**Activity 1**

Write a program to get data from port B and send it to port C continuously while an interrupt will do the following: One of the timers will toggle pin PB1 every 100 microseconds.

Question does not specifiy which timer/mode/prescalar to use, so I used timer 1 in normal mode w/ no prescalar. I know it says to toggle pin PB1, but Dr. Adla told us in class to use D1 since PORTB and PORTC are already all being used. This also goes for Activity 2, where PD1 is still used instead of PB1. We were also instructed to use the Function Generator in the lab to generate the square wave for the external interrupts, so I didn’t make my own square wave through the Arduino.

#include <avr/io.h>

#include <avr/interrupt.h>

int main(void)

{

//data direction, port B all input, port C all output, port D1 is output, then for activity 2, ports D2/D3 are input, port D4 is output

DDRB = 0x00;

DDRC = 0xFF;

DDRD |= 1<<1;

//timer 0 setup, CTC mode w/ 8 prescalar

TCCR0A = 0x02;

TCCR0B = 0x02;

//set output compare

OCR0A = 200-1;

//interrupt mask

TIMSK0 = 1<<OCIE0A;

//enable global interrupt

sei();

//wait here, send data

while (1) {

PORTC = PINB;

}

return 0;

}

//timer compare interrupt

ISR(TIMER0\_COMPA\_vect) {

//upon interrupt starting, toggle pin 1 on PORTD per rawa's instructions

PORTD ^= 1<<1;

}

**YouTube Link:**

[**https://youtube.com/shorts/xxkinfQe\_hU?feature=share**](https://youtube.com/shorts/xxkinfQe_hU?feature=share)

**Picture of Wave:**

A screen shot of a computer

Description automatically generated

**Activity 2**

Add the following interrupt to Activity 1:

Assume a square wave is connected to an external interrupt, and any time a L-to-H pulse comes in, a single LED connected to PB2 is turned on, and any time a H-to-L pulse comes in, the LED is turned off.

The rate of "On" and "Off" is the same as the square wave frequency.

#include <avr/io.h>

#include <avr/interrupt.h>

int main(void)

{

//data direction, port B all input, port C all output, port D1 is output, then for activity 2, ports D2/D3 are input, port D4 is output

DDRB = 0x00;

DDRC = 0xFF;

DDRD |= (1<<4 | 1<<1);

DDRD &= ~((1<<3 | 1<<2));

//timer 0 setup, CTC mode w/ 8 prescalar

TCCR0A = 0x02;

TCCR0B = 0x02;

//set output compare

OCR0A = 200-1;

//interrupt masks

EIMSK = 1<<INT0 | 1<<INT1;

TIMSK0 = 1<<OCIE0A;

//external interrupt setup (INT 0 is low to high, INT1 is high to low, EICRA = 0000 1110

EICRA = 0x0E;

//enable global interrupt

sei();

//for activity 2, turn LED off initially

PORTD &= ~(1<<4);

//wait here, send data

while (1) {

PORTC = PINB;

}

return 0;

}

//timer compare interrupt

ISR(TIMER0\_COMPA\_vect) {

//upon interrupt starting, toggle pin 1 on PORTD per rawa's instructions

PORTD ^= 1<<1;

}

//external interrupt 0

ISR(INT0\_vect) {

//int0, low to high pulse, turn LED on

PORTD |= (1<<4);

}

//external interrupt 1

ISR(INT1\_vect) {

//int 1, high to low pulse, turn LED off

PORTD &= ~(1<<4);

}

**YouTube Link:**

[**https://youtube.com/shorts/guxrXpLkVMY?feature=share**](https://youtube.com/shorts/guxrXpLkVMY?feature=share)

**Picture of Wave:**

A screen shot of a computer

Description automatically generated