

# ENGINE



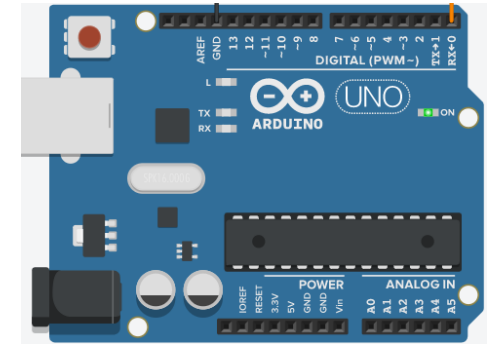
TEACHING ONLINE ELECTRONICS, MICROCONTROLLERS AND PROGRAMMING  
IN HIGHER EDUCATION

---

Module\_1-3. *Communication and ADC*

# Περιεχόμενα

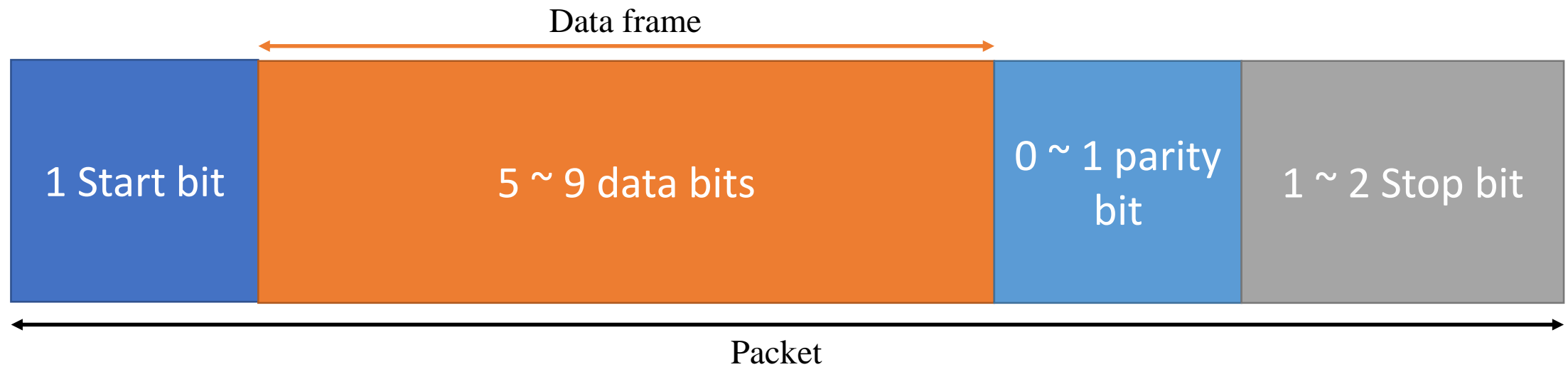
- Σειριακή επικοινωνία
- Μετατροπέας αναλογικού σήματος σε ψηφιακό
- Εντολές προγραμματισμού για το Arduino Uno
- Παράδειγμα



### 3. Communication and ADC

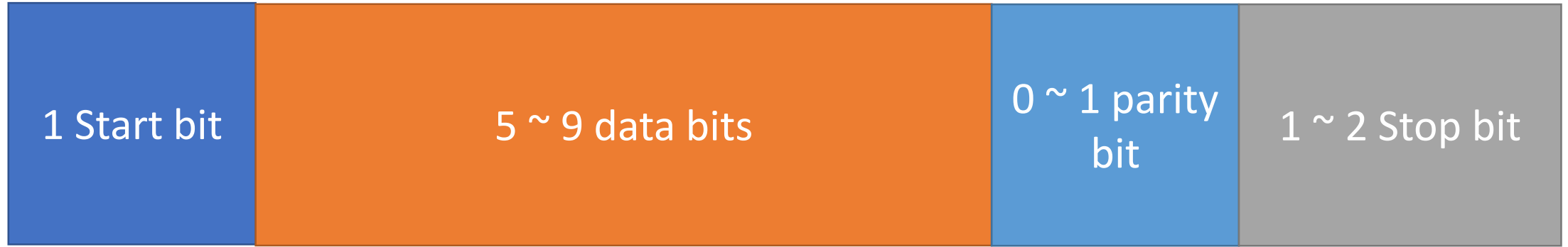
#### Σειριακή επικοινωνία

Η σειριακή επικοινωνία είναι ένας ευρέως διαδεδομένος τρόπος να ανταλλάξουν δεδομένα δύο συσκευές.



