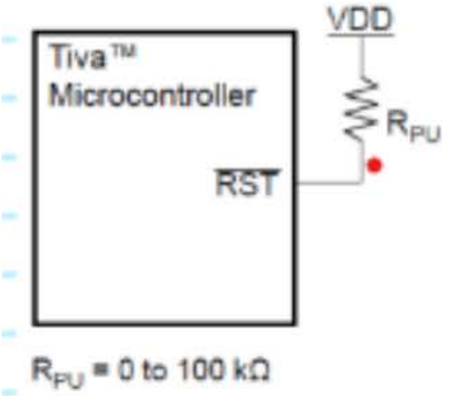
NVIC : NESTED VECTOR INTERRUPTS CONTROLLER

MUHAMMAD ELZEINY

RESET CONTROL

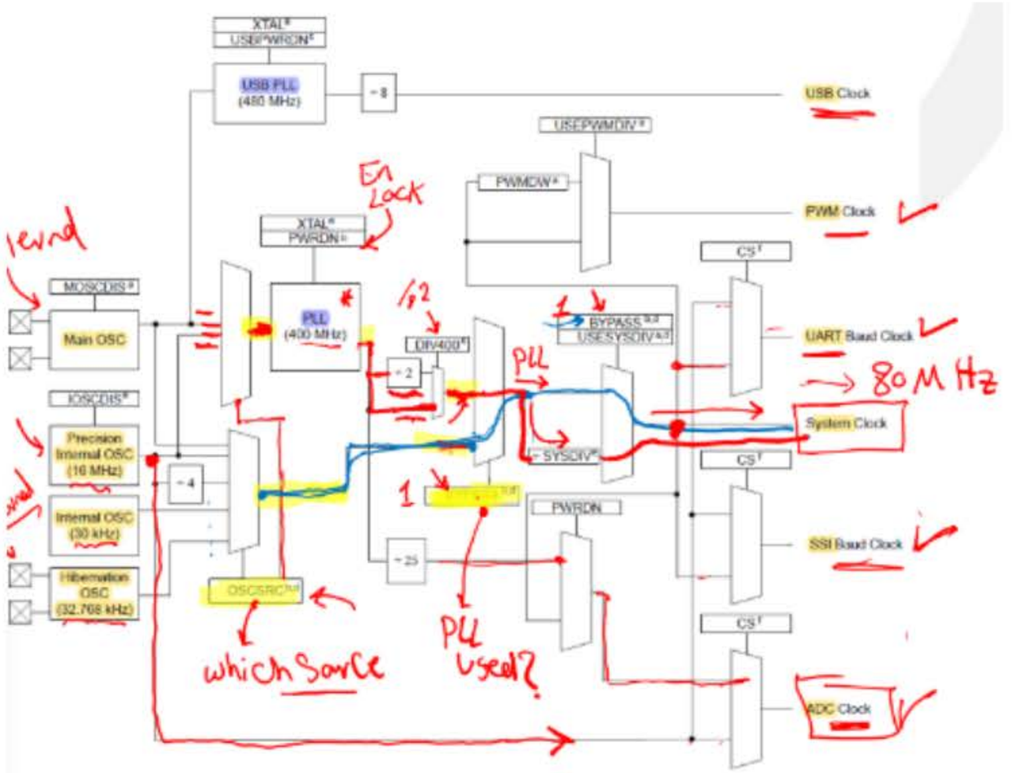
- Reset Sources
  - · Power-on reset (POR)
  - · External reset input pin (RST) assertion
  - · A brown-out detection
  - · Software-initiated reset (with the software reset registers)
  - · A watchdog timer reset condition violation
  - · MOSC(Main Oscillator) failure



- Q: Reset Exception Priority?

CLOCK CONTROL – CLOCK SOURCES

- There are multiple clock sources for use in the microcontroller:
  - Precision Internal Oscillator (PIOSC).
  - Main Oscillator (MOSC).
  - Low-Frequency Internal Oscillator (LFIOSC).
  - Hibernation Module Clock Source.

