Clock System

These lecture notes created by Mr. James B. (Jim) Carlson, NCSU

In These Notes . . .

- Clocking structure of the Processor
  - Auxiliary clock (ACLK)
  - Main clock (MCLK)
  - Sub-Main clock (SMCLK)
  - MODOSC

Why It’s Needed

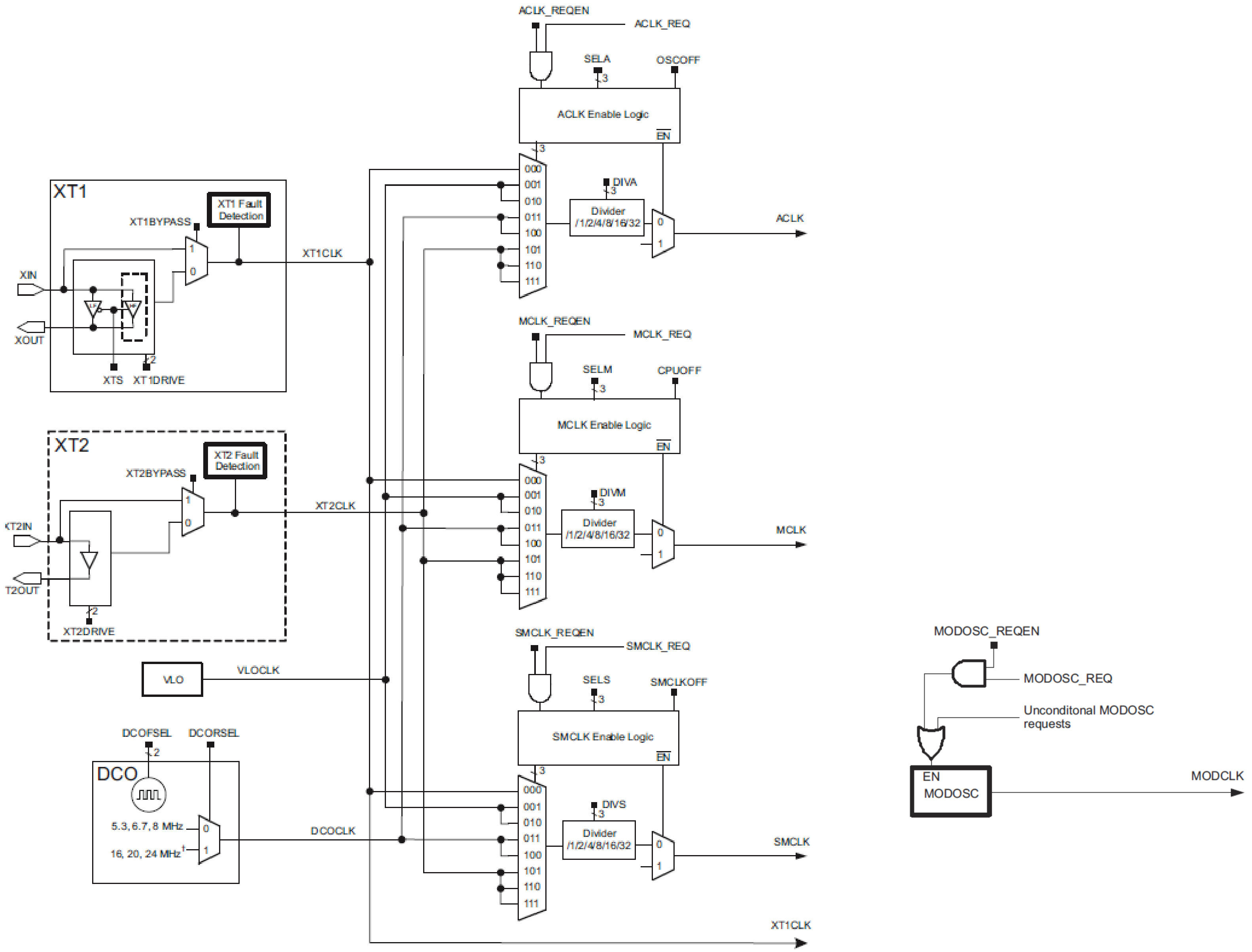
- At the core of all processors is a clock
- Multiple clock options allow for different timing for optimized peripheral operation.
- Faster Clocks use more power, Slow clocks use less power

MSP430 Clock System

- For details see **MSP430FR57xx Family User's Guide**
  - Clock System Chapter 3

- The clock system module supports low system cost and low power consumption. Using three internal clock signals [a forth with some other peripheral], the user can select the best balance of performance and low power consumption. The clock module can be configured to operate without any external components, with one or two external crystals, or with resonators, under full software control.