1. <https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html>
2. <https://www.circuitbasics.com/basics-uart-communication/>

### 3. *Communication and ADC* *Σειριακή επικοινωνία*



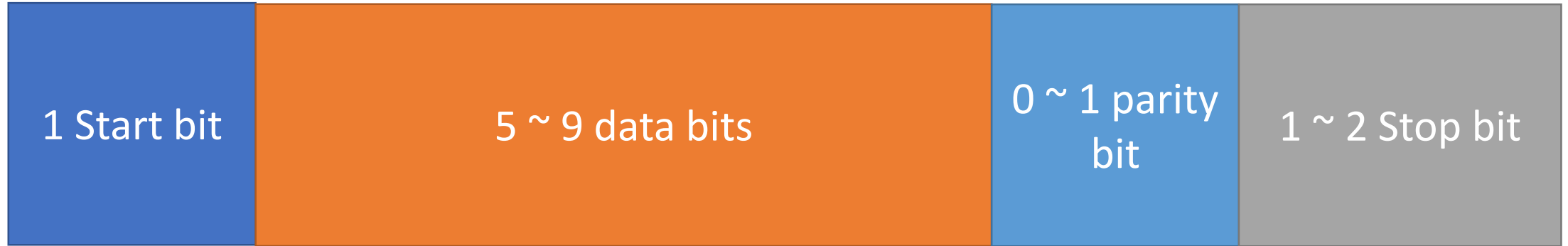
#### Πλεονεκτήματα

- Επικοινωνία με μόλις δύο καλώδια: Rx και Tx
- Έλεγχος σφαλμάτων: parity bit
- Ασύγχρονη επικοινωνία

1. <https://www.analog.com/en/analog-dialogue/articles/uart-a-hardware-communication-protocol.html>

2. <https://www.circuitbasics.com/basics-uart-communication/>

### 3. *Communication and ADC* *Σειριακή επικοινωνία*

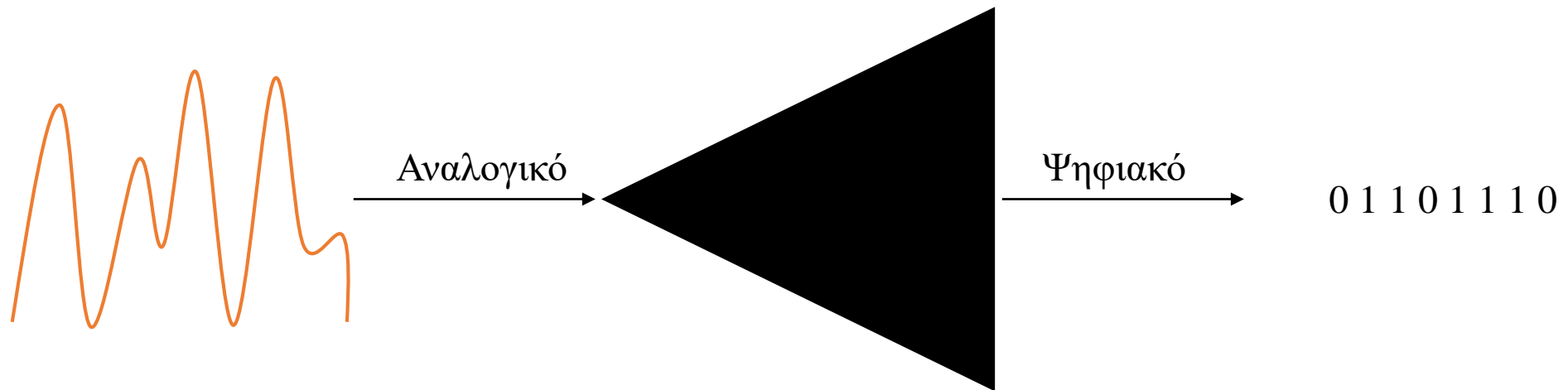


#### Μειονεκτήματα

- Το μέγιστο εύρος δεδομένων ανά πακέτο είναι μόλις 9 bits
- Δεν υποστηρίζει πολλαπλές συσκευές

### 3. *Communication and ADC* *Analog to Digital Converter*

Ο Μετατροπέας αναλογικού σήματος σε ψηφιακό, παίρνει ένα στιγμιότυπο του άγνωστου αναλογικού σήματος και το μετατρέπει σε σειρά δυαδικών ψηφίων.



