

/\* Formål: Anvendelse af interrupt indput pin

\* Fil: IntPin.c

\* Forfatter STH

\* dato 28/3 2024

\* Beskrivelse: I main skifter RC4 status hvert sekund så man kan se programmet

\* er igang, ISR skifter status på RC3 hver gang RA2 aktiveres det er en god

\* ide at sætte en god stor kondensator over modstanden på kontakten for at

\* undgå prell.

\*/

#define \_XTAL\_FREQ 4000000

#include <xc.h>

#pragma config FOSC = INTOSCIO // Oscillator Selection bits (HS oscillator)

#pragma config WDTE = OFF // Watchdog Timer Enable bit (WDT disabled)

#pragma config PWRTE = ON // Power-up Timer Enable bit (PWRT enabled)

#pragma config BOREN = ON // Brown-out Reset Enable bit (BOR enabled)

#pragma config CPD = OFF // (Data EEPROM code protection off)

#pragma config CP = OFF // (Code protection off)

## void setup (){

ANSEL = 0B00000000;

CMCON0 = 0B00000111;

TRISA = 0B00000100; //RA2 er indgang

TRISC = 0B00000000;

INTCON = 0B11010000; //interrupts enabled, Interrupt pin aktiv

OPTION\_REG = 0B01000000; //interrupt pin opadgående flanke

}

## void \_\_interrupt() ISR() {

if (INTF == 1){ // hvis der er flere interrupts

RC3 = !RC3; // RC3 skifter Høj/Lav hver interrupt

INTF = 0; // Interruptflaget cleares

}

}

## void main(void)

{

setup();

while(1)

{

RC4 = !RC4; // RC4 skiftes hvert sekund for at se

\_\_delay\_ms(1000); // om programmet kører

}

}

C:/Users/sth/MPLABXProjects/Racer2023sth.X/InterruptPin.c