



Instituto Politécnico Nacional



Escuela Superior de Cómputo

Generador de ondas

TAREA 9

Materia:

Introducción a los microcontroladores

Grupo:

3CM16

Profesor:

Pérez Pérez José Juan

Integrantes:

Castro Cruces Jorge Eduardo

Cortes Ramírez Roberto Carlos

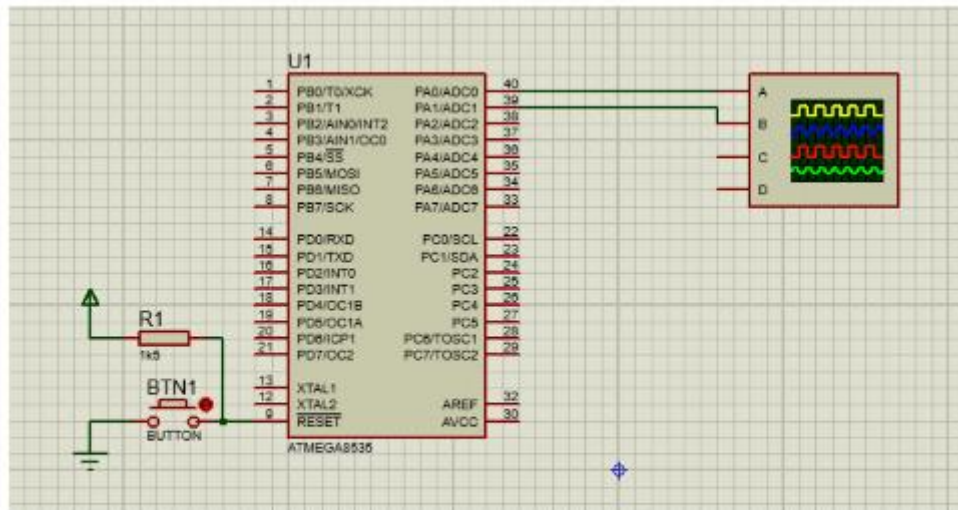
Domínguez Acosta José Práxedes

Fecha:

Jueves, 11 de noviembre de 2021

Descripción del problema

Descripción:



Ejercicios

- Escribir una secuencia de instrucciones para generar una señal de 10 KHz (Suponer que se está usando el oscilador interno a 1 MHz). Utilizando:
 - Desbordamientos atendidos por interrupción.
- Repetir el ejercicio anterior, pero ahora la frecuencia de la señal deberá ser de 500 Hz.
- Combinar los dos ejercicios anteriores de manera que el microcontrolador genere ambas señales en forma simultánea.

Código del programa #1

```
1.      .include "m8535def.inc"
2.      .def aux = r16
3.      .def msk = r17
4.      .def sem = r18
5. reset:
6.      rjmp main
7.      .org $009
8.      rjmp onda0
9. main:
10.     ldi aux, low(ramend)
11.     out spl, aux
12.     ldi aux, high(ramend)
13.     out sph, aux
14.     ser aux
15.     out ddra, aux
16.     ldi aux, 1
17.     out tccr0, aux
18.     ldi aux, 1
19.     out tmsk, aux
20.     sei
21.     ldi msk, 0b00000001
22.     ldi sem, 214
23. nada:
24.     rjmp nada
25.
26. onda0:
27.     nop
28.     out tcnt0, sem
29.     in aux, pina
30.     eor aux, msk
31.     out porta, aux
32.     reti
33.
```

Código del programa #2

```
1.      .include "m8535def.inc"
2.      .def aux = r16
3.      .def aux2 = r17
4.
5. reset:
6.      rjmp main
7.      .org $009
8.      rjmp onda
9. main:
10.     ldi aux, low(RAMEND)
11.     out spl, aux
12.     ldi aux, high(RAMEND)
13.     out sph, aux
14.     rcall config_io
15. fin:
16.     nop
17.     nop
18.     rjmp fin
19. config_io:
20.     ser aux
21.     out ddra, aux
22.     ldi aux, 2
23.     out tccr0, aux
24.     ldi aux, 1
25.     out timsk, aux
26.     sei
27.     ldi aux2, 132
28.     ret
29. onda:
30.     nop
31.     out tcnt0, aux2
32.     in aux, pina
33.     com aux
34.     out porta, aux
35.     reti
36.
```

