#include <stdint.h>

#include <stdbool.h>

#include "inc/tm4c1294ncpdt.h"

#include "inc/hw\_memmap.h"

#include "inc/hw\_types.h"

#include "driverlib/sysctl.h"

#include "driverlib/interrupt.h"

#include "driverlib/gpio.h"

#include "driverlib/timer.h"

int count = 0;

int main(void)

{

uint32\_t ui32Period;

uint32\_t ui32SysClkFreq;

//SET SYSTEM CLK

ui32SysClkFreq = SysCtlClockFreqSet((SYSCTL\_XTAL\_25MHZ | SYSCTL\_OSC\_MAIN | SYSCTL\_USE\_PLL | SYSCTL\_CFG\_VCO\_480), 120000000);

//ENABLE PERIPHERALS

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPION);

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_TIMER0);

//CONFIGURE PERIPHERALS

GPIOPinTypeGPIOOutput(GPIO\_PORTN\_BASE,GPIO\_PIN\_1);

TimerConfigure(TIMER0\_BASE, TIMER\_CFG\_SPLIT\_PAIR | TIMER\_CFG\_A\_ONE\_SHOT);

//SET THE PERIOD OF THE TIMER

ui32Period = ui32SysClkFreq;

TimerLoadSet(TIMER0\_BASE, TIMER\_A, ui32Period -1);

//ENABLE INTERRUPTS

IntEnable(INT\_TIMER0A);

TimerIntEnable(TIMER0\_BASE, TIMER\_TIMA\_TIMEOUT);

IntMasterEnable();

//ENABLE TIMERS

TimerEnable(TIMER0\_BASE, TIMER\_A);

while(1)

{

}

}

//ISR

void Timer0IntHandler(void)

{

// Clear the timer interrupt flag

TimerIntClear(TIMER0\_BASE, TIMER\_TIMA\_TIMEOUT);

GPIOPinWrite(GPIO\_PORTN\_BASE,GPIO\_PIN\_1,0x2);

}