MSP430 Clock System Block Diagram



MSP430 Clock System

- The clock system module includes up to five clock sources:
  - XT1CLK: Low-frequency or high-frequency oscillator that can be used with low-frequency 32768-Hz watch crystals, standard crystals, resonators, or external clock sources in the 4 MHz to 24 MHz range. When optional XT2 is present , the XT1 high-frequency mode may or may not be available.
  - VLOCLK: Internal very-low-power low-frequency oscillator with 10-kHz typical frequency
  - DCOCLK: Internal digitally controlled oscillator (DCO) with three selectable fixed frequencies
  - XT2CLK: Optional high-frequency oscillator that can be used with standard crystals, resonators, or external clock sources in the 4 MHz to 24 MHz range. See the device-specific data sheet for availability.
  - MODCLK: Module clock. MODCLK is used by various peripheral modules and is sourced by MODOSC.

MSP430 Clock Module

- Four system clock signals are available from the clock module:
  - ACLK: Auxiliary clock. The ACLK is software selectable as XT1CLK, VLOCLK, DCOCLK, and when available, XT2CLK. ACLK can be divided by 1, 2, 4, 8, 16, or 32. ACLK is software selectable by individual peripheral modules.
  - MCLK: Master clock. MCLK is software selectable as XT1CLK, VLOCLK, DCOCLK, and when available, XT2CLK. MCLK can be divided by 1, 2, 4, 8, 16, or 32. MCLK is used by the CPU and system.
  - SMCLK: Subsystem master clock. SMCLK is software selectable as XT1CLK, VLOCLK, DCOCLK, and when available, XT2CLK. SMCLK is software selectable by individual peripheral modules.
  - MODCLK: Module clock. MODCLK is used by various peripheral modules and is sourced by MODOSC.

MSP430 Clock System Operation

- After Power Up Condition, the Clock System module default configuration is:
  - XT1 in low frequency (LF) mode (XTS = 0) is selected as the oscillator source for XT1CLK. XT1CLK is selected for ACLK (SELA = {0}).
  - DCOCLK is selected for MCLK and SMCLK (SELM = SELS = {3}) and each are divided by 8 (DIVM = DIVS = {3}).
  - XIN and XOUT pins are set to general-purpose I/Os and XT1 remains disabled until the I/O ports are configured for XT1 operation.
  - When XT2 is an available option, XT2IN and XT2OUT pins are set to general-purpose I/Os and XT2 is disabled.
- XT1 is selected by default, but XT1 is disabled. The crystal pins (XIN, XOUT) are shared with general-purpose I/Os. To enable XT1, the PSEL bits associated with the crystal pins must be set. When a 32768-Hz crystal is used for XT1CLK, the fault control logic immediately causes ACLK to be sourced by the VLOCLK, because XT1 is not stable immediately.

MSP430 Clock System Operation

- Status register control bits (SCG0, SCG1, OSCOFF, and CPUOFF) configure the device operating modes and enable or disable portions of the clock system module.

- Registers CSCTL0 through CSCTL6 configure the Clock System module.

- The Clock System module can be configured or reconfigured by software at any time during program execution. The Clock System control registers are password protected to prevent inadvertent access.