Código del programa #3

```
1.      .include "m8535def.inc"
2.      .def aux = r16
3.      .def msk = r17
4.      .def msk2 = r19
5.      .def sem = r18
6.      .def sem2 = r20
7. reset:
8.      rjmp main
9.      .org $004
10.     rjmp ondal
11.     .org $009
12.     rjmp onda0
13. main:
14.     ldi aux, low(ramend)
15.     out spl, aux
16.     ldi aux, high(ramend)
17.     out sph, aux
18.     ser aux
19.     out ddra, aux
20.     ldi aux, 1
21.     out tccr0, aux
22.     ldi aux, 1
23.     out tccr2, aux
24.     ldi aux, 0b01000001
25.     out timsk, aux
26.     sei
27.     ldi msk, 0b00000001
28.     ldi msk2, 0b00000010
29.     ldi sem, 256-43
30.     ldi sem2, 256-125
31.     ldi r21, 10
32. nada:
33.     nop
34.     nop
35.     nop
36.     nop
37.     nop
38.     nop
39.     nop
40.     nop
41.     nop
42.     nop
43.     rjmp nada
44. onda0:
45.     nop
46.     out tcnt0, sem
```

```
47.         in aux,pina
48.         eor aux,msk
49.         out porta, aux
50.         reti
51.  ondal:
52.         out tcnt2,sem2
53.         in aux,pina
54.         eor aux,msk2
55.         out porta,aux
56.         reti
57.
```

Simulación en AVR Studio 4

AVR Studio - [C:\Users\georg\Desktop\ESCOM\8vo Semestre\Micros\Tareas\9GeneradorOndas\AVR\generadorOndas10KHz\generadorOndas10KHz.asm]

File Project Build Edit View Tools Debug Window Help

Trace Disabled

Project

- generadorOndas10KHz
 - Source Files
 - generadorOndas10KHz.asm
 - Included Files
 - Labels
 - Output
 - Object File

```
.include "m8535def.inc"
.def aux = r16
.def msk = r17
.def sem = r18

reset:
    rjmp main
    .org $009
    rjmp onda0

main:
    ldi aux, low(ramend)
    out spl, aux
    ldi aux, high(ramend)
    out sph, aux
    ser aux
    out ddra, aux
    ldi aux, 1
    out tccr0, aux
    ldi aux, 1
    out tmsk, aux
    sei
    ldi msk, 0b00000001
    ldi sem, 214

nada:
    rjmp nada

onda0:
    nop
    out tcnt0, sem
    in aux, pina
    eor aux, msk
```

I/O View

Name	Value
AD_CONVERTER	
ANALOG_COMPARA...	
CPU	
EEPROM	
EXTERNAL_INTERR...	
PORTA	
PORTB	
PORTC	
PORTD	
SPI	
TIMER_COUNTER_0	
TIMER_COUNTER_1	
TIMER_COUNTER_2	
TWI	
USART	

Name	Address	Value	Bits
------	---------	-------	------

Build

```
[.cseg] 0x000000 0x000003c 44 0 44 8192 0.5%
[.dseg] 0x000060 0x0000060 0 0 0 512 0.0%
[.eseg] 0x000000 0x0000000 0 0 0 512 0.0%
```

Assembly complete, 0 errors. 0 warnings

Build Message Find in Files Breakpoints and Tracepoints

ATmega8535 AVR Simulator Auto Ln 6, Col 1 CAP NUM OVR

12:59 p. m. 11/11/2021

AVR Studio - [C:\Users\georg\Desktop\ESCOM\8vo Semestre\Micros\Tareas\9GeneradorOndas\AVR\generadorOndas500Hz\generadorOndas500Hz.asm]

