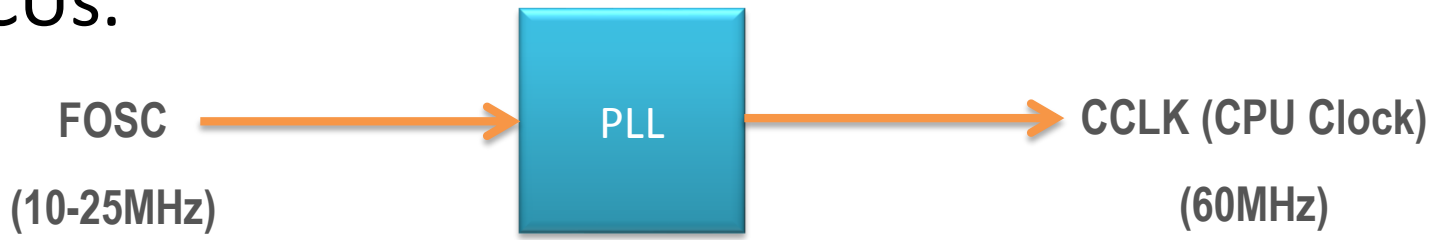
Dr.R.Sundaramurthy

Department of EIE
Pondicherry Engineering College

sundar@pec.edu

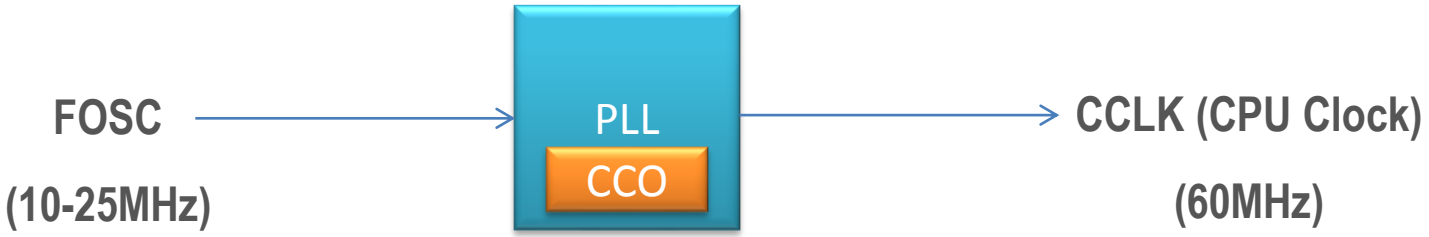
PLL

- There are 2 PLLs in LPC2148 ➔ PLL0, PLL1
- PLL0 ➔ System Clock
- PLL1 ➔ USB Clock
- Input clock to both the PLLs must be between 10MHz to 25MHz strictly.
- This input clock is multiplied with a suitable multiplier and scaled accordingly
- We have a upper limit of 60MHz which is the maximum frequency of operation for lpc214x MCUs.



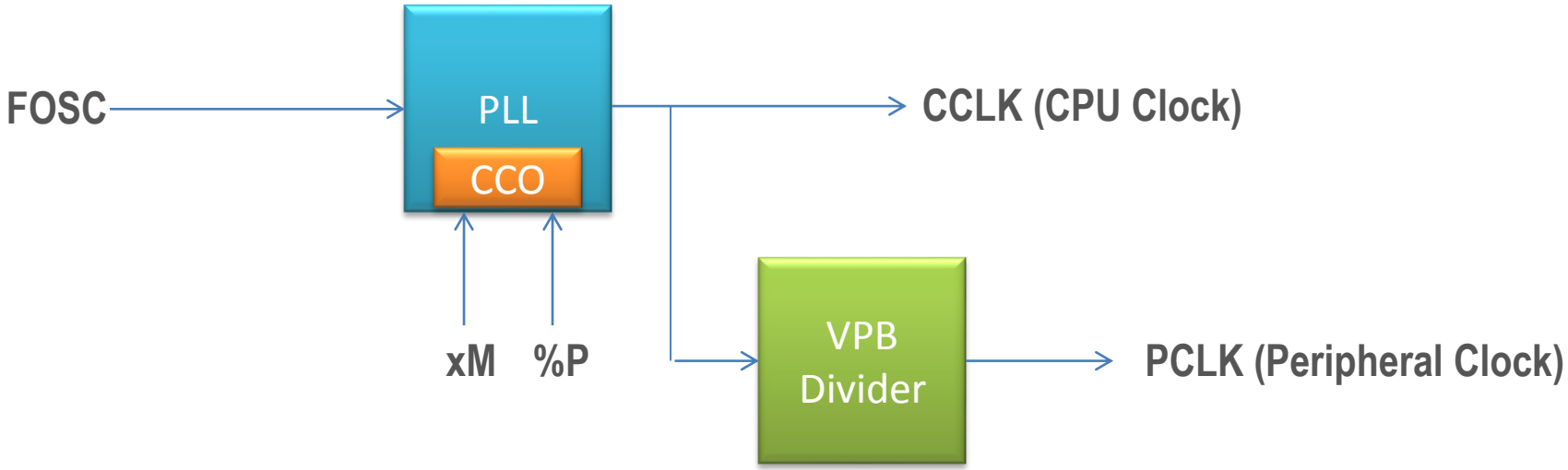
Current Controlled Oscillators (CCO)

- CCOs operate in range of 156Mhz to 320Mhz and there is also a divider to force CCOs to remain in their range.