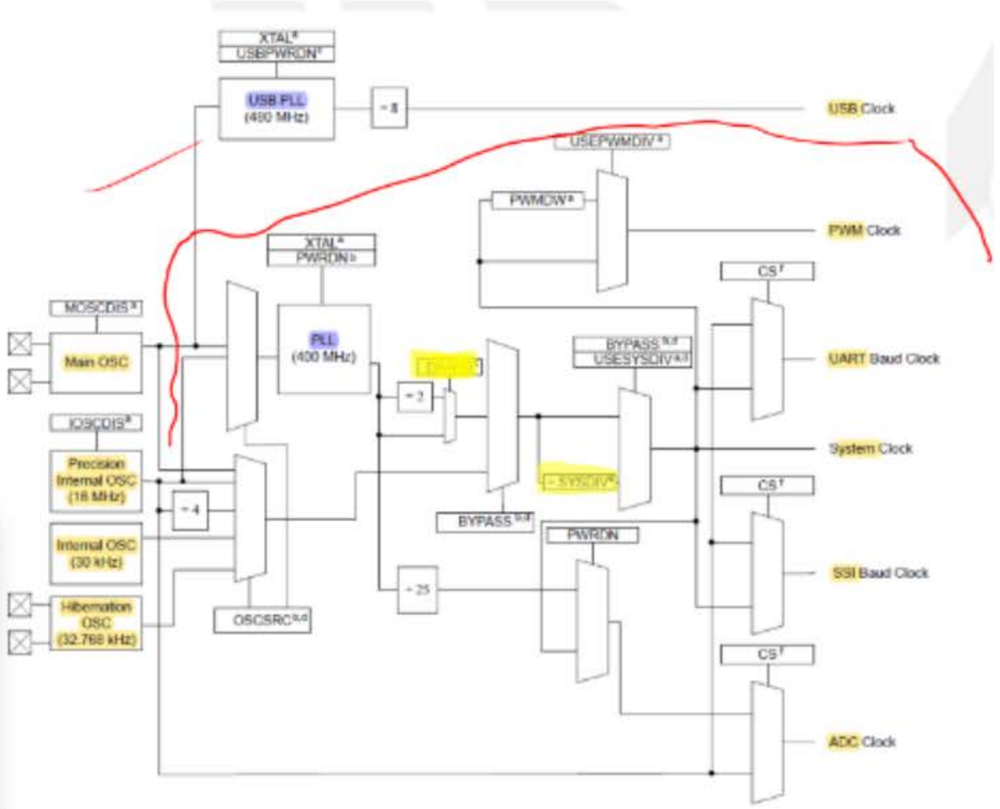


Clock Source	Drive PLL?		Used as SysClk?	
Precision Internal Oscillator	Yes	BYPASS = 0, OSCSRC = 0x1	Yes	BYPASS = 1, OSCSRC = 0x1
Precision Internal Oscillator divide by 4 (4 MHz ± 1%)	No	-	Yes	BYPASS = 1, OSCSRC = 0x2
Main Oscillator	Yes	BYPASS = 0, OSCSRC = 0x0	Yes	BYPASS = 1, OSCSRC = 0x0
Low-Frequency Internal Oscillator (LFIOSC)	No	-	Yes	BYPASS = 1, OSCSRC = 0x3
Hibernation Module 32.768-kHz Oscillator	No	-	Yes	BYPASS = 1, OSCSRC2 = 0x7

CLOCK CONTROL – RUN MODE CONFIGURATION

- The following parameter can be controlled by RCC Registers:
  - Source of clock
  - Pll Use
  - Clock divisor

Clock Source	Drive PLL?		Used as SysClk?	
Precision Internal Oscillator	Yes	BYPASS = 0, OSCSRC = 0x1	Yes	BYPASS = 1, OSCSRC = 0x1
Precision Internal Oscillator divide by 4 (4 MHz ± 1%)	No	-	Yes	BYPASS = 1, OSCSRC = 0x2
Main Oscillator	Yes	BYPASS = 0, OSCSRC = 0x0	Yes	BYPASS = 1, OSCSRC = 0x0
Low-Frequency Internal Oscillator (LFIOSC)	No	-	Yes	BYPASS = 1, OSCSRC = 0x3
Hibernation Module 32.768-kHz Oscillator	No	-	Yes	BYPASS = 1, OSCSRC2 = 0x7



CLOCK CONTROL – PLL MODES

1. Mode Normal: The PLL multiplies the input clock reference and drives the output.

2. Mode Power-Down: Most of the PLL internal circuitry is disabled and the PLL does not drive the output.

CLOCK CONTROL – PLL OPERATIONS

- If a PLL configuration is changed, the PLL output frequency is unstable until it relocks to the new setting.
- If the main PLL is enabled and the system clock is switched to use the PLL → the system control hardware continues to clock the microcontroller from the oscillator selected by the **RCC/RCC2** register until the main PLL is stable, after which it changes to the PLL.

POINTS TO DEFINE

- How to Trigger Soft-Reset – APINT (SYSRESREQUEST \ VECTRESREQ)
- How to Get Reset cause - RESC
- ----------
- How to select Clock source RCC\RCC2
- How to assign certain clock frequency to System Clock (Division) RCC\RCC2
- PLL operation :
    - Enable RCC-RCC2 PWRDN
    - WaitforLock-GetStatus PLLSTAT
    - Distribute after locking BYPASS
- How To En\Disable Clock gates for all Peripherals RCGCx

#TASK – MCU DRIVER

API: Types

- Mcu_PllStatusType
- Mcu_ClockType
- Mcu_RawResetType
- Mcu_ConfigType

API: Functions

- void Mcu_Init(const Mcu_ConfigType* ConfigPtr)
- Mcu_RawResetType Mcu_GetResetRawValue(void)
- void Mcu_PerformReset(void)
- Std_ReturnType Mcu_InitClock(Mcu_ClockType ClockSetting)
- Std_ReturnType Mcu_DistributePllClock(void)
- Mcu_PllStatusType Mcu_GetPllStatus(void)

Configuration

- **Reset Configuration**
  - SW Reset(EN\Disable)
- **Definition of Clock settings**
  - Peripheral clock Gates
  - ClockType used