دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

گزارش کار آزمایشگاه ریزپردازنده

آزمایش شماره‌ی ۱۱

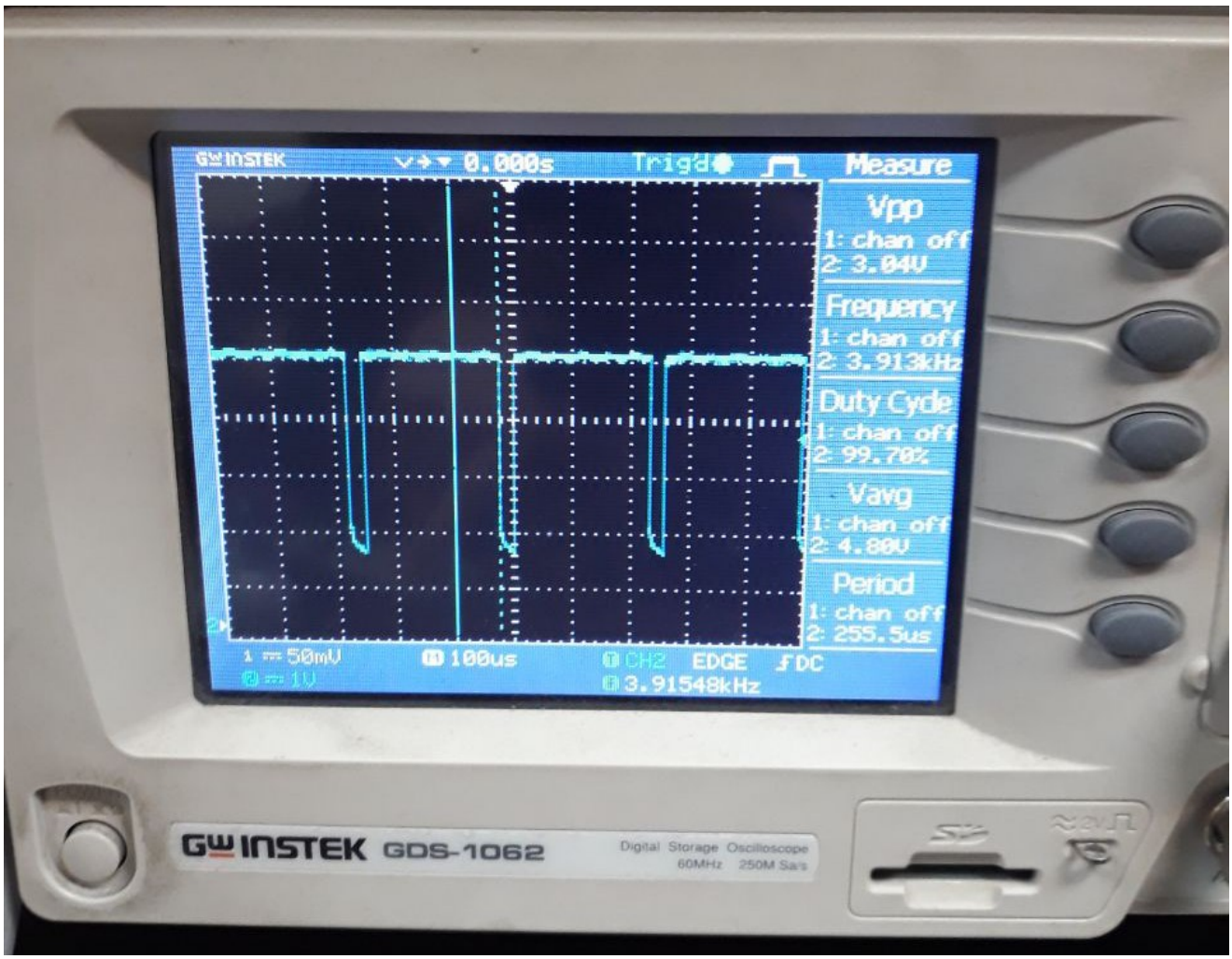
آریا وارسته‌نژاد – ۹۵۳۱۴۳۵
مهران تقیان – ۹۵۳۱۰۱۸
رضا توسلی – ۹۵۳۱۰۱۹