File Project Build Edit View Tools Debug Window Help

Trace Disabled

Project

- generadorOndas500Hz
 - Source Files
 - generadorOndas500Hz.asm
 - Included Files
 - Labels
 - Output
 - Object File

```
.include "m8535def.inc"
.def aux = r16
.def aux2 = r17

reset:
    rjmp main
    .org $009
    rjmp onda

main:
    ldi aux, low(RAMEND)
    out spl, aux
    ldi aux, high(RAMEND)
    out sph, aux
    rcall config_io

fin:
    nop
    nop
    rjmp fin

config_io:
    ser aux
    out ddra, aux
    ldi aux, 2
    out tccr0, aux
    ldi aux, 1
    out tmsk, aux
    sei
    ldi aux2, 132
    ret

onda:
    nop
```

I/O View

Name	Value
AD_CONVERTER	
ANALOG_COMPARA...	
CPU	
EEPROM	
EXTERNAL_INTERRUPT	
PORTA	
PORTB	
PORTC	
PORTD	
SPI	
TIMER_COUNTER_0	
TIMER_COUNTER_1	
TIMER_COUNTER_2	
TWI	
USART	

Name	Address	Value	Bits
------	---------	-------	------

Build

```
[.cseg] 0x000000 0x000042 50 0 50 8192 0.6%
[.dseg] 0x000060 0x000060 0 0 0 512 0.0%
[.eseg] 0x000000 0x000000 0 0 0 512 0.0%
```

Assembly complete, 0 errors. 0 warnings

Build Message Find in Files Breakpoints and Tracepoints

Loading event memory... ATmega8535 AVR Simulator Auto Stopped Ln 1, Col 1 CAP NUM OVR

12:59 p. m. 11/11/2021

AVR Studio - [C:\Users\georg\Desktop\ESCOM\8vo Semestre\Micros\Tareas\9GeneradorOndas\AVR\generadorOndas500Hz_10KHz\generadorOndas500Hz_10KHz.asm]

File Project Build Edit View Tools Debug Window Help

Trace Disabled

Project

- generatorOndas500Hz_10KHz
 - Source Files
 - generatorOndas500Hz_10KHz.asm
 - Included Files
 - Labels
 - Output
 - Object File

```
.include "m8535def.inc"
.def aux = r16
.def msk = r17
.def msk2 = r19
.def sem = r18
.def sem2 = r20

reset:
    rjmp main
    .org $004
    rjmp ondal
    .org $009
    rjmp onda0

main:
    ldi aux, low(ramend)
    out spl, aux
    ldi aux, high(ramend)
    out sph, aux
    ser aux
    out ddra, aux
    ldi aux, 1
    out tccr0, aux
    ldi aux, 1
    out tccr2, aux
    ldi aux, 0b01000001
    out tmsk, aux
    sei
    ldi msk, 0b00000001
    ldi msk2, 0b00000010
    ldi sem, 256-43
    ldi sem2, 256-125
```

I/O View

ANALOG_COMPARATOR

Name	Value
AD_CONVERTER	
ANALOG_COMPARATOR	
CPU	
EEPROM	
EXTERNAL_INTERRUPT	
PORTA	
PORTB	
PORTC	
PORTD	
SPI	
TIMER_COUNTER_0	
TIMER_COUNTER_1	
TIMER_COUNTER_2	
TWI	
USART	

Name	Address	Value	Bits
------	---------	-------	------

Build

```
[.cseg] 0x000000 0x000064 86 0 86 8192 1.0%
[.dseg] 0x000060 0x000060 0 0 0 512 0.0%
[.eseg] 0x000000 0x000000 0 0 0 512 0.0%
```

