/\* p4\_1.c: Sending "YES" to UART0 on TI ARM Launchpad (TM4C123GH6PM) \*/

/\* UART0 is on USB/Debug \*/

/\* Use TeraTerm to see the message "YES" on a PC \*/

#include <stdint.h>

#include "tm4c123gh6pm.h"

void UART0Tx(char c);

void delayMs(int n);

int main(void)

{

SYSCTL->RCGCUART |= 1; /\* provide clock to UART0 \*/

SYSCTL->RCGCGPIO |= 1; /\* enable clock to PORTA \*/

/\* UART0 initialization \*/

UART0->CTL = 0; /\* disable UART0 \*/

UART0->IBRD = 104; /\* 16MHz/16=1MHz, 1MHz/104=9600 baud rate \*/

UART0->FBRD = 11; /\* fraction part, see Example 4-4 \*/

UART0->CC = 0; /\* use system clock \*/

UART0->LCRH = 0x60; /\* 8-bit, no parity, 1-stop bit, no FIFO \*/

UART0->CTL = 0x301; /\* enable UART0, TXE, RXE \*/

/\* UART0 TX0 and RX0 use PA0 and PA1. Set them up. \*/

GPIOA->DEN = 0x03; /\* Make PA0 and PA1 as digital \*/

GPIOA->AFSEL = 0x03; /\* Use PA0,PA1 alternate function \*/

GPIOA->PCTL = 0x11; /\* configure PA0 and PA1 for UART \*/

delayMs(1); /\* wait for output line to stabilize \*/

for(;;)

{

UART0Tx('Y');

UART0Tx('E');

UART0Tx('S');

UART0Tx(' ');

}

}

/\* UART0 Transmit \*/

/\* This function waits until the transmit buffer is available then \*/

/\* writes the character in the transmit buffer. It does not wait \*/

/\* for transmission to complete. \*/

void UART0Tx(char c)

{

while((UART0->FR & 0x20) != 0); /\* wait until Tx buffer not full \*/

UART0->DR = c; /\* before giving it another byte \*/

}

/\* Append delay functions and SystemInit() here \*/