

Γλώσσα προγραμματισμού Arduino

σχόλια	Κώδικας - δομές	
Τύπος μεταβλητής	<p>byte Byte stores an 8-bit numerical value without decimal points. They have a range of 0-255.</p> <pre>byte someVariable = 180; // declares 'someVariable' // as a byte type</pre> <p>int Integers are the primary datatype for storage of numbers without decimal points and store a 16-bit value with a range of 32,767 to -32,768.</p> <pre>int someVariable = 1500; // declares 'someVariable' // as an integer type</pre> <p>Note: Integer variables will roll over if forced past their maximum or minimum values by an assignment or comparison. For example, if $x = 32767$ and a subsequent statement adds 1 to x, $x = x + 1$ or $x++$, x will then rollover and equal -32,768.</p> <p>long Extended size datatype for long integers, without decimal points, stored in a 32-bit value with a range of 2,147,483,647 to -2,147,483,648.</p> <pre>long someVariable = 90000; // declares 'someVariable' // as a long type</pre> <p>float A datatype for floating-point numbers, or numbers that have a decimal point. Floating-point numbers have greater resolution than integers and are stored as a 32-bit value with a range of 3.4028235E+38 to -3.4028235E+38.</p> <pre>float someVariable = 3.14; // declares 'someVariable' // as a floating-point type</pre>	
	πίνακας	<pre>int myArray[5]; // declares integer array w/ 6 positions myArray[3] = 10; // assigns the 4th index the value 10</pre> <pre>int ledPin = 10; // LED on pin 10 byte flicker[] = {180, 30, 255, 200, 10, 90, 150, 60}; // above array of 8</pre>
	Ανάθεση -εκχώρηση τιμής	<pre>y = y + 3; x = x - 7; i = j * 6; r = r / 5;</pre>
	Αύξηση μείωση τιμής μεταβλητής	<pre>x ++ // same as x = x + 1, or increments x by +1 x -- // same as x = x - 1, or decrements x by -1 x += y // same as x = x + y, or increments x by +y x -= y // same as x = x - y, or decrements x by -y x *= y // same as x = x * y, or multiplies x by y x /= y // same as x = x / y, or divides x by y</pre>
Συγκριτικοί τελεστές	<pre>x == y // x is equal to y x != y // x is not equal to y x < y // x is less than y x > y // x is greater than y x <= y // x is less than or equal to y x >= y // x is greater than or equal to y</pre>	
Λογικοί τελεστές	<p>Logical AND:</p> <pre>if (x > 0 && x < 5) // true only if both // expressions are true</pre>	
	<p>Logical OR:</p> <pre>if (x > 0 y > 0) // true if either // expression is true</pre>	
	<p>Logical NOT:</p> <pre>if (!x > 0) // true only if // expression is false</pre>	
Δήλωση θύρας	<pre>int ledPin = 10; // LED on pin 10</pre>	
Δήλωση -είσοδος έξοδος	<pre>pinMode(ledPin, OUTPUT); // sets OUTPUT pin</pre>	

Γλώσσα προγραμματισμού Arduino

<p>Πρώτα δηλώνω τις μεταβλητές και στη συνέχεια αν είναι είσοδος ή έξοδος</p>	<pre>int led = 13; // connect LED to pin 13 int pin = 7; // connect pushbutton to pin 7 int value = 0; // variable to store the read value void setup() { pinMode(led, OUTPUT); // sets pin 13 as output pinMode(pin, INPUT); // sets pin 7 as input } int ledPin = 10; int potPin = A4; int potValue = 0; // μεταβλητή για την τιμή που θα διαβάσω από το ποτενσιόμετρο void setup() { pinMode(ledPin, OUTPUT); pinMode(potPin, INPUT); } int ledRed = 11; int ledOrange = 10; int ledGreen = 9; void setup() { pinMode(ledRed, OUTPUT); pinMode(ledOrange, OUTPUT); pinMode(ledGreen, OUTPUT); }</pre>
<p>Δήλωση πίνακα</p>	<pre>int ledPin[8]; // δημιουργώ πίνακα 8 θέσεων int i; void setup() { for (i=0;i<8;i++) { ledPin[i] = i+6; // βάζω ως τιμές τα Pins από 6 έως 13 pinMode(ledPin[i], OUTPUT); } }</pre>
<p>Δομή επανάληψης</p>	<pre>for (i=0;i<=7;i++) { digitalWrite(ledPin[i], HIGH); delay(200); digitalWrite(ledPin[i], LOW); }</pre>
	<pre>value = random(100, 200); // sets 'value' to a random // number between 100-200</pre>
	<pre>value = max(value, 100); // sets 'value' to the larger of // 'value' or 100, ensuring that // it is at least 100.</pre>
	<pre>value = min(value, 100); // sets 'value' to the smaller of // 'value' or 100, ensuring that // it never gets above 100.</pre>
<p>Δομή επανάληψης</p>	<pre>for (int i=0; i<20; i++) // declares i, tests if less { // than 20, increments i by 1 digitalWrite(13, HIGH); // turns pin 13 on delay(250); // pauses for 1/4 second digitalWrite(13, LOW); // turns pin 13 off delay(250); // pauses for 1/4 second }</pre>
<p>Δήλωση πίνακα</p>	<pre>byte flicker[] = {180, 30, 255, 200, 10, 90, 150, 60};</pre>
<p>προσπέλαση πίνακα</p>	<pre>for(int i=0; i<7; i++) // loop equals number { // of values in array analogWrite(ledPin, flicker[i]); // write index value delay(200); // pause 200ms }</pre>