MSP430 CS Module Features for Low-Power

- Conflicting requirements typically exist in battery-powered applications:
  - Low clock frequency for energy conservation and time keeping.
  - High clock frequency for fast response times and fast burst processing capabilities.
  - Clock stability over operating temperature and supply voltage .
  - Low-cost applications with less-constrained clock accuracy requirements.
- The Clock System module addresses these conflicting requirements by allowing the user to select from the three available clock signals: ACLK, MCLK, and SMCLK.
- All three available clock signals can be sourced from any of the available clock sources (XT1CLK, VLOCLK, DCOCLK, or XT2CLK), giving complete flexibility in the system clock configuration. A flexible clock distribution and divider system is provided to fine-tune the individual clock requirements.

MSP430 Clock Module

- ***Internal Very-Low-Power Low-Frequency Oscillator (VLO)***
- The internal VLO provides a typical frequency of 10 kHz (see the device-specific data sheet for parameters) without requiring a crystal. The VLO provides for a low-cost ultra-low-power clock source for applications that do not require an accurate time base.
- The VLO can be used to source ACLK, MCLK, or SMCLK (SELA = {1} or SELM = {1} or SELS = {1}).

MSP430 XT1 Oscillator

- The XT1 oscillator supports ultra-low-current consumption using a 32768-Hz watch crystal in low frequency (LF) mode (XTS = 0). The watch crystal connects to XIN and XOUT and requires external capacitors on both terminals. These capacitors should be sized according to the crystal or resonator specifications.
- On devices that do not include the optional XT2 oscillator, the XT1 oscillator also supports high-speed crystals or resonators when in high-frequency (HF) mode (XTS = 1). The high-speed crystal or resonator connects to XIN and XOUT and requires external capacitors on both terminals. These capacitors should be sized according to the crystal or resonator specifications.
- In XT1 LF or HF modes, different crystal or resonator ranges are supported by choosing the proper XT1DRIVE settings. XT1 may be used with an external clock signal on the XIN pin in either LF or HF mode by setting XT1BYPASS = 1. When used with an external signal, the external frequency must meet the data sheet parameters for the chosen mode. XT1 is powered down when used in bypass mode.

MSP430 XT1 Oscillator

- The XT1 pins are shared with general-purpose I/O ports. At power up, the default operation is XT1, LF mode of operation. However, XT1 remains disabled until the ports shared with XT1 are configured for XT1 operation. The configuration of the shared I/O is determined by the PSEL bit associated with XIN and the XT1BYPASS bit.
- Setting the PSEL bit causes the XIN and XOUT ports to be configured for XT1 operation. If XT1BYPASS is also set, XT1 is configured for bypass mode of operation, and the oscillator associated with XT1 is powered down. In bypass mode of operation, XIN can accept an external clock input signal and XOUT is configured as a general-purpose I/O. The PSEL bit associated with XOUT is a don't care.
- If the PSEL bit associated with XIN is cleared, both XIN and XOUT ports are configured as general purpose I/Os, and XT1 is disabled.
- XT1 is enabled under any of the following conditions:
  – XT1 is a source for ACLK (SELA = 0) and in active mode (AM) through LPM3 (OSCOFF = 0)
  – XT1 is a source for MCLK (SELM = 0) and in active mode (AM) (CPUOFF = 0)
  – XT1 is a source for SMCLK (SELS = 0) and in active mode (AM) through LPM1 (SMCLKOFF = 0)
  – XT1OFF = 0. XT1 enabled in active mode (AM) through LPM4.

NC STATE UNIVERSITY

MSP430 XT2 Oscillator

- Some devices have a second crystal oscillator, XT2. XT2 sources XT2CLK, and its characteristics are identical to XT1 in HF mode. The XT2DRIVE bits select the frequency range of operation of XT2. Devices that support XT2 may or may not support XT1 in HF mode; see the device-specific data sheet for availability.
- XT2 may be used with external clock signals on the XT2IN pin by setting XT2BYPASS = 1. When used with an external signal, the external frequency must meet the data-sheet parameters for XT2. XT2 is powered down when used in bypass mode.
- The XT2 pins are shared with general-purpose I/O ports. At power up, the default operation is XT2. However, XT2 remains disabled until the ports shared with XT2 are configured for XT2 operation. The configuration of the shared I/O is determined by the PSEL bit associated with XT2IN and the XT2BYPASS bit. Setting the PSEL bit causes the XT2IN and XT2OUT ports to be configured for XT2 operation. If XT2BYPASS is also set, XT2 is configured for bypass mode of operation, and the oscillator associated with XT2 is powered down. In bypass mode of operation, XT2IN can accept an external clock input signal and XT2OUT is configured as a general-purpose I/O. The PSEL bit associated with XT2OUT is a don't care.

