unsigned long t0=0;

volatile unsigned long t1=0;

bool b=false;

void setup() {

// put your setup code here, to run once:

Serial.begin(115200);

pinMode(9,OUTPUT);

analogWrite(9,122);

pinMode(2,INPUT);

attachInterrupt(digitalPinToInterrupt(2),ISR\_measureFreq,RISING);

}

void loop() {

// put your main code here, to run repeatedly:

if (b){

Serial.println((String)"Freq. is: "+ 1E6/(t1-t0) +" Hz.");

t0=t1;

b=false;

}

}

void ISR\_measureFreq(){

t1=micros();

b=true;

}

unsigned long t0=0;

volatile unsigned long t1=0;

bool b=false;

unsigned long t2;

unsigned long t3;

int deltaT=1000;

void setup() {

// put your setup code here, to run once:

Serial.begin(115200);

pinMode(9,OUTPUT);

analogWrite(9,122);

pinMode(2,INPUT);

attachInterrupt(digitalPinToInterrupt(2),ISR\_measureFreq,RISING);

}

void loop() {

// put your main code here, to run repeatedly:

if (b){

t3=millis();

if ((t3-t2)>deltaT){

Serial.println((String)"Freq. is: "+ 1E6/(t1-t0));

t2=t3;

}

t0=t1;

b=false;

}

}

void ISR\_measureFreq(){

t1=micros();

b=true;

}

//Download the library from https://github.com/PaulStoffregen/FreqCount/archive/master.zip

// input signal must be applied to pin 5 of Arduino UNO, 12 of Leonardo or 47 of Mega

#include <FreqCount.h>

void setup() {

Serial.begin(9600);

FreqCount.begin(1000); //counts the number of pulses in 1000 ms.

}

void loop() {

if (FreqCount.available()) {

unsigned long count = FreqCount.read();

Serial.println((String)"Frequency is: "+count+"Hz.");

}

}

// Pulse width measurement

// https://www.arduino.cc/reference/en/language/functions/advanced-io/pulsein/

#define pin 7

unsigned long posDuration=0;

unsigned long negDuration=0;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

pinMode(pin,INPUT);

}

void loop() {

posDuration=pulseIn(pin,HIGH); //reads the positive width in micro seconds

negDuration=pulseIn(pin,LOW); //reads the positive width in micro seconds

Serial.println((String)"Length of positive section:"+posDuration);

Serial.println((String)"Length of negative section:"+negDuration);

Serial.println();

delay(1000);

}

unsigned long t0=0;

unsigned long t1=0;

volatile unsigned long n=0;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

pinMode(13,OUTPUT);

digitalWrite(13,LOW);

pinMode(2,INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(2),LEDblink,FALLING);

}

void loop() {

// put your main code here, to run repeatedly:

Serial.println(n);

}

void LEDblink(){

n++;

for (int i=0;i<3;i++){

digitalWrite(13, HIGH);

for (int j=0;j<500;j++)

delayMicroseconds(1000);

digitalWrite(13, LOW);

for (int k=0;k<500;k++)

delayMicroseconds(1000);

}

}

unsigned long t0=0;

unsigned long t1=0;

volatile unsigned long n=0;

void setup() {

// put your setup code here, to run once:

Serial.begin(9600);

pinMode(13,OUTPUT);

digitalWrite(13,LOW);

pinMode(2,INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(2),LEDblink,LOW);

}

void loop() {

// put your main code here, to run repeatedly:

Serial.println(n);

}

void LEDblink(){

n++;

for (int i=0;i<3;i++){

digitalWrite(13, HIGH);

for (int j=0;j<500;j++)

delayMicroseconds(1000);

digitalWrite(13, LOW);

for (int k=0;k<500;k++)

delayMicroseconds(1000);

}

}

/\*pin 6 of arduino UNO (+ of comparator) to center terminal of potentiometer.

\*pin 7 of arduino (- of comparator) to 3.3 V.

\*/

void setup() {

// put your setup code here, to run once:

ACSR=0x00; //clear bits of ACSR register

ADCSRB=0x00; //clear bits of ADCSRB register

pinMode(LED\_BUILTIN,OUTPUT);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

// when output of comparator is low, i.e., 0, ACSR&0x20 generates 0.

// when output of comparator is high, i.e., 1, ACSR&0x20 generates 32.

if ((ACSR&0x20)==0){

digitalWrite(LED\_BUILTIN,LOW);

}else{

digitalWrite(LED\_BUILTIN,HIGH);

}

}

/\* + terminal of comparator is connected to reference voltage of 1.1 V.

- terminal is connected to pin 7. center terminal of potentiometer

is connected to pin 7, as well.

When input voltage is bigger than 1.1 V, onboard LED is off.

When input voltage is less than 1.1 V, onboard LED is on.

\*/

void setup() {

// put your setup code here, to run once:

ACSR=0b01000000; //makes ACBG=1 which connect the + terminal of the

//comparator to internal reference voltage source.

ADCSRB=0x00; //clear bits of ADCSRB register

pinMode(LED\_BUILTIN,OUTPUT);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

// when output of comparator is low, i.e., 0, ACSR&0x20 generates 0.

// when output of comparator is high, i.e., 1, ACSR&0x20 generates 32.

if ((ACSR&0x20)==0){

digitalWrite(LED\_BUILTIN,HIGH);

}else{

digitalWrite(LED\_BUILTIN,LOW);

}

}

/\* + terminal of comparator is connected to reference voltage of 1.1 V.