Γλώσσα προγραμματισμού Arduino

Απλή δομή επιλογής	<pre>if (b == TRUE); { doSomething; }</pre>	Αισθητήρα //Σταθερές const int photoresistor = A0; const int ledpin=9; //Μεταβλητές int value; void setup(){ pinMode(ledpin, OUTPUT); //pinMode(photoresistor, INPUT); Δεν χρειάζεται! Serial.begin(9600); } void loop(){ value = analogRead(photoresistor); if (value > 25){ digitalWrite(ledpin, LOW); } else{ digitalWrite(ledpin, HIGH); } //Ελεγχος Serial.print("Value:"); Serial.println(value); delay(1000); //Εκτύπωση ανα 1sec... }
Σύνθετη δομή επιλογής	<pre>if (inputPin == HIGH) { doThingA; } else { doThingB; }</pre>	
Πολλαπλή δομή επιλογής	<pre>if (inputPin < 500) { doThingA; } else if (inputPin >= 1000) { doThingB; } else { doThingC; }</pre>	
Δομή επανάληψης όσο int i=1; while (i <=10) { digitalWrite (led1, HIGH); delay, 200; digitalWrite (led1, LOW); delay, 200; i++; }	<pre>While (someVariable < 200) // tests if less than 200 { doSomething; // executes enclosed statements someVariable++; // increments variable by 1 }</pre>	
Δομή επανάληψης Αρχή.... Μέχρις ότου do { digitalWrite (led1, HIGH); i++; } while(i<10);	<pre>do { x = readSensors(); // assigns the value of // readSensors() to x delay(50); // pauses 50 milliseconds } while (x < 100); // loops if x is less than 100</pre>	
Φόρτωση βιβλιοθήκης κινητήρα	<pre>#include <Servo.h> int sPin = 10; int potPin = A1; // το pin του ποτενσιόμετρου Servo s; int potVal = 0; // η τιμή του ποτενσιόμετρου int sVal = 0; void setup() { s.attach(sPin); pinMode(potPin, INPUT); }</pre>	

Συνάρτηση: πριν την void setup. Void nmpako {entoles}. Η κλήση γίνεται μέσα στην void loop nmpako();

Σχόλια	// όταν θέλω να γράψω κάτι μέσα στο κώδικά μου (μια γραμμή)
	/* όταν θέλω να γράψω κάτι μέσα στο κώδικά μου πολλές γραμμές, όταν θέλω να γράψω κάτι μέσα στο κώδικά μου πολλές γραμμές, όταν θέλω να γράψω κάτι μέσα στο κώδικά μου πολλές γραμμές*/

Γλώσσα προγραμματισμού Arduino

Χρήση σταθερών και μεταβλητών Ή μη χρήση σταθερών και μεταβλητών	Const Int ledRed = 6; Const int ledGreen = 5;	
	METABΛHTEΣ	ΧΩΡΙΣ ΜΕΤΑΒΛΗΤΕΣ
	PinMode (ledRed, OUTPUT); PinMode (ledGREEN, OUTPUT);	PinMode (5, OUTPUT); PinMode (6, OUTPUT);
	Int redPin = 11; intgreenPin = 10; int bluePin = 9; int l;	
	METABΛHTEΣ	ΧΩΡΙΣ ΜΕΤΑΒΛΗΤΕΣ
	pinMode (redPin, output); pinMode (greenPin, output); pinMode (bluePin, output);	pinMode (11, output); pinMode (10, output); pinMode (9, output);

Γεννήτρια `i=random(1.100);`

Δομή επανάληψης for (i=0;i<=7;i++) Για i από 0 μέχρι 7 με βήμα 1 (i=i+1)	<pre> for (i=0;i<=7;i++) { digitalWrite(ledPin[i], HIGH); delay(200); digitalWrite(ledPin[i], LOW); } </pre>	
	i=1	Αυτό σημαίνει ότι η τιμή του i θα είναι 1, αρχικοποίηση του i.
	i<=7;	Αυτό σημαίνει ότι μέχρι το i είναι μικρότερο ίσο από το 7 θα ισχύει η συνθήκη και θα εκτελούντε οι εντολές εντός της δομής επανάληψης, <pre> digitalWrite(ledPin[i], HIGH); delay(200); digitalWrite(ledPin[i], LOW); </pre>
	i++	Μετα από την τελευταία εντολή το i θα αυξηθεί κατά 1 αυτό γίνεται μετά την τελευταία εντολή και πριν το } Θα γίνει: <pre> 0<=7 0+1=1 <=7 1+1=2<=7 2+1=3<=7 3+1=4<=7 4+1=5<=7 5+1=6<=7 6+1=7<=7 </pre> <div style="display: flex; align-items: center;"> <div style="font-size: 2em; margin-right: 10px;">}</div> <div>8 φορές</div> </div> <p>7+1=8>7 άρα δεν ισχύει η συνθήκη και βγαίνει από την δομή επανάληψης.</p>

void setup() { Serial.begin(9600); } και

void loop() {Serial.println("to mynima einai"); delay (500); Serial.println("Hello World"); }

**Καλό είναι αυτό τον μίνι οδηγό να το έχετε κοντά σας τουλάχιστον στην αρχή!
 Όπως καλό θα ήταν να κρατάτε σημειώσεις όπως το καταλαβαίνετε εσείς!**