System Clock

What is the system clock?

- The source clock that all other clocks are derived from
- Many different sources for system clock:
  - Internal 2MHz Osc.
  - Internal 32MHz Osc.
  - Internal 32.768kHz Osc.
  - Internal PLL.
  - External Osc./Clock
- Can be selected at run time via software
- Can be multiplied/divided via software at run time

Selecting a system clock

- By default, 2MHz internal clock is enabled and selected as source
- Easy to change:
  - Use OSC.CTRL (8331 Manual Section 7.10.1) to enable a source.
  - Use OSC.STATUS (8331 Manual Section 7.10.2) to wait for the source to stabilize.
  - Use CLK.CTRL (8331 Manual Section 7.9.1) to switch to new clock source.
- And that's it! Sort of…
- Turns out CLK.CTRL is protected by the configuration change protection mechanism… so...

Configuration Change Protection

- Protects system critical I/O register settings from accidental modification.
- Sequence for writing protected I/O registers
  - Enable change of protected I/O registers using CCP register.
  - Within four clock cycles, write to the protected register.
    - Sometimes, the protected register also was a write enable/change bit that MUST be written at the same time as data.
    - The protected change is immediately disabled if the CPU performs a write operation to the I/O registers, data memory, or if SPM, LMP or SLEEP instruction is executed.
- The narrow write window means use assembly, mixed C, or the GCC compiler optimizations to insure execution is less than four cycles.

Selecting a system clock

- By default, 2MHz internal clock is enabled and selected as source
- Easy to change:
  - Use OSC.CTRL (8331 Manual Section 7.10.1) to enable a source.
  - Use OSC.STATUS (8331 Manual Section 7.10.2) to wait for the source to stabilize.
  - Enable change to protected I/O registers using CCP (8331 Manual Section 3.14.1)
  - Use CLK.CTRL (8331 Manual Section 7.9.1) to switch to new clock source.
- That's it, but it's a little boring.
- But wait, there's more!

Digital Frequency Locked Loops (DFLLs)

- Used to improve the accuracy of the 2MHz and 32MHz internal oscillators.
- Compares oscillator frequency with a more accurate reference to do automatic run-time calibration.
- Choices for references are:
  - 32.678KHz calibrated internal Osc.
  - 32.768KHz crystal Osc. Connected to the TOSC pins
  - External clock
  - USB start of frame
- Reference clock is then divided by 32 to use as 1.024kHz reference

Using DFLLs

- Enable the reference osc.
  - µPad has a 32.768kHz external osc.
  - Use OSC.XOSCCTRL (8331 Manual Section 7.10.3) to select it and followed by OSC.CTRL (8331 Manual Section 7.10.1) to enable it for use.
  - Again, use OSC.STATUS to poll until stable.
- Select DFLL reference with OSC.DFLLCTRL (8331 Manual Section 7.10.7)
- Enable DFLL using DFLLx.CTRL (8331 Manual Section 7.11.1)
- That's it. Calibration data is automatically loaded on reset.

Phase Locked Loops(PLLs)

- Used to generate a high-frequency system clock.
- User-selectable multiplication factor of 1 to 31 (integer values).
  - $f_{out} = f_{in} \cdot PLL\_FAC$
- Choices for clock sources are:
  - Internal 2MHz Osc.
  - Internal 32MHz Osc.
  - 0.4MHz-16MHz Crystal Osc.
  - External Clock
- PLL configuration cannot be changed when in use.
  - Must be disabled before new configuration can be written.
- PLL cannot be used BEFORE clock source is stable.

Using PLLs

- To enable the PLL:
  - Enable the reference clock source
    - Wait until reference source is stable
  - Set the multiplication factor and select the clock reference using OSC.PLLCTRL (8331 Manual Section 7.10.6)
  - Enable the PLL using OSC.CTRL
- Simple… right?

Phase Locked Loops(PLLs)

- Used to generate a high-frequency system clock.
- User-selectable multiplication factor of 1 to 31 (integer values).
  - $f_{out} = f_{in} \cdot PLL\_FAC$
- Choices for clock sources are:
  - Internal 2MHz Osc.
  - Internal 32MHz Osc.
    - Gets divided by four before it reaches PLL, actually 8MHz
  - 0.4MHz-16MHz Crystal Osc.
  - External Clock
- PLL configuration cannot be changed when in use.
  - Must be disabled before new configuration can be written.
- PLL cannot be used BEFORE clock source is stable.

Clock Prescalers

- Used to divide the clock signal by a factor of 1 to 2048 before it is routed to the CPU and peripherals.
- Broken up into three precalers:
  - Prescaler A
    - Divide CLK_{SYS} by factor of 1 to 512 to generate CLK_{PER4}
  - Prescaler B
    - Divide CLK_{PER4} by factor of 1, 2, or 4 to generate CLK_{PER2}
  - Prescaler C
    - Divide CLK_{PER2} by factor of 1 or 2 to generate CLK_{CPU} and CLK_{PER}