MSP430 XT2 Oscillator

- If the PSEL bit associated with XT2IN is cleared, both XT2IN and XT2OUT ports are configured as general-purpose I/Os, and XT2 is disabled.

- XT2 is enabled under any of the following conditions:
  - XT2 is a source for ACLK (SELA = {5,6,7}) and in active mode (AM) through LPM3 (OSCOFF = 0)
  - XT2 is a source for MCLK (SELM = {5,6,7}) and in active mode (AM) (CPUOFF = 0)
  - XT2 is a source for SMCLK (SELS = {5,6,7}) and in active mode (AM) through LPM1 (SMCLKOFF = 0)
  - XT2OFF = 0. XT2 enabled in active mode (AM) through LPM4.

MSP430 Digitally Controlled Oscillator (DCO)

- The DCO is an integrated digitally controlled oscillator. The DCO has three frequency settings determined by the DCOFSEL bits. Each frequency is trimmed at the factory. The DCO can be used as a source for ACLK, MCLK, or SMCLK.

- The DCO frequency can be changed at any time, but care should be taken to ensure no other system clock frequency constraints are exceeded with the new frequency selection.

- Any change in the DCOFSEL or DCORSEL bits causes the DCOCLK to be held for four clock cycles before releasing the new value into the system. This allows for the DCO to settle properly.

MSP430 Operation From Low-Power Modes

- A peripheral module requests its clock sources automatically from the Clock System module if required for its proper operation, regardless of the current power mode of operation.
- A peripheral module asserts one of three possible clock request signals based on its control bits: ACLK_REQ, MCLK_REQ, or SMCLK_REQ. These request signals are based on the configuration and clock selection of the respective module.
- If a timer selects ACLK as its clock source and the timer is enabled, the timer generates an ACLK_REQ signal to the Clock System. The Clock System, in turn, enables ACLK regardless of the power mode settings. Any clock request from a peripheral module causes its respective clock off signal to be overridden, but does not change the setting of clock off control bit.
- For example, a peripheral module may require ACLK that is currently disabled by the OSCOFF bit (OSCOFF = 1). The module can request ACLK by generating an ACLK_REQ. This causes the OSCOFF bit to have no effect, thereby allowing ACLK to be available to the requesting peripheral module. The OSCOFF bit remains at its current setting (OSCOFF = 1).