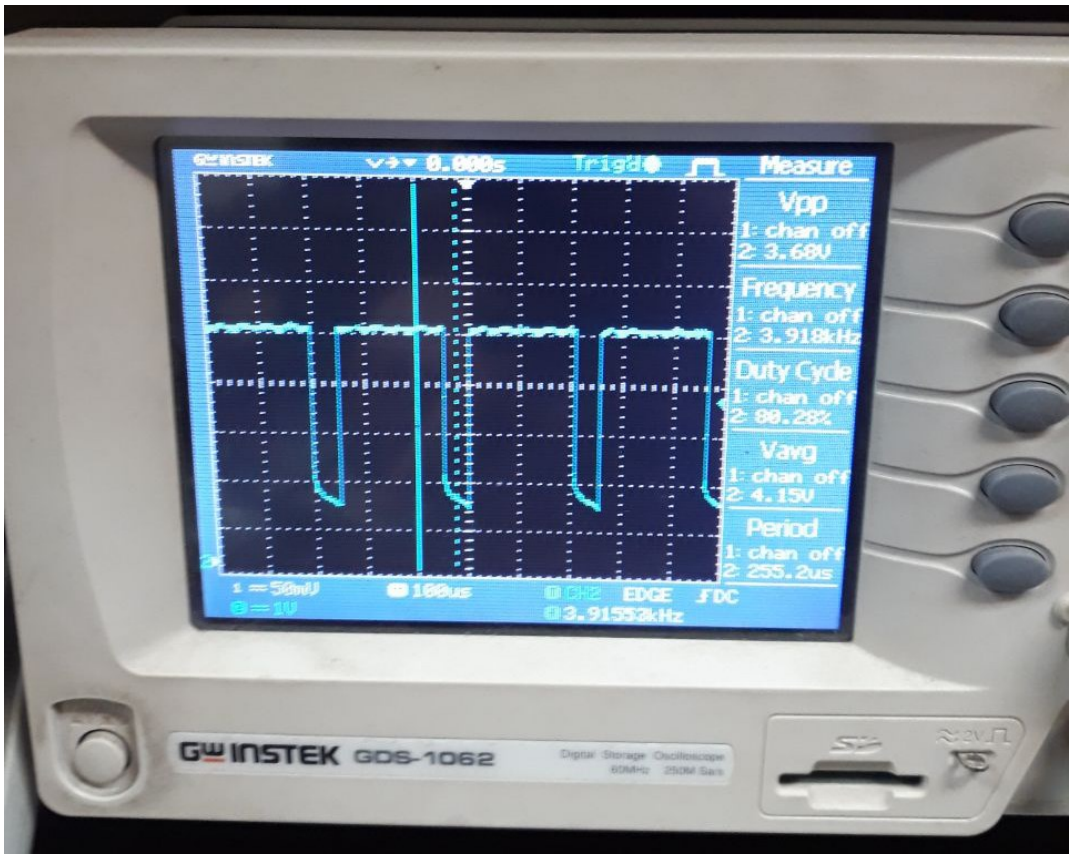
گروه چهارشنبه ساعت ۱۳:۳۰ – ۱۶:۰۰
بازرگانی

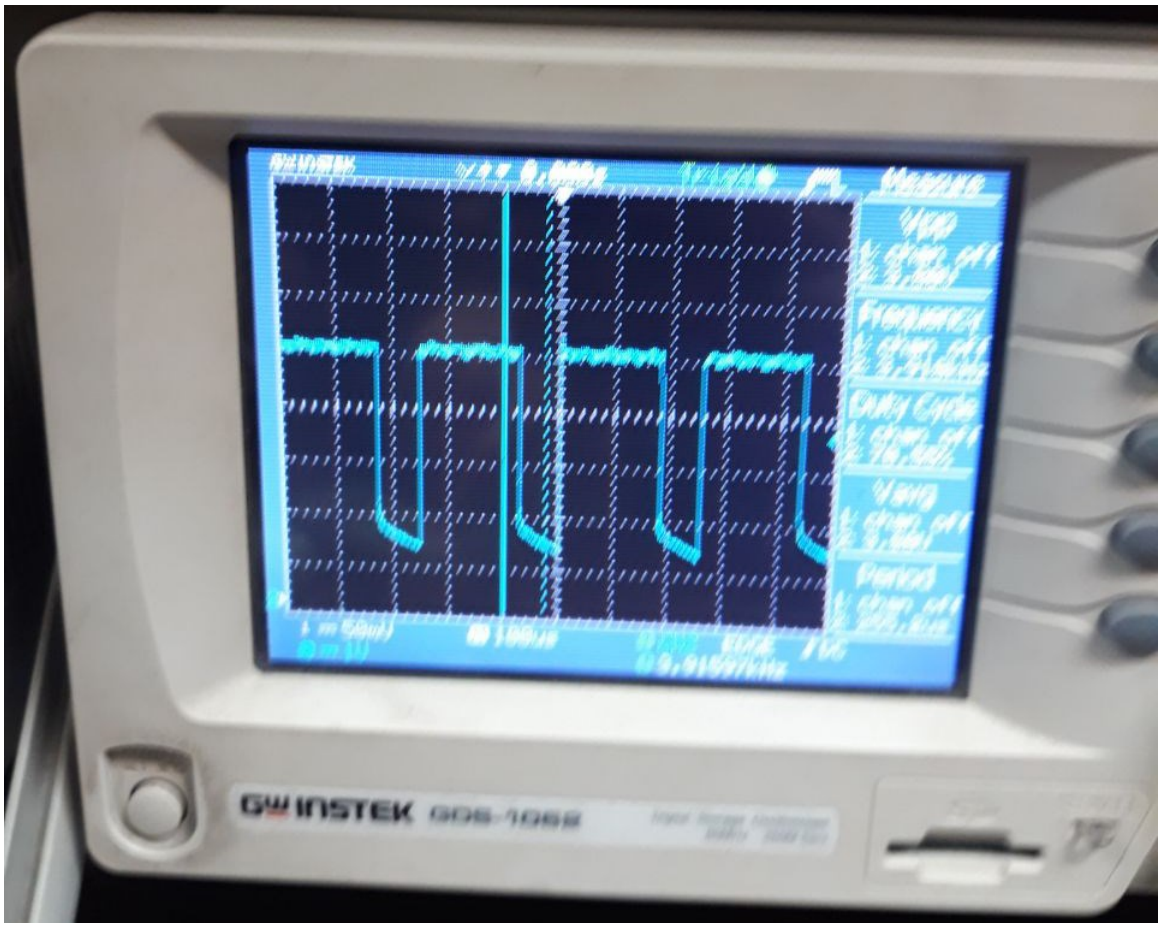
آزمایش ۱۱:

