**NO DELAY**

/\*

\* Engr220L - Lab 9

\* Date: today

\* Author: Jacob BRINK

\*

\* counts up with no delay

\*

\*/

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

/\* INCLUDES \*/

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

.include "nios\_macros.s"

.include "nios\_defs.s" /\* .equ statements specific to this system \*/

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

/\* CONSTANTS \*/

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

.equ MS100, 5000000 /\* number of clock cycles in 100 msec provided as example \*/

.equ MS100LOW, 0x4b40 /\* 16 least signif bits \*/

.equ MS100HIGH, 0x4c /\* 16 most signif bits \*/

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

/\* TEXT SECTION \*/

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

.text

/\* Place the main routine at the reset address \*/

.org RESET\_VECTOR

/\* Program start location must be identified \*/

.global \_start

\_start:

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

/\* MAIN PROGRAM CODE \*/

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

MAIN\_PROG\_INIT:

/\* TODO: initialize PIO devices if needed \*/

/\* TODO: initialize the timer with the proper timeout period \*/

MAIN\_PROG:

addi r15, r0, 0x40000

movia r17, LEDR\_BASE

add r16, r0, r0

LOOP\_TOP:

addi r16, r16, 0x1

stwio r16, 0(r17)

blt r16, r15, LOOP\_TOP

MAIN\_PROG\_END:

/\* infinite loop to keep out of global memory, useful for final breakpoint \*/

br MAIN\_PROG\_END

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

/\* DATA SECTION \*/

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

.data

/\* if any global variables are needed, place them here \*/

.end

**COUNTER DELAY**

/\*

\* Engr220L - Lab 9

\* Date: today

\* Author: Jacob Brink

\*

\* <TODO describe the program summary>

\*

\*/

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

/\* INCLUDES \*/

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

.include "nios\_macros.s"

.include "nios\_defs.s" /\* .equ statements specific to this system \*/

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

/\* CONSTANTS \*/

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

.equ MS100, 5000000 /\* number of clock cycles in 100 msec provided as example \*/

.equ MS100LOW, 0x4b40 /\* 16 least signif bits \*/

.equ MS100HIGH, 0x4c /\* 16 most signif bits \*/

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

/\* TEXT SECTION \*/

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

.text

/\* Place the main routine at the reset address \*/

.org RESET\_VECTOR

/\* Program start location must be identified \*/

.global \_start

\_start:

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

/\* MAIN PROGRAM CODE \*/

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

MAIN\_PROG\_INIT:

/\* TODO: initialize PIO devices if needed \*/

/\* TODO: initialize the timer with the proper timeout period \*/

MAIN\_PROG:

movia r20, 0x6FFFF0

addi r15, r0, 0x40000

movia r17, LEDR\_BASE

add r16, r0, r0

LOOP\_TOP:

addi r16, r16, 0x1

stwio r16, 0(r17)

/\*

delay\*/

add r18, r0, r0

DELAY\_LOOP:

addi r18, r18, 0x1

blt r18, r20, DELAY\_LOOP

blt r16, r15, LOOP\_TOP

MAIN\_PROG\_END:

/\* infinite loop to keep out of global memory, useful for final breakpoint \*/

br MAIN\_PROG\_END

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

/\* DATA SECTION \*/

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

.data

/\* if any global variables are needed, place them here \*/

.end

**TIMER**

/\*

\* Engr220L - Lab 9\* Date: today

\* Author: Jacob Brink

\*

\* <TODO describe the program summary>

\*

\*/

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

/\* INCLUDES \*/

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

.include "nios\_macros.s"

.include "nios\_defs.s" /\* .equ statements specific to this system \*/

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

/\* CONSTANTS \*/

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

.equ MS100, 5000000 /\* number of clock cycles in 100 msec provided as example \*/

.equ MS100LOW, 0x4b40 /\* 16 least signif bits \*/

.equ MS100HIGH, 0x4c /\* 16 most signif bits \*/

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

/\* TEXT SECTION \*/

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

.text

/\* Place the main routine at the reset address \*/

.org RESET\_VECTOR

/\* Program start location must be identified \*/

.global \_start

\_start:

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

/\* MAIN PROGRAM CODE \*/

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

MAIN\_PROG\_INIT:

/\* TODO: initialize PIO devices if needed \*/

MAIN\_PROG:

movia r17, LEDR\_BASE

.equ TIME\_CYCLES, 0x17D783F

.equ TIME\_OUT\_MASK, 0x1

/\* initialize the timer with the proper timeout period \*/

movia r10, TIME\_CYCLES /\* PERIOD (clock ticks for .5 seconds) \*/

movia gp, TIMER\_BASE

/\* initialize PERIODH and PERIODL to PERIOD \*/

addi r10, r0, 0x17D

stwio r10, PERIODH\_OFFSET(gp)

addi r10, r0, 0x783F

stwio r10, PERIODL\_OFFSET(gp)

/\* Turn on RUN bit in status\_control \*/

stwio r0, STATUS\_OFFSET(gp)

/\* Turn on CONT bit and START bit in control\_OFFSET \*/

addi r10, r0, 0b110

stwio r10, CONTROL\_OFFSET(gp)

/\*movia r18, r0, 0x1 r9 is mask to get timeout bit \*/

/\* initialize r16 (loop counter) to 0\*/

add r16, r0, r0

movia r15, 0x40000

RESTART\_TIMER:

stwio r0, STATUS\_OFFSET(gp)

br LOOP\_TOP

LOOP\_TOP:

/\* increment counter and leds \*/

addi r16, r16, 0x1

stwio r16, 0(r17)

/\*delay, loop until some time out bit goes to one\*/

DELAY\_LOOP:

ldwio r10, STATUS\_OFFSET(gp)

andi r10, r10, TIME\_OUT\_MASK

bgt r10, r0, RESTART\_TIMER

br DELAY\_LOOP

blt r16, r15, LOOP\_TOP

br MAIN\_PROG

MAIN\_PROG\_END:

/\* infinite loop to keep out of global memory, useful for final breakpoint \*/

br MAIN\_PROG\_END

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

/\* DATA SECTION \*/

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

.data

/\* if any global variables are needed, place them here \*/

.end