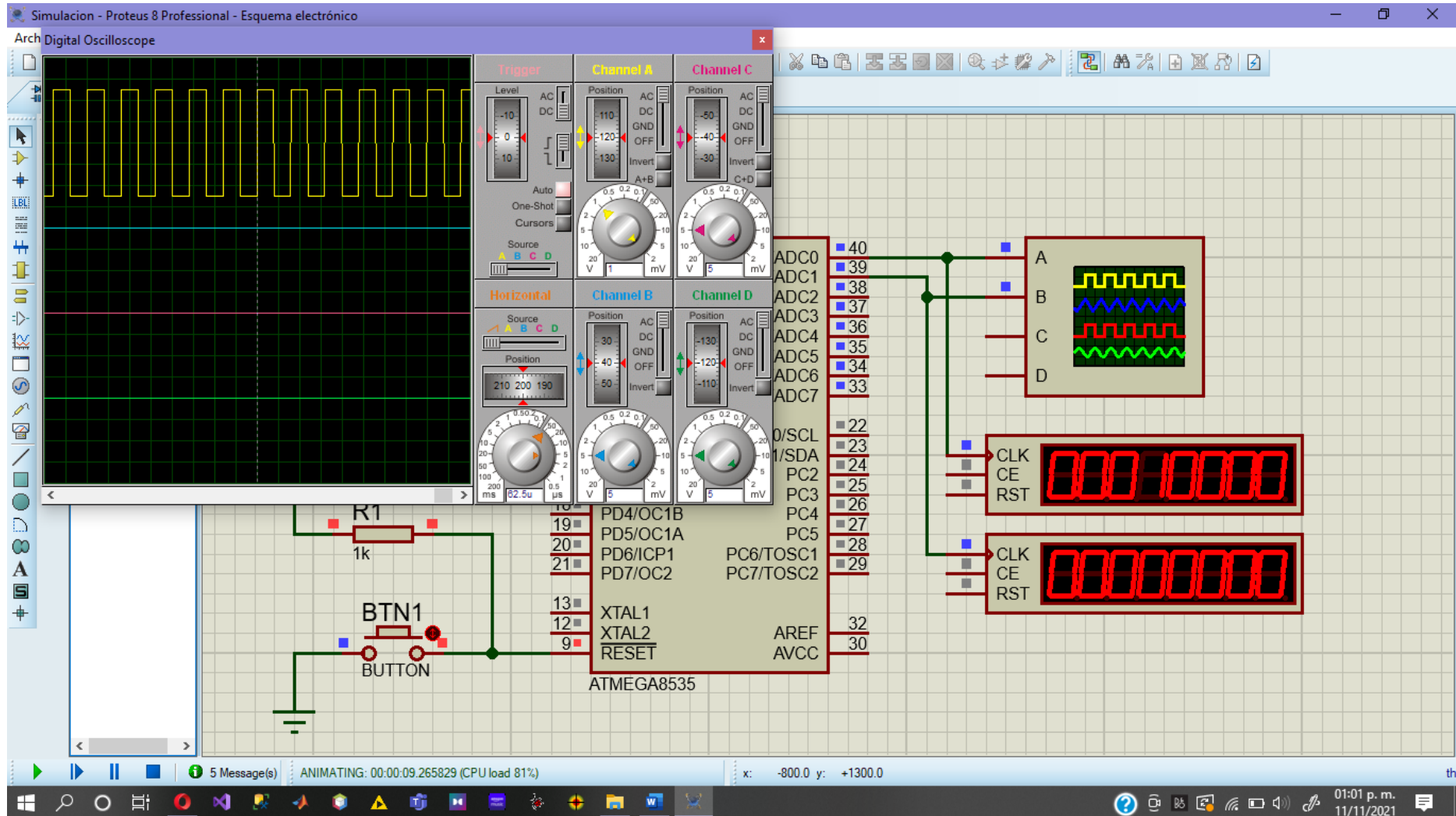
Assembly complete, 0 errors, 0 warnings

