// ----------------------------------------------------------------------------

// myClocks.c (for lab\_04a\_clock project) ('F5529 Launchpad)

//

// This routine sets ACLK to run from REFO, then configures MCLK and SMCLK to

// run from the and high-freq internal clock source (DCO).

//

// Oscillators:

// DCO = 8MHz (default is ~1MHz) Internal high-frequency oscillator

// REFO = 32KHz Internal 32KHz reference oscillator

// MODOSC = 5MHz Internal 5MHz oscillator

// VLO = ~10KHz Internal very low power, low frequency oscillator

// XT1 = --KHz (not configured) External crystal input

// XT2 = --MHz (not configured) External crystal input

//

// Reference Clock:

// FLL = REFO = 32KHz Internal reference clock; used for calibrating DCO at runtime

// Internal Clocks:

// ACLK = REFO = 32KHz

// SMCLK = DCO = 8MHz

// MCLK = DCO = 8MHz

// MODCLK = MODOSC = 5MHz (default)

// ----------------------------------------------------------------------------

//\*\*\*\*\* Header Files \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//#include <stdbool.h>

**#include** <driverlib/MSP430F5xx\_6xx/driverlib.h>

**#include** "myClocks.h"

//\*\*\*\*\* Defines \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**#define** LF\_CRYSTAL\_FREQUENCY\_IN\_HZ 32768

**#define** HF\_CRYSTAL\_FREQUENCY\_IN\_HZ 4000000

**#define** MCLK\_DESIRED\_FREQUENCY\_IN\_KHZ 8000

**#define** MCLK\_FLLREF\_RATIO MLCK\_DESIRED\_FREQUENCY\_IN\_KHZ / (UCS\_REFOCLK\_FREQUENCY/1024) // Ratio = 250

//\*\*\*\*\* Global Variables \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

uint32\_t myACLK = 0;

uint32\_t mySMCLK = 0;

uint32\_t myMCLK = 0;

//\*\*\*\*\* initClocks \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**void** **initClocks**(**void**) {

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Configure core voltage level

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Set core voltage level to handle 8MHz clock rate

PMM\_setVCore( PMM\_CORE\_LEVEL\_1 );

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Configure Oscillators

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Set the XT1/XT2 crystal frequencies used on the LaunchPad, and connected

// to the clock pins, so that driverlib knows how fast they are (these are

// needed for the DriverLib clock 'get' and crystal start functions)

UCS\_setExternalClockSource(

LF\_CRYSTAL\_FREQUENCY\_IN\_HZ, // XT1CLK input

HF\_CRYSTAL\_FREQUENCY\_IN\_HZ // XT2CLK input

);

// Verify if the default clock settings are as expected

myACLK = UCS\_getACLK();

mySMCLK = UCS\_getSMCLK();

myMCLK = UCS\_getMCLK();

UCS\_initClockSignal(

UCS\_ACLK, // Clock you're configuring

**USC\_REFOCLK\_SELECT, // Clock source**

UCS\_CLOCK\_DIVIDER\_1 // Divide down clock source by this much

);

UCS\_initClockSignal(

UCS\_FLLREF, // Clock you're configuring

USC\_REFOCLK\_SELECT, // Clock source

UCS\_CLOCK\_DIVIDER\_1 // Divide down clock source by this much

);

// Set MCLK and SMCLK to use the DCO/FLL as their oscillator source (8MHz)

// The function does a number of things: Calculates required FLL settings; Configures FLL and DCO,

// and then sets MCLK and SMCLK to use the DCO (with FLL runtime calibration)

UCS\_initFLLSettle(

MCLK\_DESIRED\_FREQUENCY\_IN\_KHZ, // MCLK frequency

**MCLK\_FLLREF\_RATIO // Ratio between MCLK and FLL's reference clock source**

);

// Verify that the modified clock settings are as expected

myACLK = UCS\_getACLK();

mySMCLK = UCS\_getSMCLK();

myMCLK = UCS\_getMCLK();

}