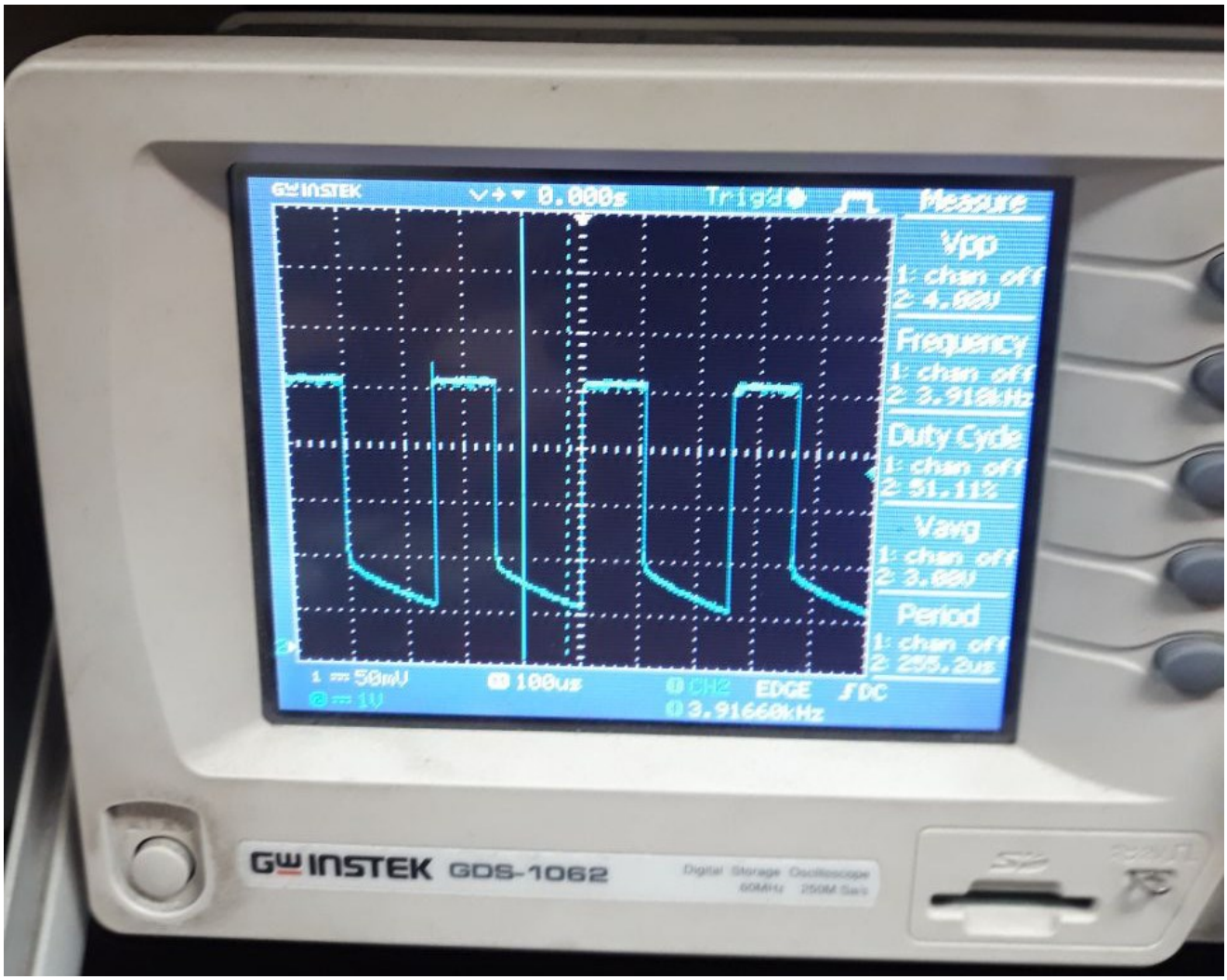
هدف آزمایش: کار با زمان‌سنج/شمارنده ۰ در حالت PWM سریع و فاز صحیح