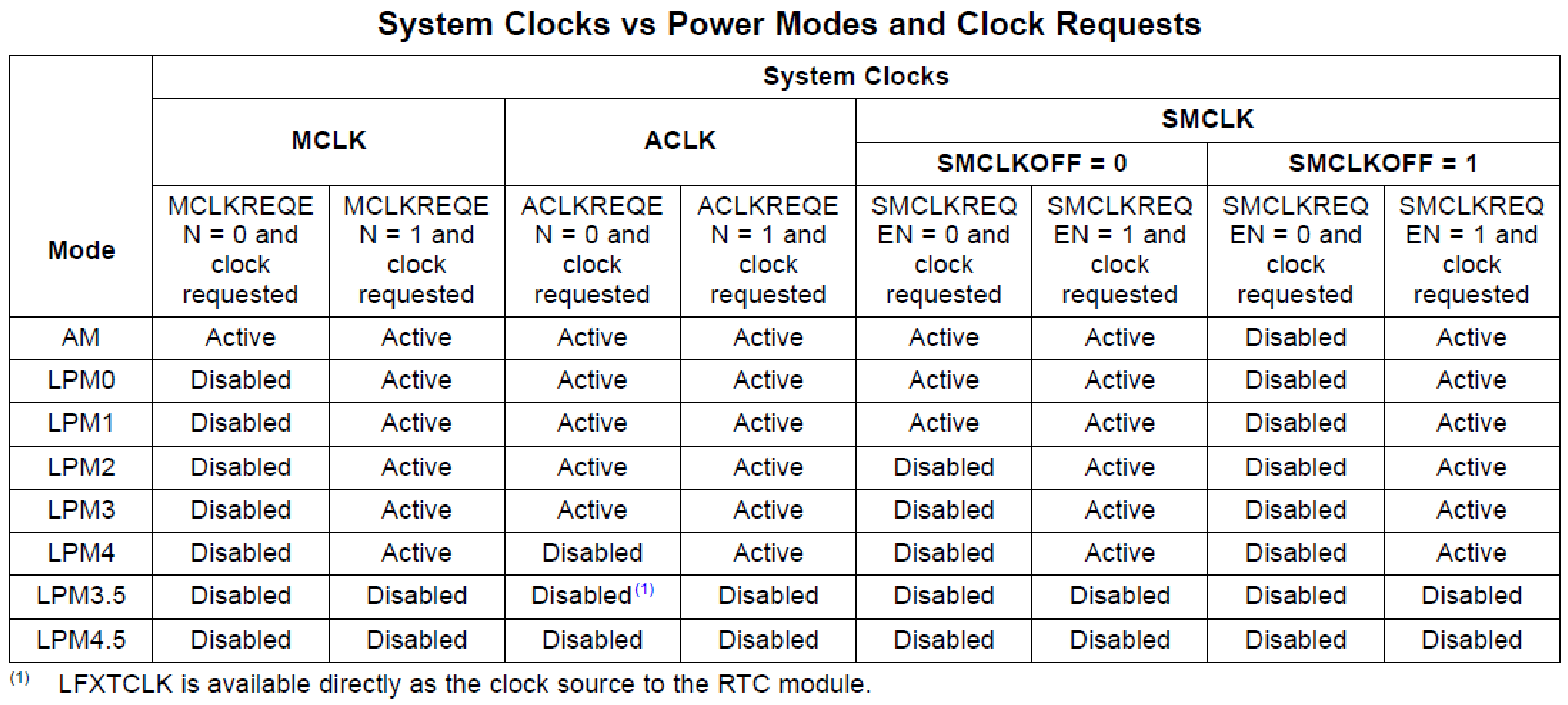
MSP430 Operation From Low-Power Modes

- If the requested source is not active, the software NMI handler must manage the required actions.
- For the previous example, if ACLK was sourced by XT1, and XT1 was not enabled, an oscillator fault condition occurs and the software must handle the event. The watchdog, due to its security requirement, actively selects the VLOCLK source if the originally selected clock source is not available.
- Due to the clock request feature, care must be taken in the application when entering low-power modes to save power. Although the device enters the selected low-power mode, a clock request causes more current consumption than the specified values in the data sheet.
- By default, the clock request feature is enabled. The feature can be disabled for each system clock by clearing ACLKREQEN, MCLKREQEN, or SMCLKREQEN for the respective clocks. This does not disable fail-safe clock requests; for example, those of the watchdog timer or the clock system itself.

MSP430 Operation From Low-Power Modes

- The function of the ACLKREQEN, MCLKREQEN, and SMCLKREQEN bits are dependent upon which power mode is selected; that is, they do not have an effect across all power modes.
- For example, ACLKREQEN is used to enable or disable ACLK requests. It is only effective in LPM4, because in all other modes (AM, LPM0, LPM1, LPM2, LPM3) ACLK is always active.
- SMCLKREQEN is used to enable or disable SMCLK requests. When SMCLKOFF = 0 and in AM, LPM0, or LPM1, it is a don't care because SMCLK is always on in these cases. For SMCLKOFF = 0 and in LPM2, LPM3, and LPM4, SMCLKREQEN can be used to enable or disable SMCLK requests, because in these modes, SMCLK is normally off.
- When SMCLKOFF = 1, SMCLKREQEN can be used to enable or disable SMCLK requests, because under this condition SMCLK is normally off in all power modes

