

sketch_jun23b | Arduino 1.8.13

sketch_jun23b

```
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly
8
9 }
```

1 Arduino Nano, ATmega328P (Old Bootloader) on /dev/cu.wchusbserial1430

Blink | Arduino 1.8.13

Blink

```
1 /*
2  * Blink
3  *
4  * Turns an LED on for one second, then off for one second, repeatedly.
5  *
6  * Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
7  * it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
8  * the correct LED pin independent of which board is used.
9  * If you want to know what pin the on-board LED is connected to on your Arduino
10  * model, check the Technical Specs of your board at:
11  * https://www.arduino.cc/en/Main/Products
12  *
13  * modified 8 May 2014
14  * by Scott Fitzgerald
15  * modified 2 Sep 2016
16  * by Arturo Guadalupi
17  * modified 8 Sep 2016
18  * by Colby Newman
19  *
20  * This example code is in the public domain.
21  *
22  * http://www.arduino.cc/en/Tutorial/Blink
23  */
24
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
34   delay(1000); // wait for a second
35   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
36   delay(1000); // wait for a second
37 }
```

1 Arduino Nano, ATmega328P (Old Bootloader) on /dev/cu.wchusbserial1430


```
sketch_jun23b | Arduino 1.8.13

1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeated
8
9 }
```

```
Blink | Arduino 1.8.13

1 /*
2  Blink
3
4  Turns an LED on for one second, then off for one second, repeating.
5
6  Most Arduinos have an on-board LED you can control. On the Uno
7  it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN
8  is the correct LED pin independent of which board is used.
9  If you want to know what pin the on-board LED is connected to on
10 your board, check the Technical Specs of your board at:
11 https://www.arduino.cc/en/Main/Products
12
13 modified 8 May 2014
14 by Scott Fitzgerald
15 modified 2 Sep 2016
16 by Arturo Guadalupi
17 modified 8 Sep 2016
18 by Colby Newman
19
20 This example code is in the public domain.
21
22 http://www.arduino.cc/en/Tutorial/Blink
23 */
24
25 // the setup function runs once when you press reset or power
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the
34   delay(1000); // wait for a second
35   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the
36   delay(1000); // wait for a second
37 }
```

```
linedancer_node | Arduino 1.8.13

linedancer_node HystFilter.cpp HystFilter.h system.h utils
51 const uint8_t dipPins[] = {pin_addr_1, pin_addr_2, pin_addr_3, pin_addr_4}; // DIP Switch Pins
52 uint8_t boardAddress; // board address from DIP switch
53
54 // variables for distance signals and filtering
55 long distA = 0; // raw distance
56 long distB = 0;
57 long distA_filtered = 0; // filtered distance signal (two stage exponential filter)
58 long distB_filtered = 0;
59 long oldDistA = 0; // distance value from previous loop
60 long oldDistB = 0;
61 long aHyst = 0; // hysteresis from filtered signal
62 long bHyst = 0;
63
64 // variables for step calculation
65 uint16_t closeness = 60; // 55 cm as detection distance for wall side / 60 for no wall
66 long deviation = round(closeness * 0.6); // how large is the change in distance data from one loop to the other - larger
67 long thresh = 4; // threshold for detection of change
68 uint16_t dipSteps = 1650; // maximum steps the stepper can travel to dip into the water
69 uint16_t minSteps = 750; // minimum steps the stepper should travel to dip into the water
70
71 // variables for homing purposes
72 const long initSteps = 600; // steps to travel to the start position after homing
73 const long maxHomingDistanceInSteps = 2000;
74 const long homingSpeedInSteps = 50;
75 const uint8_t directionTowardHome = -1;
76
77 // variable for several states and eventualities while program is executed
78 boolean homed = false;
79 boolean allowMotionA = false;
80 boolean allowMotionB = false;
81 boolean movedA = false;
82 boolean movedB = false;
83
84 // several debug sets
85 boolean debug = true; // view single program steps and high level operation = setup -> homing -> distance sensing
86 boolean debug2 = false; // view only distance data in serial monitor
87 boolean debugCurves = false; // view filtered distances and hysteresis in the signal plotter
88
89 boolean RS_Comms = false; // set true if RS485 communication is used -> implement later
90
91 // use to only run with one stepper motor
92 boolean mot_a = true;
93 boolean mot_b = false;
94
95 // individual timers and intervals for sensor reading
96 elapsedMillis sensorATimer;
97 elapsedMillis sensorBTimer;
98 uint8_t sensorAInterval = 40; // interval to read the sensor in (40 equals 25hz refresh rate)
99 uint8_t sensorBInterval = 52; // 52 was ok
100 uint8_t switchInterval = 10; // microsecond delay in sensor read function
101
102 void setup() {
103   Serial.begin(115200);
104   if (debug) {
105     Serial.println();
106   }
107 }
108
109 Done Saving.
110 at java.lang.Thread.dumpStack(Thread.java:1336)
111 java.lang.Exception: Stack trace
112 at java.lang.Thread.dumpStack(Thread.java:1336)
113
114 304 Arduino Nano, ATmega328P (Old Bootloader) on /dev/cu.wchusbserial1430
```