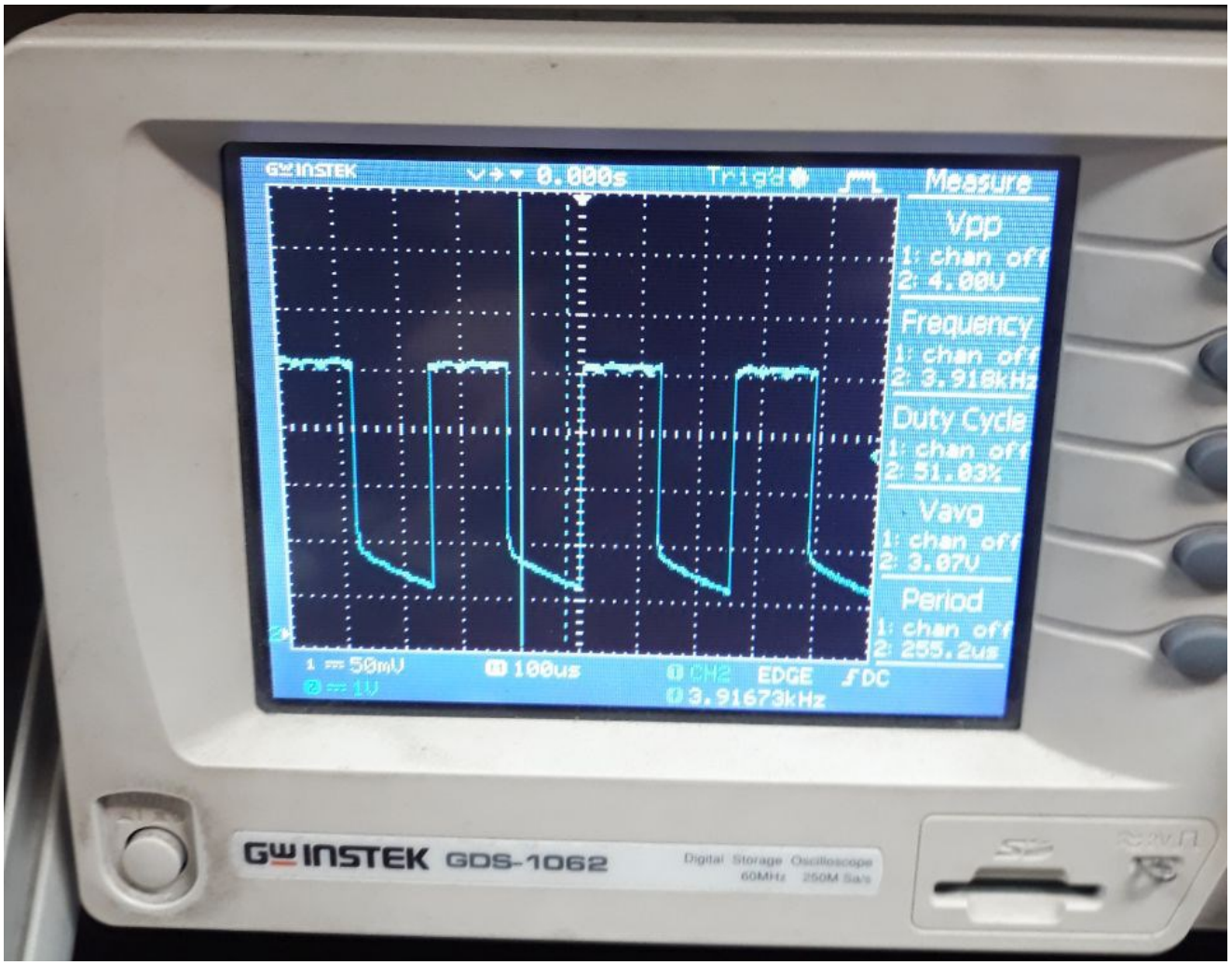
کد: تنظیمات ثبات‌های کنترلی زمان‌سنج/شمارنده ۰ و برنامه کار سیستم در حالت PWM نوشته شده است.