- terminal is connected to pin A0 when ADMUX=0 and it is connected to A1

when ADMUX=1.

\*/

void setup() {

// put your setup code here, to run once:

ACSR=0b01000000; //makes ACBG=1 which connect the + terminal of the

//comparator to internal reference voltage source.

ADCSRA=0b00000000; //ADEN=0.

ADCSRB=0b01000000; //ACME=1

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

// when output of comparator is low, i.e., 0, ACSR&0x20 generates 0.

// when output of comparator is high, i.e., 1, ACSR&0x20 generates 32.

ADMUX=0; //select A0 for - terminal of comparator

// ADMUX= 2 selects A2; ADMUX=3 selects A3,...

if ((ACSR&0x20)==0){

Serial.println("voltage of A0 is bigger than 1.1 V.");

}else{

Serial.println("voltage of A0 is less than 1.1 V.");

}

ADMUX=1; //select A1 for - terminal of comparator

if ((ACSR&0x20)==0){

Serial.println("voltage of A1 is bigger than 1.1 V.");

}else{

Serial.println("voltage of A1 is less than 1.1 V.");

}

Serial.println("------------------------------------");

delay(2000);

}

/\* + terminal of comparator is connected to reference voltage of 1.1 V.

- terminal is connected to pin A0. Interrupt is used here process the output

of the output of the comparator.

\*/

volatile int n=0;

void setup() {

// put your setup code here, to run once:

cli();

ACSR=0b01001011; //makes ACBG=1 which connect the + terminal of the

//comparator to internal reference voltage source.

//set interrupt on rising edge

ADCSRA=0b00000000; //ADEN=0.

ADCSRB=0b01000000; //ACME=1.

ADMUX=0; //select A0 for - terminal of comparator.

sei();

pinMode(LED\_BUILTIN,OUTPUT);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

Serial.println(n);

digitalWrite(LED\_BUILTIN,ACSR&0b00100000); //shows the output of the comparator.

//when led is on output of the comparator is high.

//when led is off output of the comparator is low.

}

ISR (ANALOG\_COMP\_vect){

n++;

}

/\* + terminal of comparator is connected to reference voltage of 1.1 V.

- terminal is connected to pin A0. Interrupt is used here process the output

of the output of the comparator.

\*/

volatile int n=0;

void setup() {

// put your setup code here, to run once:

cli();

ACSR=0b01001010; //makes ACBG=1 which connect the + terminal of the

//comparator to internal reference voltage source.

//set interrupt on falling edge

ADCSRA=0b00000000; //ADEN=0.

ADCSRB=0b01000000; //ACME=1.

ADMUX=0; //select A0 for - terminal of comparator.

sei();

pinMode(LED\_BUILTIN,OUTPUT);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

Serial.println(n);

digitalWrite(LED\_BUILTIN,ACSR&0b00100000); //shows the output of the comparator.

//when led is on output of the comparator is high.

//when led is off output of the comparator is low.

}

ISR (ANALOG\_COMP\_vect){

n++;

}

/\* + terminal of comparator is connected to reference voltage of 1.1 V.

- terminal is connected to pin A0. Interrupt is used here process the output

of the output of the comparator.

\*/

volatile int n=0;

void setup() {

// put your setup code here, to run once:

cli();

ACSR=0b01001000; //makes ACBG=1 which connect the + terminal of the

//comparator to internal reference voltage source.

//set interrupt on toggle

ADCSRA=0b00000000; //ADEN=0.

ADCSRB=0b01000000; //ACME=1.

ADMUX=0; //select A0 for - terminal of comparator.

sei();

pinMode(LED\_BUILTIN,OUTPUT);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

Serial.println(n);

digitalWrite(LED\_BUILTIN,ACSR&0b00100000); //shows the output of the comparator.

//when led is on output of the comparator is high.

//when led is off output of the comparator is low.

}

ISR (ANALOG\_COMP\_vect){

n++;

}

volatile bool state=false;

void setup() {

// put your setup code here, to run once:

PCICR|=0b00000001; // PCINT for pins 8, 9, 10, 11, 12 and 13 is enabled.

PCMSK0|=0b00011000; // Select PCINT3 and PCINT4 = pins 11 and 12 of Arduino UNO

pinMode(LED\_BUILTIN,OUTPUT);

pinMode(11,INPUT\_PULLUP);

pinMode(12,INPUT\_PULLUP);

}

void loop() {

// put your main code here, to run repeatedly:

}

ISR (PCINT0\_vect){

state=!state;

digitalWrite(LED\_BUILTIN, state);

}

void setup() {

// put your setup code here, to run once:

PCICR|=0b00000001; // PCINT for pins 8, 9, 10, 11, 12 and 13 is enabled.

PCMSK0|=0b00011000; // Select PCINT3 and PCINT4, pins 11 and 12 of Arduino UNO

pinMode(LED\_BUILTIN,OUTPUT);

pinMode(11,INPUT\_PULLUP);

pinMode(12,INPUT\_PULLUP);

}

void loop() {

// put your main code here, to run repeatedly:

}

ISR (PCINT0\_vect){

if (digitalRead(11)==LOW)

digitalWrite(LED\_BUILTIN, LOW);

if (digitalRead(12)==LOW)

digitalWrite(LED\_BUILTIN, HIGH);

}