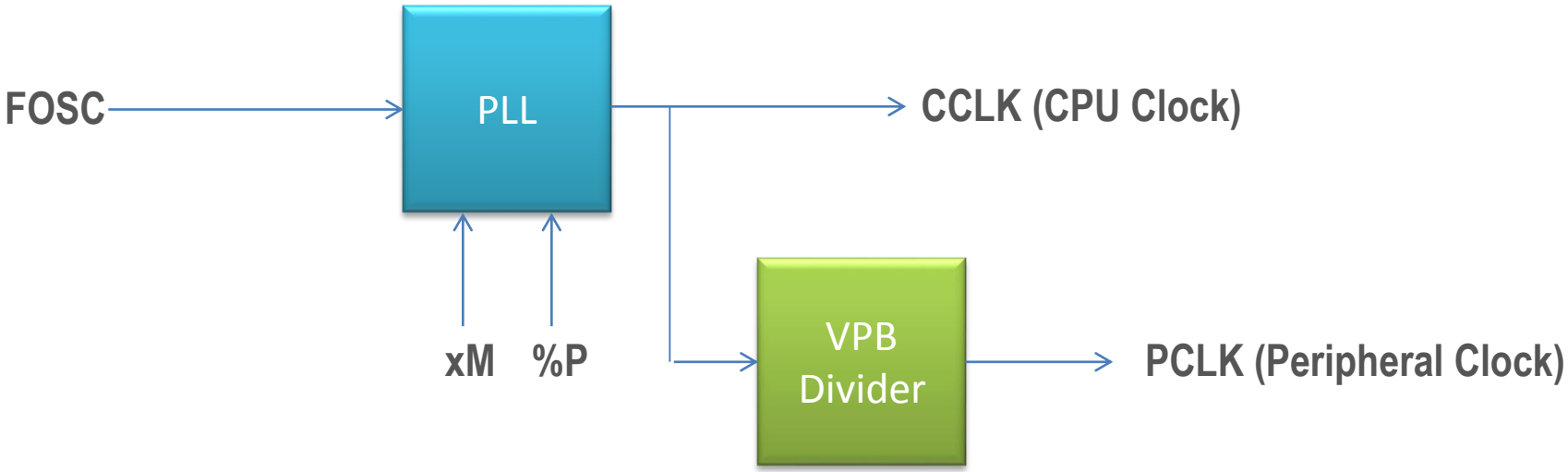


PLL Architecture

FOSC	=> frequency from the crystal oscillator(XTAL)/external clock
FCCO	=> frequency of the PLL Current Controlled Oscillator(CCO)
CCLK	=> PLL output frequency (CPU Clock)
M	=> PLL Multiplier value from the MSEL bits in the PLLCFG register
P	=> PLL Divider value from the PSEL bits in the PLLCFG register
PCLK	=> Peripheral Clock which is derived from CCLK



Equations



CCLK = M x FOSC

CCLK = FCCO / (2 x P)

FCCO = FOSC x M x 2 x P

Frequency Ranges

- FOSC is in the range of **10 MHz to 25 MHz**.
- CCLK is in the range of **10 MHz to 60MHz**
- FCCO is in the range of **156 MHz to 320 MHz.**

SFRS Used

1. **PLL0FEED → Feed Sequence SFR**
2. **PLL0CON → PLL Control Register**
3. **PLL0CFG → PLL Configure Register**
4. **PLL0STAT → PLL Status Register**
5. **VPBDIV → VLSI Peripheral Bus Divider**

Setting Up PLL

How to Setup PLL

1. Setup PLL
2. Apply Feed Sequence
3. Wait for PLL to lock and then connect PLL
4. Apply Feed Sequence.

1. PLL Feed Sequence

- **Assignment of two particular fixed values to PLLFEED register to configure PLL.**
- **PLL0FEED = 0xAA;**
- **PLL0FEED = 0x55;**
- **Something similar to a Key to open a Safe**

2. PLL Control Register (PLL0CON)

- Used to Enable and connect the PLL.
- **First bit** is Enable the PLL
- **Second bit** is Connect the PLL from internal RC oscillator to CPU.