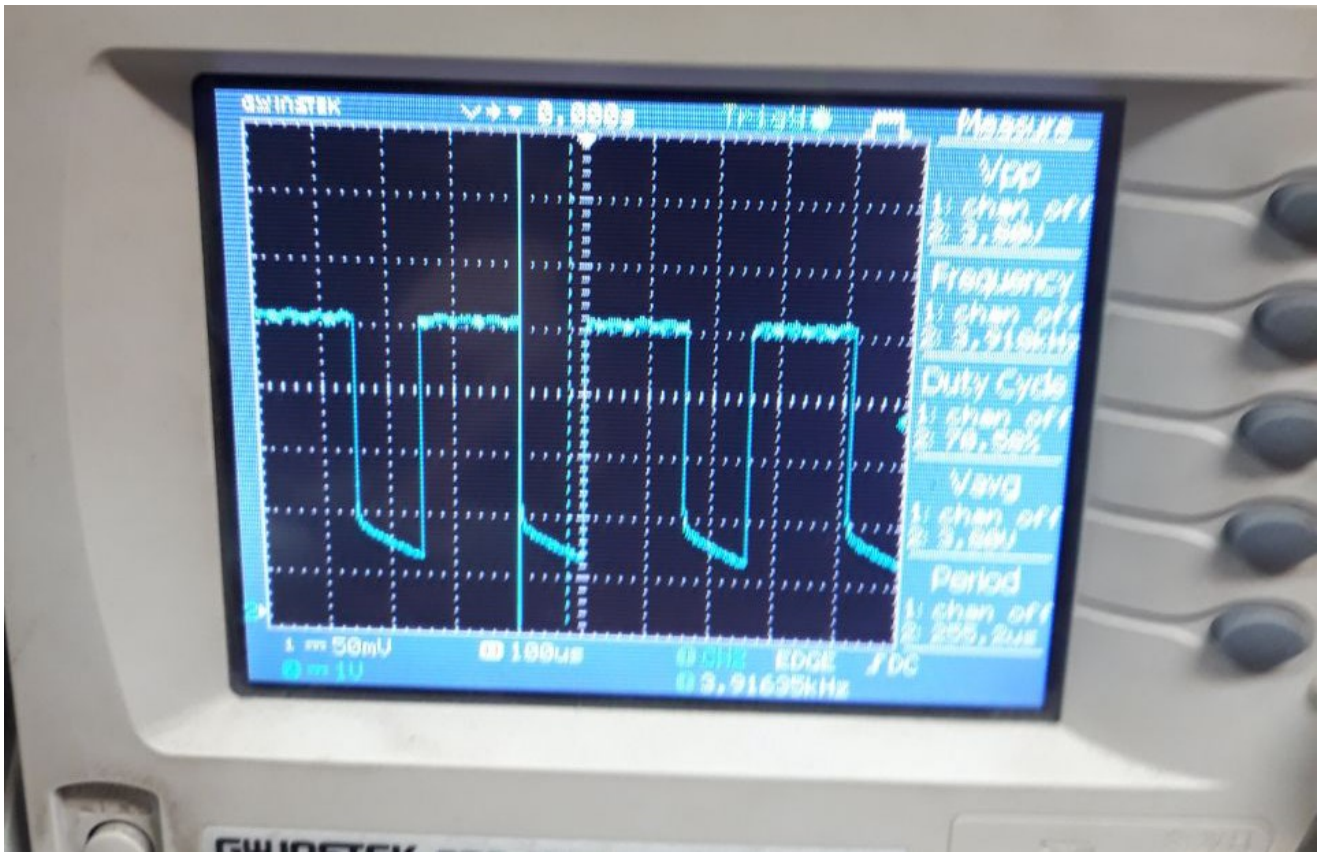


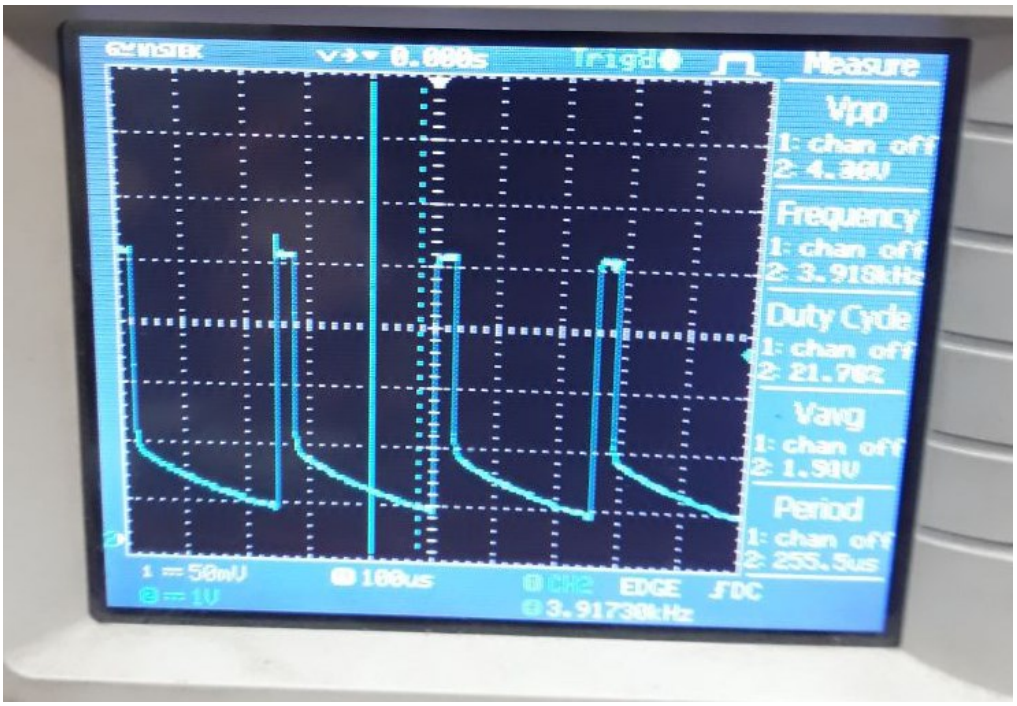

Measure
Vpp
1: chan off
2: 3.68V
Frequency
1: chan off
2: 3.918kHz
Duty Cycle
1: chan off
2: 80.28%
Vavg
1: chan off
2: 4.15V
Period
1: chan off
2: 255.2us
EDGE
GW INSTEK GDS-1062