1. <https://www.electronics-tutorials.ws/combination/analogue-to-digital-converter.html>
2. <http://www.onmyphd.com/?p=analog.digital.converter>
3. <https://dewesoft.com/daq/types-of-adc-converters>

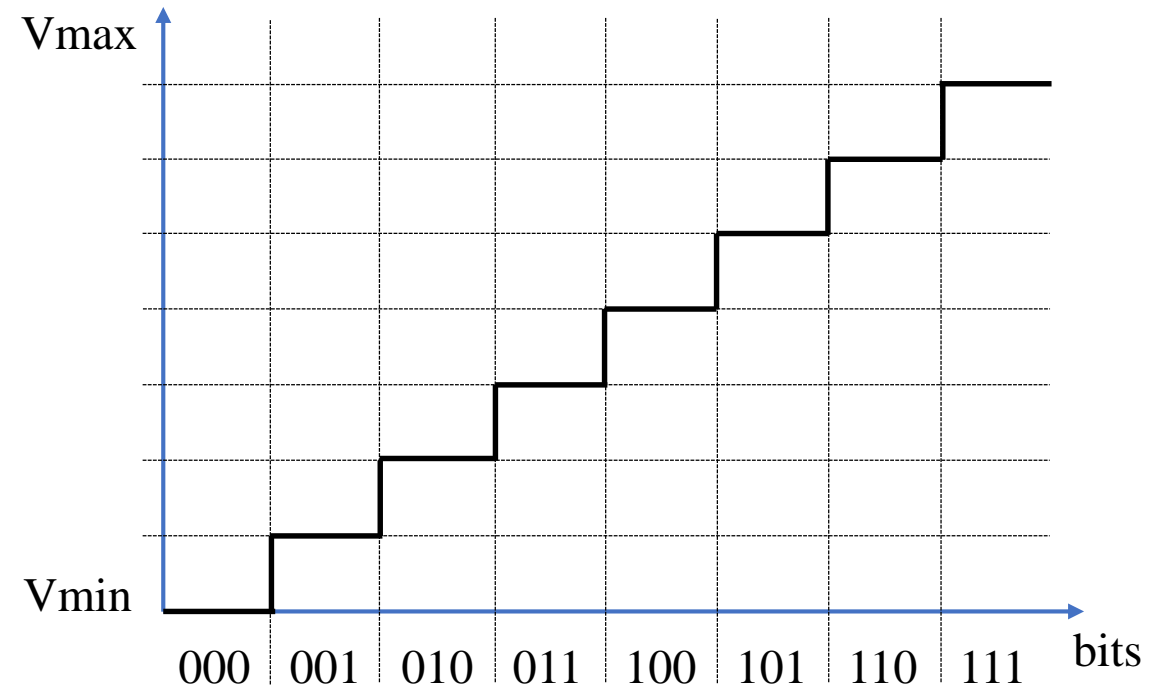
### 3. Communication and ADC Analog to Digital Converter

Ένας μετατροπέας 3 bit, μπορεί να δώσει  $2^3-1 = 7$  συνδυασμούς, δηλαδή 8 διαφορετικές καταστάσεις (0 ~ 7). Με άλλα λόγια, ένα εύρος τάσης (span) μπορεί να το χωρίσει σε 8 καταστάσεις και να αντιστοιχήσει κάθε κατάσταση με μία σειρά από 3 bits.

Η ελάχιστη ποσότητα τάσης που μπορεί να μετρήσει ένας μετατροπέας και να την αντιστοιχήσει στο 1<sup>ο</sup> bit είναι