PLL0CON

Bit	Symbol	Description	Reset value
0	PLLE	PLL Enable. When one, and after a valid PLL feed, this bit will activate the PLL and allow it to lock to the requested frequency.	0
1	PLLC	PLL Connect. When PLLC and PLLE are both set to one, and after a valid PLL feed, connects the PLL as the clock source for the microcontroller. Otherwise, the oscillator clock is used directly by the microcontroller.	0
7:2	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

PLL0CON = 0x01 ➔ PLL Enable

PLL0CON = 0x03 ➔ PLL Enable & Connect

3.PLL Configure Register (PLL0CFG)

- Multiplier and Divider are stored here.
- First 5 bits are **MSEL (Multiplier)**
- Bit 5,6 are called **PSEL (Divider)**

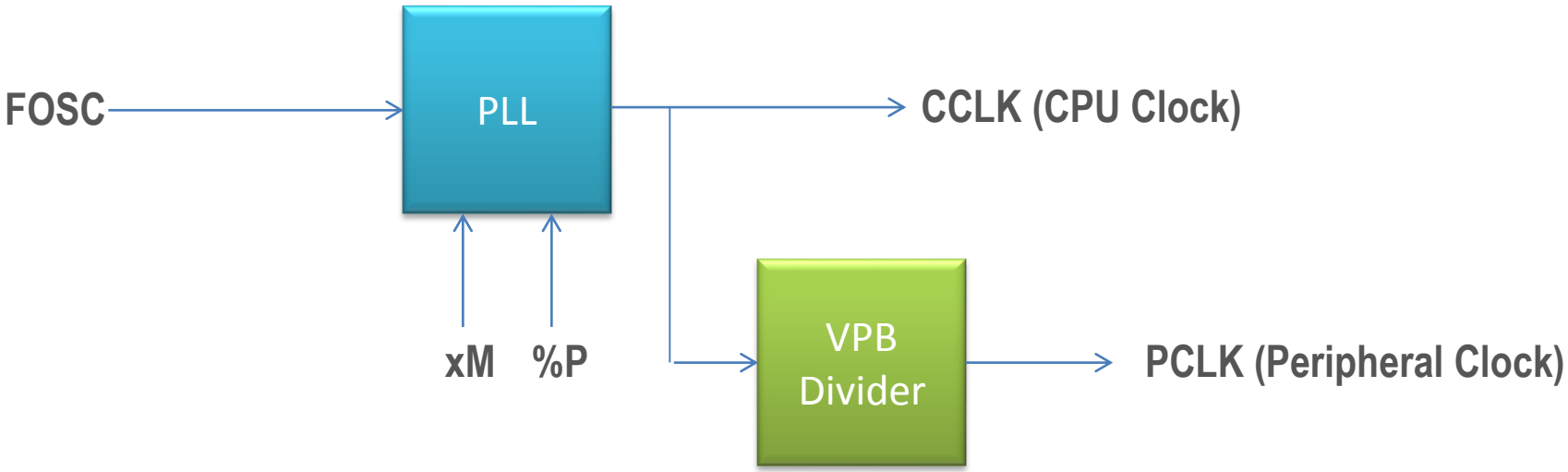
PLL0CFG

Bit	Symbol	Description	Reset value
4:0	MSEL	PLL Multiplier value. Supplies the value "M" in the PLL frequency calculations.	0
6:5	PSEL	PLL Divider value. Supplies the value "P" in the PLL frequency calculations.	0
7	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA

PLL0CFG = 0x24

00100100

MSEL



- **M = CCLK / FOSC**
- M = 60MHz / 12MHz
- M = 5
- Load 5-1 = 4
- The value written to the MSEL bits in PLLCFG is (M – 1)

PSEL

Values of PSEL for P are :

P	Value(binary) in PSEL Bit(5,6)
1	00
2	01
4	10
8	11

P = FCCO/(2xCCLK).

We want FCCO in range 156Mhz to 320Mhz

For FCCO = 156 and we get P = 1.3

For FCCO i.e 320Mhz we get P = 2.67

Now , P must be an integer between 1.3 and 2.67.

So we will use P=2.

PLL Status Register (PLL0STAT)

- Read Only Register
- 10th Bit of PLL0STAT is called **LOCK Bit**
- If LOCK Bit = 1, this means PLL has now latched to the Target Frequency.