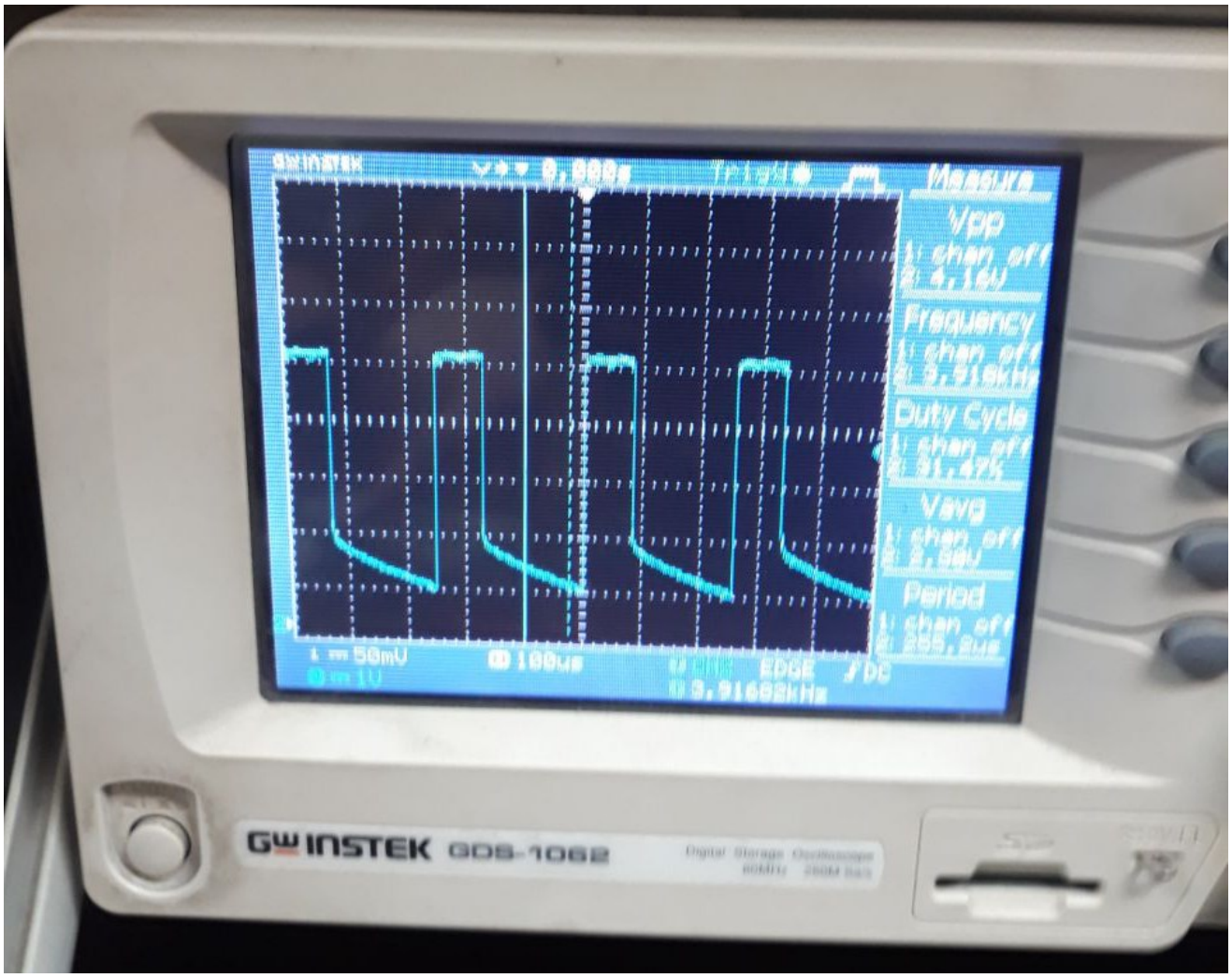

GW INSTEK

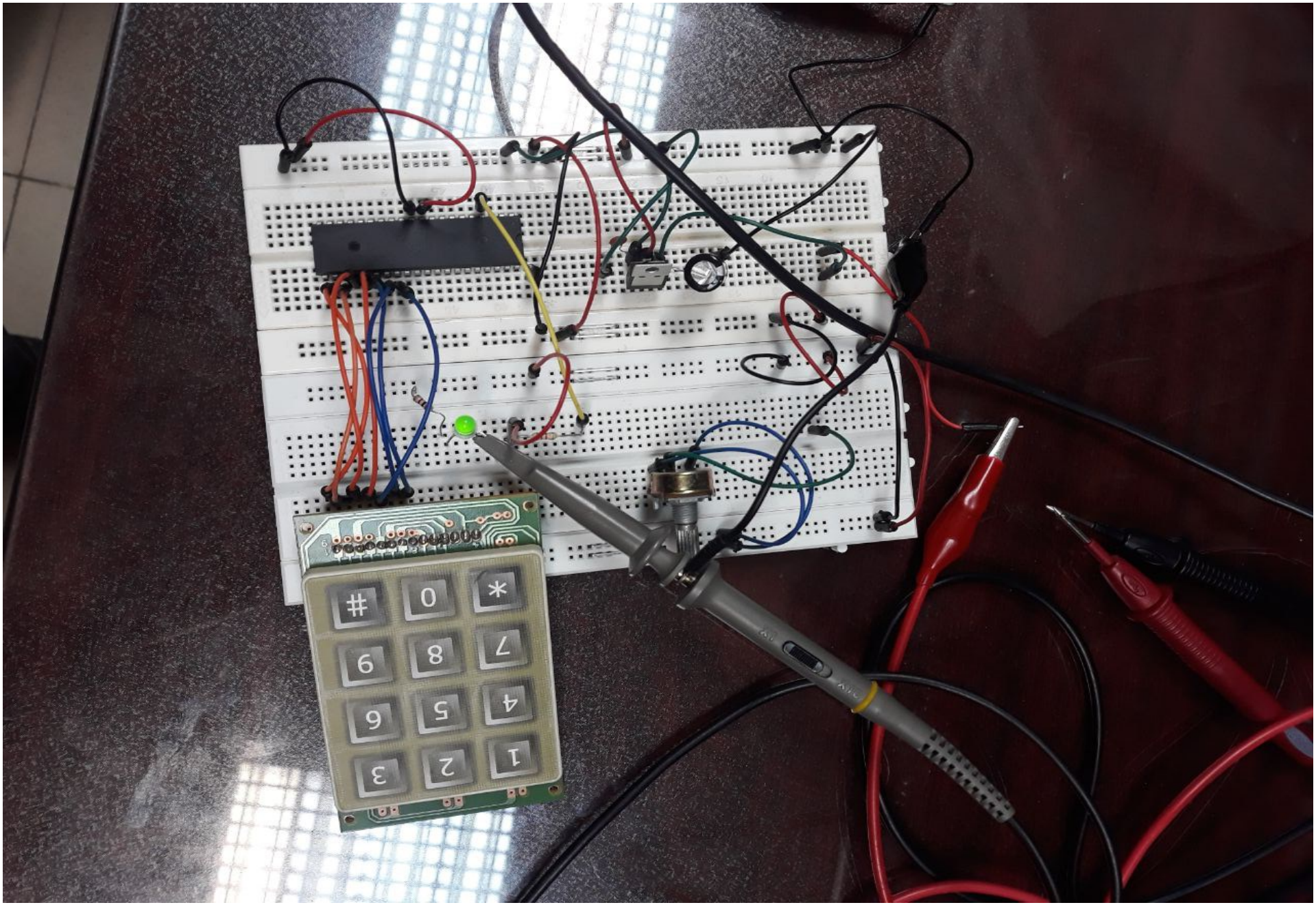

GW INSTEK
0.000s
Trig'd
Measure
Vpp
1: chan off
2 4.88V
Frequency
1: chan off
2 3.916kHz
Duty Cycle
1: chan off
2 51.11%
Vavg
1: chan off
2 3.88V
Period
1: chan off
2 255.2us
1 50mV
2 1V
100us
CH2 EDGE DC
3.91660kHz
GW INSTEK GDS-1062
Digital Storage Oscilloscope
60MHz 250M Sa/s