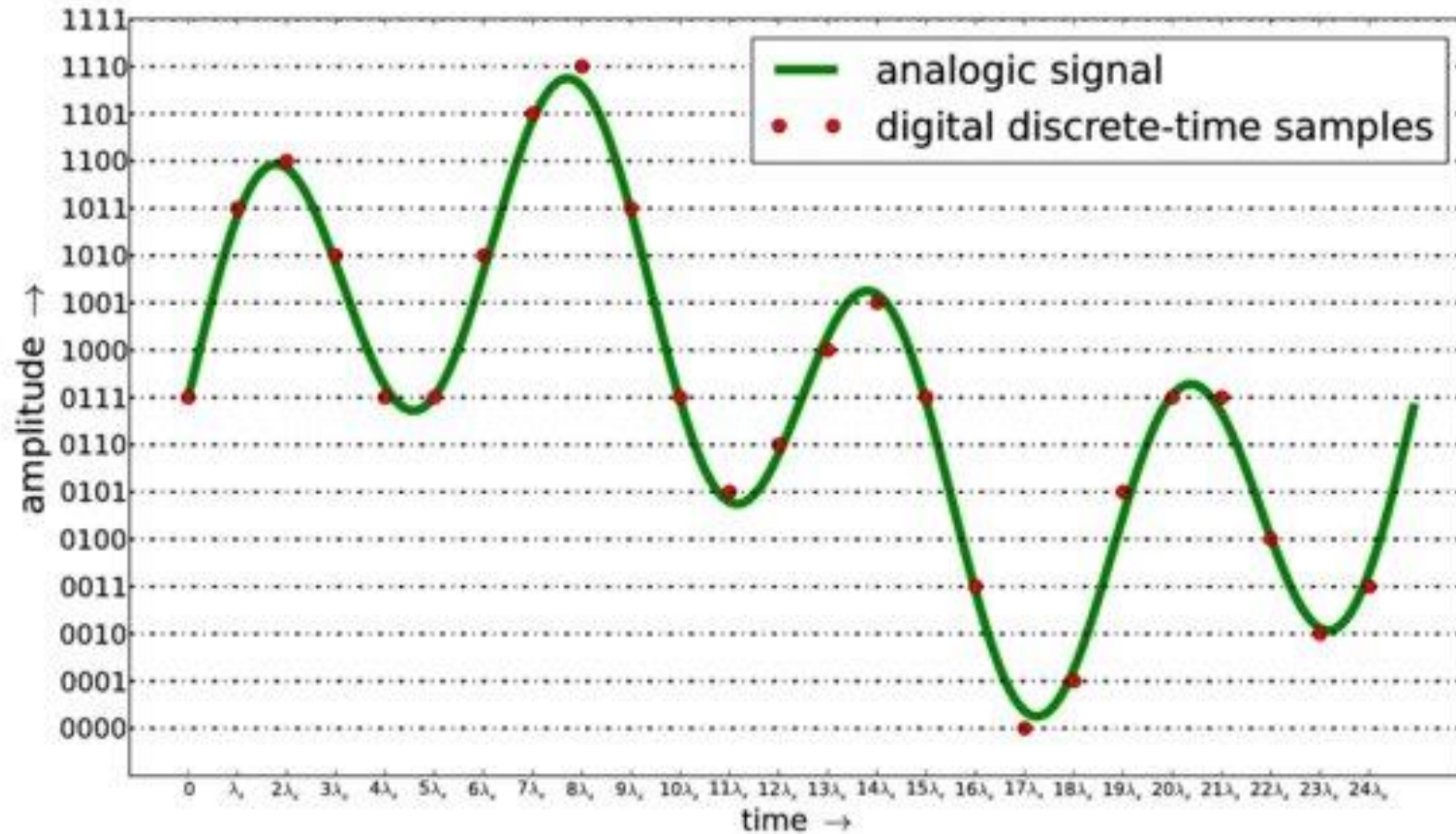
$$LSB = \frac{Span}{2^n - 1}$$

όπου n τα διαθέσιμα bits του μετατροπέα.



1. <https://www.electronics-tutorials.ws/combination/analogue-to-digital-converter.html>
2. <http://www.onmyphd.com/?p=analog.digital.converter>
3. <https://dewesoft.com/daq/types-of-adc-converters>

### 3. Communication and ADC Analog to Digital Converter



1. [https://www.researchgate.net/publication/269935208\\_Psychophysics\\_of\\_musical\\_elements\\_in\\_the\\_discrete-time\\_representation\\_of\\_sound/figures?lo=1](https://www.researchgate.net/publication/269935208_Psychophysics_of_musical_elements_in_the_discrete-time_representation_of_sound/figures?lo=1)



### 3. *Communication and ADC*

#### *Εντολές προγραμματισμού*

Εντολές που μπορούν να χρησιμοποιηθούν στο Arduino Uno όπως έχουμε δει:

- `pinMode(pin, value), delay(value)`
- `digitalRead(pin), digitalWrite(pin, value)`
- `analogWrite(pin, value)`