MSP430 Operation From Low-Power Modes



System Clocks vs Power Modes and Clock Requests

Mode	System Clocks							
	MCLK		ACLK		SMCLK			
					SMCLKOFF = 0		SMCLKOFF = 1	
	MCLKREQEN = 0 and clock requested	MCLKREQEN = 1 and clock requested	ACLKREQEN = 0 and clock requested	ACLKREQEN = 1 and clock requested	SMCLKREQEN = 0 and clock requested	SMCLKREQEN = 1 and clock requested	SMCLKREQEN = 0 and clock requested	SMCLKREQEN = 1 and clock requested
AM	Active	Active	Active	Active	Active	Active	Disabled	Active
LPM0	Disabled	Active	Active	Active	Active	Active	Disabled	Active
LPM1	Disabled	Active	Active	Active	Active	Active	Disabled	Active
LPM2	Disabled	Active	Active	Active	Disabled	Active	Disabled	Active
LPM3	Disabled	Active	Active	Active	Disabled	Active	Disabled	Active
LPM4	Disabled	Active	Disabled	Active	Disabled	Active	Disabled	Active
LPM3.5	Disabled	Disabled	Disabled[1]	Disabled	Disabled	Disabled	Disabled	Disabled
LPM4.5	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled	Disabled

[1] LFXTCLK is available directly as the clock source to the RTC module.

MSP430 CS Module Fail-Safe Operation

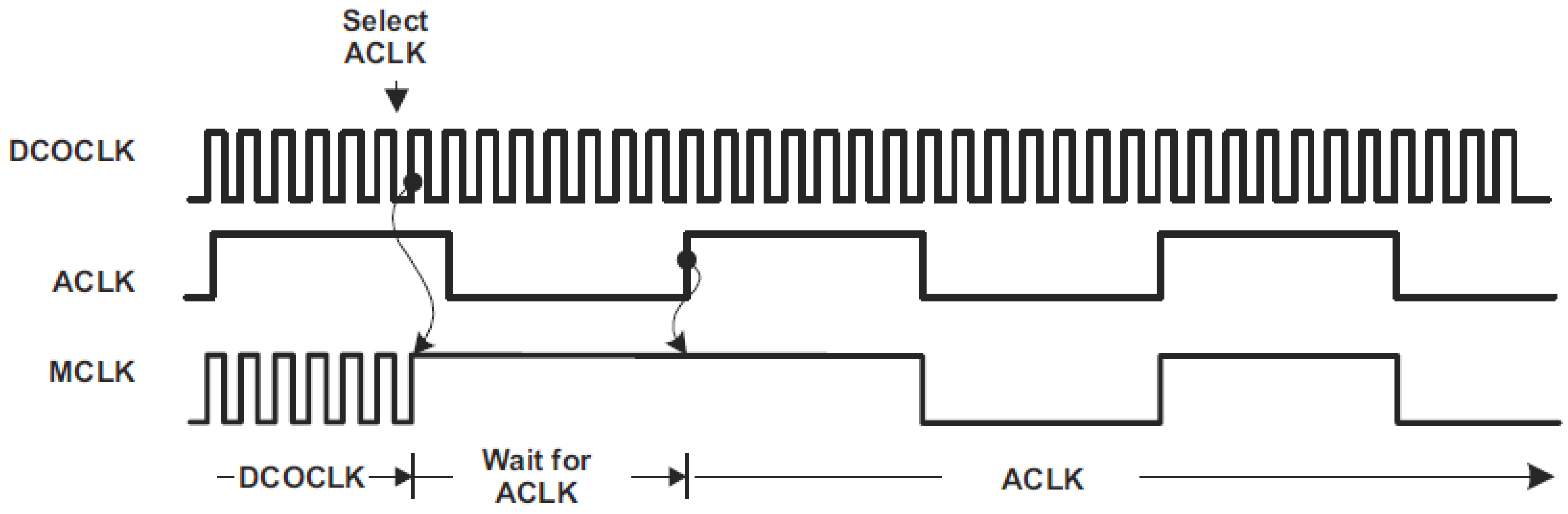
- The Clock System module incorporates an oscillator-fault fail-safe feature. This feature detects an oscillator fault for XT1 and XT2. The available fault conditions are:
  - Low-frequency oscillator fault (XT1OFFG) for XT1 in LF mode
  - High-frequency oscillator fault (XT1OFFG) for XT1 in HF mode
  - High-frequency oscillator fault (XT2OFFG) for XT2
  - External clock signal faults for all bypass modes; XT1BYPASS = 1 or XT2BYPASS = 1
- The crystal oscillator fault bits XT1OFFG and XT2OFFG are set if the corresponding crystal oscillator is turned on and not operating properly. Once set, the fault bits remain set until reset in software, even if the fault condition no longer exists. If the user clears the fault bits and the fault condition still exists, the fault bits are automatically set again, otherwise they remain cleared.