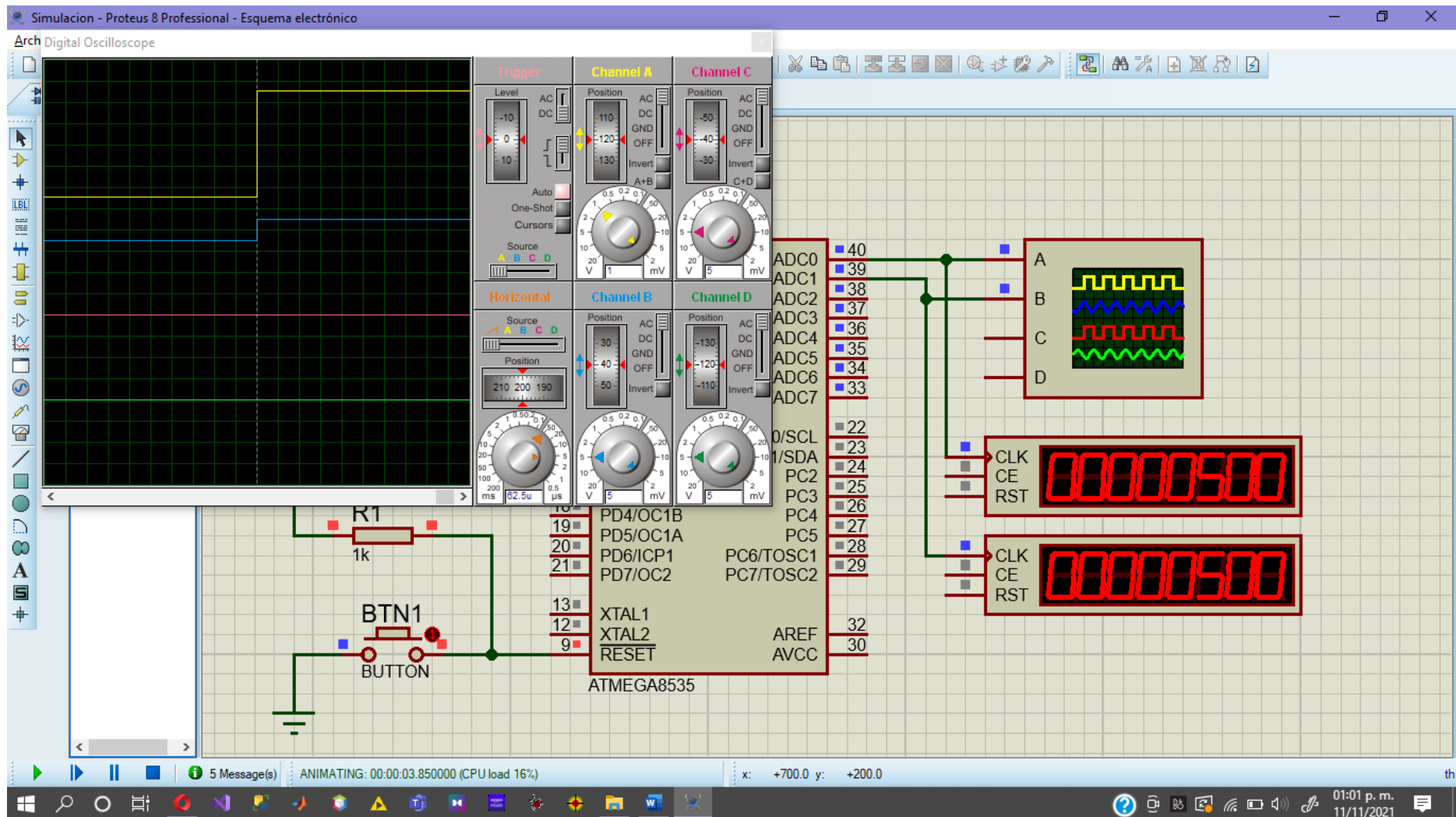
Build Message Find in Files Breakpoints and Tracepoints