PLL0STAT

Bit	Symbol	Description	Reset value
4:0	MSEL	Read-back for the PLL Multiplier value. This is the value currently used by the PLL.	0
6:5	PSEL	Read-back for the PLL Divider value. This is the value currently used by the PLL.	0
7	-	Reserved, user software should not write ones to reserved bits. The value read from a reserved bit is not defined.	NA
8	PLLE	Read-back for the PLL Enable bit. When one, the PLL is currently activated. When zero, the PLL is turned off. This bit is automatically cleared when Power-down mode is activated.	0
9	PLLC	Read-back for the PLL Connect bit. When PLLC and PLLE are both one, the PLL is connected as the clock source for the microcontroller. When either PLLC or PLLE is zero, the PLL is bypassed and the oscillator clock is used directly by the microcontroller. This bit is automatically cleared when Power-down mode is activated.	0
10	PLOCK	Reflects the PLL Lock status. When zero, the PLL is not locked. When one, the PLL is locked onto the requested frequency.	0
15:11	-	Reserved, user software should not write ones to reserved bits.	NA

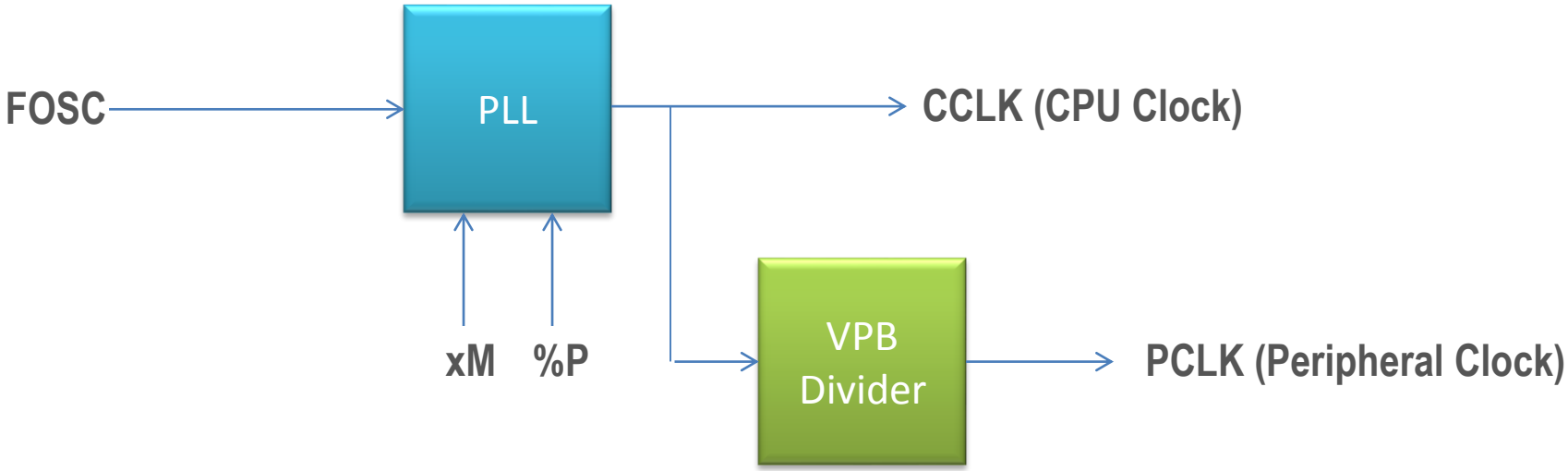
```
while(  PLL0STAT & 0x400 ==0 );
```

100 0000 0000

Void InitPLL()

```
Void initPLL()
{
PLL0CON = 0x01 ;     // Enable PLL
PLL0CFG = 0x24 ;     // Set Up PLL for CClk = 60MHz
PLL0FEED = 0xAA; PLL0FEED = 0x55; // Apply Feed Sequence
while(  PLL0STAT & 0x400 ==0 );  // Check Whether CCO Latched
PLL0CON = 0x03;    // Enable & Connect PLL
PLL0FEED = 0xAA; PLL0FEED = 0x55; // Apply Feed Sequence
}
```

Deriving Pclk from Cclk



VPBDIV=**0x00;**	(PCLK) is one fourth of the processor clock (CCLK)
VPBDIV=**0x01;**	(PCLK) is the same as the processor clock (CCLK)
VPBDIV=**0x02;**	(PCLK) is one half of the processor clock (CCLK)
VPBDIV=**0x03;**	Reserved. If this value is written to the APBDIV register, it has no effect (the previous setting is retained).

Void InitPLL()

```
Void initPLL()
{
PLL0CON = 0x01 ;     // Enable PLL
PLL0CFG = 0x24 ;     // Set Up PLL for CClk = 60MHz
PLL0FEED = 0xAA; PLL0FEED = 0x55; // Apply Feed Sequence
while(  PLL0STAT & 0x400 ==0 );  // Check Whether CCO Latched
PLL0CON = 0x03;    // Enable & Connect PLL
PLL0FEED = 0xAA; PLL0FEED = 0x55; // Apply Feed Sequence
VPBDIV=0x01;  // Pclk is same as Cclk (60 MHz)
}
```

End of Session

sundar@pec.edu