MSP430 CS Module Fail-Safe Operation

- The OFIFG oscillator-fault interrupt flag is set and latched at POR or when any oscillator fault (XT1OFFG or XT2OFFG) is detected. When OFIFG is set and OFIE is set, the OFIFG requests a user NMI. When the interrupt is granted, the OFIE is not reset automatically as it is in previous MSP430 families. It is no longer required to reset the OFIE. NMI entry and exit circuitry removes this requirement. The OFIFG flag must be cleared by software. The source of the fault can be identified by checking the individual fault bits.
- If XT1 in LF mode is sourcing any system clock (ACLK, MCLK, or SMCLK), and a fault is detected, the system clock is automatically switched to the VLO for its clock source (VLOCLK). Similarly, if XT1 in HF mode is sourcing any system clock and a fault is detected, the system clock is automatically switched to MODOSC for its clock source (MODCLK). When XT2 (if available) is sourcing any system clock and a fault is detected, the system clock is automatically switched to MODOSC for its clock source (MODCLK). The fail-safe logic does not change the respective SELA, SELM, and SELS bit settings. The fail-safe mechanism behaves the same in normal and bypass modes.

MSP430 Synchronization of Clock Signals

- When switching ACLK, MCLK, or SMCLK from one clock source to the another, the switch is synchronized to avoid critical race conditions:
  - The current clock cycle continues until the next rising edge.
  - The clock remains high until the next rising edge of the new clock.
  - The new clock source is selected and continues with a full high period.



Switch MCLK from DCOCLK to XT1CLK

MSP430 MODOSC Operation

- The Clock System module also supports an internal oscillator, MODOSC, that is used by the power management module and, optionally, by other modules in the system. It is also used as a fail-safe clock source. The MODOSC sources MODCLK.
- To conserve power, MODOSC is powered down when not needed and enabled only when required. When the MODOSC source is required, the respective module requests it.
- MODOSC is enabled based on unconditional and conditional requests. Setting ODOSCREQEN enables conditional requests.
- Unconditional requests are always enabled. It is not necessary to set MODOSCREQEN for modules that use unconditional requests; for example, PMM, ADC, and fail-safe.

Clock Configuration – Demonstration Code

```
//------------------------------------------------------------------------------
// Clock Configurations
// Initial clock configuration, runs immediately after boot.
// Disables 1ms // watchdog timer, then configures MCLK to 24MHz and SMCLK to
// 3MHz. Since the X1 oscillator is not currently connected, X1CLK is an unknown
// speed (probably ~10kHz).
//------------------------------------------------------------------------------
void Init_Clocks(void){

// Disable watchdog
  WDTCTL = WDTPW | WDTHOLD;

// Clocks:
//   MCLK: 8MHz
//   SMCLK: 8MHz
// Startup clock system in max. DCO setting ~8MHz
// This value is closer to 10MHz on untrimmed parts
  CSCTL0 = CSKEY;                     // Unlock register
  CSCTL1 |= DCOFSEL0 + DCOFSEL1;      // Set max. DCO setting [8MHz]
  CSCTL2 = SELA_1 + SELS_3 + SELM_3;  // set ACLK = vlo; SMCLK = DCO; MCLK = DCO
  CSCTL3 = DIVA_0 + DIVS_0 + DIVM_0;  // set all dividers /1
  CSCTL0_H = CSLOCK;                  // Lock registers

}
//------------------------------------------------------------------------------
```