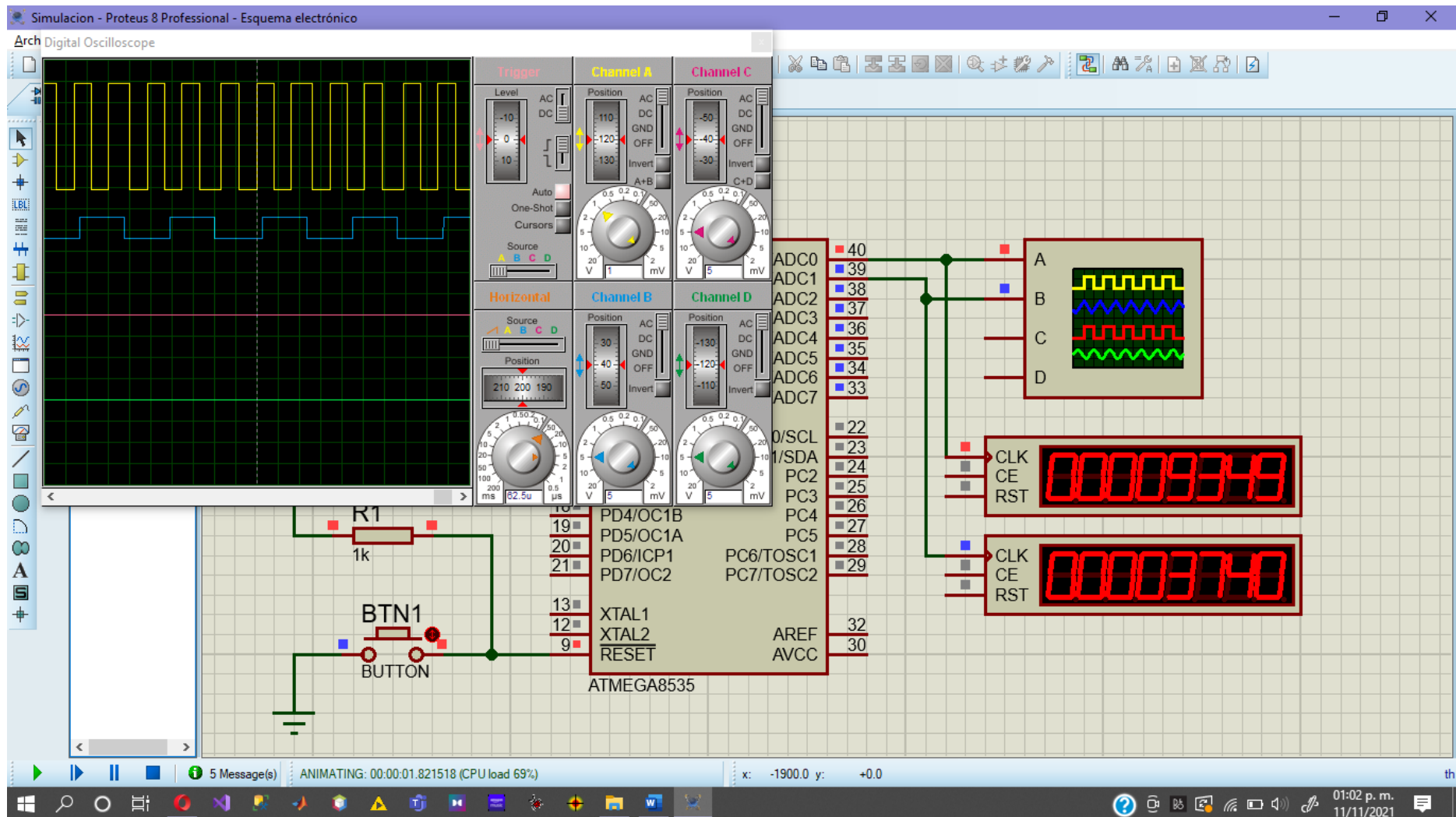
Loading event memory... ATmega8535 AVR Simulator Auto Ln 1, Col 1 CAP NUM OVR

01:00 p.m. 11/11/2021

Simulación en Proteus 8 Professional







Conclusiones

- **Castro Cruces Jorge Eduardo**

En esta práctica tuvimos dificultades para llevar a cabo la combinación de ambas señales, y poder visualizarlas en dos puertos distintos, pero finalmente nos ayudamos del código del profesor, para lograr la práctica.

- **Cortes Ramírez Roberto Carlos**

En esta práctica hicimos un código el cual podemos hacer una secuencia de instrucciones para generar señales. Para esto tuvimos que el osciloscopio trabaje a 1MHz, para después generar las señales a 10KHz. Para poder generar este código tuvimos que tener varias variables para así poder asociarlo con el simulador y que pueda correr sin problema.

Para poder ver cómo funciona usamos proteus en este fue un circuito simple, con un botón para reiniciar las señales y con el atmega8535 para poder usar el programa y conectado a ese mismo usamos un osciloscopio para ver como variaban las señales.

- **Domínguez Acosta José Práxedes**

En esta práctica primero se escribió una secuencia de instrucciones para generar una señal de 10 khz haciendo el uso de desbordamientos atendidos por interrupción para luego repetir el mismo procedimiento solo que esta vez se cambió la frecuencia a 500 hz, al final se realizó la combinación de ambos ejercicios en uno solo para que el microcontrolador las generara de manera simultánea como se pudo observar en la simulación en Proteus haciendo el uso del osciloscopio de dicha herramienta.