Assignment 1

ECE 355

Fall 2024

#define PBIN (volatile unsigned char \*)0xFFFFFFF3

#define PBOUT (volatile unsigned char \*)0xFFFFFFF4

#define PBDIR (volatile unsigned char \*)0xFFFFFFF5

#define PSTAT (volatile unsigned char \*)0xFFFFFFF6

#define CNTM (volatile unsigned int \*)0xFFFFFFD0

#define CTCON (volatile unsigned char \*)0xFFFFFFD8

#define CTSTAT (volatile unsigned char \*)0xFFFFFFD9

#define IVECT (volatile unsigned int \*)(0x20)

interrupt void intserv();

volatile unsigned char digit = 0; /\* digit for display \*/

int main()

{

unsigned char sample = 0; /\* Port B input sample \*/

\*PBDIR = 0b11110000; /\* Set Port B direction \*/

\*CTCON = 0b10; /\* if the Timer is running, then Stop it \*/

\*CTSTAT = 0b0; /\* Clear "Reached 0" flag \*/

\*CNTM = 100000000; /\* Initialize 1s timeout \*/

\*IVECT = (unsigned int \*)&intserv; /\* Setup interrupt vector \*/

asm("MoveControl PSR,#0b1000000"); /\* CPU responds to IRQ \*/

\*CTCON = 0b1; /\* Start Timer, disable interrupts for now \*/

\*PBOUT = 0b0; /\* Display 0 \*/

while (1)

{

while ((\*PSTAT & 0b100) == 0); /\* Wait for PBIN update \*/

sample = \*PBIN & 0b11; /\* Sample PBIN, isolate bits [1:0] \*/

if (sample == 0b1)

{ /\* E = 0, D = 1 \*/

\*CTCON |= 0b10000; /\* Enable Timer interrupts \*/

}

else if (sample == 0b10)

{ /\* E = 1, D = 0 \*/

\*CTCON &= 0b11101111; /\* Disable Timer interrupts \*/

}

}

exit(0);

}

interrupt void intserv()

{

\*CTSTAT = 0b0; /\* Clear "Reached 0" flag \*/

digit = (digit + 1) % 10; /\* Increment digit \*/

\*PBOUT = digit << 4; /\* Update display \*/

}