Файл metal\_detecter.ino

1 //------------НАСТРОЙКИ-------------

2 // 0 - в стиле счётчика Гейгера, 1 - верещание

3 #define sound 0

4 // если нет переключателя режимов, то по умолчанию стоит:

5 // 1 - статический, 0 - динамический

6 #define default\_mode 1

7 // 0 - нет регулировки, чувствительности, 1 - есть

8 #define sens\_change 1

9 // ручная установка чувствительности

10 float SENSITIVITY = 1000.0;

11 //------------НАСТРОЙКИ-------------

12

13 #define soundPin 12

14 #define resetPinGND 2

15 #define resetPin 3

16 #define switchPinGND 9

17 #define switchPin 10

18 #define sens 6

19 #define SET(x,y) (x |=(1<<y))

20 #define CLR(x,y) (x &= (~(1<<y)))

21 #define CHK(x,y) (x & (1<<y))

22 #define TOG(x,y) (x^=(1<<y))

23

24 float koef = 0.5;

25 unsigned long t0 = 0;

26 unsigned long last\_step;

27 int t = 0;

28 unsigned char tflag = 0;

29 float clf;

30 int v0 = 0;

31 float f = 0;

32 unsigned int FTW = 0;

33 unsigned int PCW = 0;

34 unsigned long timer = 0;

35

36 // Срабатывание каждые 1000 импульсов с генератора

37 SIGNAL(TIMER1\_COMPA\_vect)

38 {

39 OCR1A += 1000;

40 t = micros() - t0;

41 t0 += t;

42 tflag = 1;

43 }

44

45 void setup() {

46 Serial.begin(9600);

47 pinMode(soundPin, OUTPUT);

48 pinMode(resetPin, INPUT\_PULLUP);

49 pinMode(resetPinGND, OUTPUT);

50 digitalWrite(resetPinGND, 0);

51 pinMode(switchPin, INPUT\_PULLUP);

52 pinMode(switchPinGND, OUTPUT);

53 digitalWrite(switchPinGND, 0);

54

55 //Установка счётчика для счёта 5 пина

56 TCCR1A = 0;

57 TCCR1B = 0x07;

58 Serial.print("$");

59 Serial.print(f);

60 Serial.println(";");

61 SET(TIMSK1, OCF1A);

62 sei();

63 }

64

65 // Функция модуля

66 float absf(float f)

67 {

68 if (f < 0.0)

69 return -f;

70 else

71 return f;

72 }

73

74 void loop() {

75 if (tflag) {

76 Serial.print("$");

77 Serial.print(f);

78 Serial.println(";");

79 // если кнопка режимов не нажата

80 if (digitalRead(switchPin) == default\_mode) {

81 // если нажата кнопка сброса частоты

82 if (!digitalRead(resetPin))

83 v0 = t;

84 // нажат другой режим

85 } else {

86 // каждые 300 мсек

87 if (millis() - last\_step > 300) {

88 last\_step = millis();

89 // плавно сбрасываем частоту

90 v0 = v0 \* koef + (1 - koef) \* t;

91 }

92 }

93 // фильтруем сигнал

94 f = f \* 0.85 + absf(t - v0) \* 0.15;

95 // если разрешено внешнее изменение чувствительности

96 if (sens\_change)

97 // принять с аналогового пина, преобразовать

98 SENSITIVITY = map(analogRead(sens), 0, 1023,500,2000);

99 // конвертация частоты в писк

100 clf = f \* SENSITIVITY;

101 if (clf > 10000)

102 clf = 10000;

103 FTW = clf;

104 // переустанавливаем флаг

105 tflag = 0;

106 }

107

108 if (sound) {

109 if (millis() > timer) {

110 timer += 10;

111 PCW += FTW;

112 if (PCW & 0x8000)

113 {

114 digitalWrite(soundPin, HIGH);

115 PCW &= 0x7fff;

116 }

117 else

118 digitalWrite(soundPin, LOW);

119 }

120 } else {

121 if (millis() > timer) {

122 timer += 10;

123 PCW += FTW;

124 if (PCW & 0x8000)

125 {

126 PCW &= 0x7fff;

127 tone(soundPin, 3000, 30);

128 delay(20);

129 noTone(soundPin);

130 }

131 else

132 noTone(soundPin);

133 }

134 }

135 }