GW INSTEK
0.000s
Trig'd
Measure
Vpp
1: chan off
2: 4.00V
Frequency
1: chan off
2: 3.918kHz
Duty Cycle
1: chan off
2: 51.03%
Vavg
1: chan off
2: 3.07V
Period
1: chan off
2: 255.2us
1 50mV
2 1V
100us
CH2 EDGE DC
3.91673kHz
GW INSTEK GDS-1062
Digital Storage Oscilloscope
60MHz 250M Sa/s




0.000s
Measure
Vpp
1: chan off
2: 4.30V
Frequency
1: chan off
2: 3.918kHz
Duty Cycle
1: chan off
2: 21.7%
Vavg
1: chan off
2: 1.58V
Period
1: chan off
2: 255.3us
1 = 50mV
100us
EDGE
DC
2 = 1V
3.91730kHz


GW INSTEK GDS-1062


#
0
*
9
8
7
6
5
4
3
2
1

```c
#include<avr/io.h>
#include<avr/interrupt.h>
#include<util/delay.h>
void LCD_Command( unsigned char cmnd )
{
  LCD_Port = (LCD_Port & 0x0F) | (cmnd & 0xF0);
  LCD_Port &= ~ (1<<RS);
  LCD_Port |= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);

  _delay_us(200);

  LCD_Port = (LCD_Port & 0x0F) | (cmnd << 4);
  LCD_Port |= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);
  _delay_ms(2);
}


void LCD_Char( unsigned char data )
{
  LCD_Port = (LCD_Port & 0x0F) | (data & 0xF0);
  LCD_Port |= (1<<RS);
  LCD_Port|= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);

  _delay_us(200);

  LCD_Port = (LCD_Port & 0x0F) | (data << 4);
  LCD_Port |= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);
  _delay_ms(2);
}

void LCD_Init (void)
{
  LCD_Dir = 0xFF;
  _delay_ms(20);

  LCD_Command(0x33);
  LCD_Command(0x32);
  LCD_Command(0x28);
  LCD_Command(0x0c);
  LCD_Command(0x06);
  LCD_Command(0x01);
  _delay_ms(2);
  LCD_Command (0x80);
}
```

```
void LCD_String (char *str)
{
   int i;
   for(i=0;str[i]!=0;i++)
   {
      LCD_Char (str[i]);
   }
}

void LCD_String_xy (char row, char pos, char *str)
{
   if (row == 0 && pos<16)
   LCD_Command((pos & 0x0F)|0x80);
   else if (row == 1 && pos<16)
   LCD_Command((pos & 0x0F)|0xC0);
   LCD_String(str);
}

void LCD_Clear()
{
   LCD_Command (0x01);
   _delay_ms(2);
   LCD_Command (0x80);
}
int keyboard_scan(){
   PORTC |= 0b00001111;
   PORTC &= 0b11111110;
   _delay_ms(100);
   if((PINC & 0b00010000) == 0b00000000){
      OCR0 = 25;
      return 1;
   }
   if((PINC & 0b00100000) == 0b00000000){
      OCR0 = 50;
      return 2;
   }
   if((PINC & 0b01000000) == 0b00000000){
      OCR0 = 75;
      return 3;
   }
   PORTC |= 0b00000001;

   PORTC &= 0b11111101;
   _delay_ms(100);
   if((PINC & 0b00010000) == 0b00000000){
      OCR0 = 100;
      return 4;
   }
   if((PINC & 0b00100000) == 0b00000000){
      OCR0 = 125;
      return 5;
   }
   if((PINC & 0b01000000) == 0b00000000){
```

```
    OCR0 = 150;
    return 6;
  }
  PORTC |= 0b00000010;

  PORTC &= 0b11111011;
  _delay_ms(100);
  if((PINC & 0b00010000) == 0b00000000){
    OCR0 = 175;
    return 7;
  }
  if((PINC & 0b00100000) == 0b00000000){
    OCR0 = 200;
    return 8;
  }
  if((PINC & 0b01000000) == 0b00000000){
    OCR0 = 225;
    return 9;
  }
  PORTC |= 0b00000100;

  PORTC &= 0b11110111;
  _delay_ms(100);
  if((PINC & 0b00100000) == 0b00000000){
    OCR0 = 0;
    return 0;
  }
  PORTC |= 0b00001000;
  return PINB;
}

int main(){
  DDRC |= 0b00001111;
  DDRC &= 0b00001111;
  PORTC |= 0b01111111;
  DDRB = 0b00001000;
  TCCR0 = 0b01101001;
  PORTD = 0;
  DDRA = 0b11111111;
  LCD_Init();
  int key_value;
  while(1)
    keyboard_scan();
  return 0;
}
```