

```
#include "Proj_1E_header_file.h"

int m;
unsigned int PORT_1, PORT_2;

int main (void){
    unsigned long counter_squared, counter=1; //32 bits are reserved for each of these variables

    m=0;
    setup_HW;
    PORT_1=0xFFFF; //Equivalent to 0b1111111111111111
    PORT_2 = 0;
    I2C_Tx_2_integers(PORT_1, PORT_2);
    sei();
    T1_65ms_clock_tick(); //This subroutine starts HW clock Timer 1 that generates an interrupt every 65ms

    while(1){
        counter_squared = counter*counter;
        if((!(counter%33))&& (switch_2_down)){ //Interrupts enable several process to take place simultaneously
            Num_to_PC_U(10, counter); Char_to_PC('\t'); //In this case squares are calculated some of which are printed out
            Num_to_PC_U(10, counter*counter); newline(); //"counter%33" is only zero when counter is 33, 66, 99 etc.
            }counter = (counter + 1)%0x10000; //Only print out results if Switch_2 has been pressed
            Timer_T2_sub(T2_delay_2ms);} //Code in this while-loop could be interrupted at any point
        } //limits the value of counter to avoid overflow and garbage out.

    /*****ISR called evry 65ms to increment the display*****/
    ISR(TIMER1_OVF_vect) {
        if(m <=15){PORT_2 = (PORT_2 << 1) | 1; //PORT_2 is shifted one place to the left
            PORT_1 = PORT_1 << 1;} //An additional 1 is placed in the most RH bit using "|1".

            else
            {PORT_2 = (PORT_2 >> 1); //PORT_2 now shifts right
            PORT_1 = ~(~PORT_1 >> 1);} //see below for quick look at the logic

            I2C_Tx_2_integers(PORT_1, PORT_2);m++; //increment "m";
        if (m==32)m=0;} //Reset "m" when it gets to 32
    }
```