Νέες εντολές:

- `Serial.begin(value)`: ανοίγει την σειριακή θύρα - η τιμή θέτει τον ρυθμό bits per seconds
- `Serial.print()`: τυπώνει χαρακτήρες ASCII στην σειριακή θύρα
- `Serial.println()`: ό,τι και η προηγούμενη. Επιπλέον ο κέρσορας πηγαίνει σε νέα γραμμή
- `Serial.available()`: επιστρέφει τον αριθμό των χαρακτήρων που είναι διαθέσιμοι για ανάγνωση στην σειριακή θύρα
- `analogRead(pin)`: διαβάζει αναλογικό σήμα από το pin κι επιστρέφει έναν 10-bit αριθμό

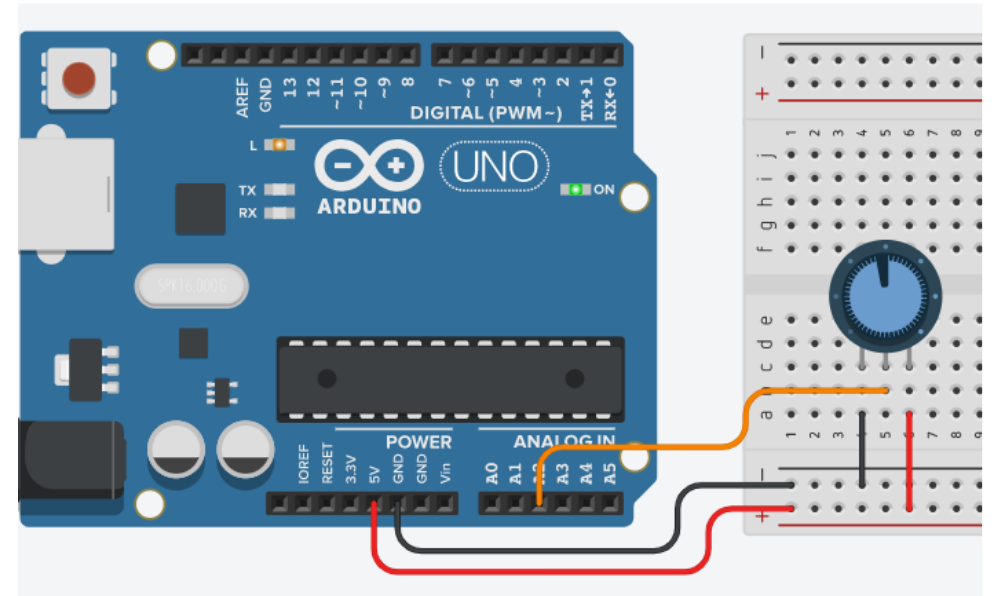
1. <https://www.arduino.cc/reference/en/language/functions/communication/serial/>

2. <https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/>

### 3. Communication and ADC Παράδειγμα

Το παράδειγμα χρησιμοποιεί το ενσωματωμένο LED που συνδέεται στο pin 13 και ένα ποτενσιόμετρο που συνδέεται στο pin A2.

Το LED αναβοσβήνει με ταχύτητα που ρυθμίζεται από το ποτενσιόμετρο.



Σύνδεση κυκλώματος

## 3. *Communication and ADC*

### *Παράδειγμα*

Ο κώδικας:

---

```
// https://www.arduino.cc/en/tutorial/potentiometer
/* Analog Read to LED
 * -----
 *
 * turns on and off a light emitting diode(LED) connected to digital
 * pin 13. The amount of time the LED will be on and off depends on
 * the value obtained by analogRead(). In the easiest case we connect
 * a potentiometer to analog pin 2.
 *
 * Created 1 December 2005
 * copyleft 2005 DojoDave <http://www.0j0.org>
 * http://arduino.berlios.de
 *
 */
int potPin = 2; // select the input pin for the potentiometer
int ledPin = 13; // select the pin for the LED
int val = 0; // variable to store the value coming from the sensor

void setup() {
  pinMode(ledPin, OUTPUT); // declare the ledPin as an OUTPUT
}

void loop() {
  val = analogRead(potPin); // read the value from the sensor
  digitalWrite(ledPin, HIGH); // turn the ledPin on
  delay(val); // stop the program for some time
  digitalWrite(ledPin, LOW); // turn the ledPin off
  delay(val); // stop the program for some time
}
```

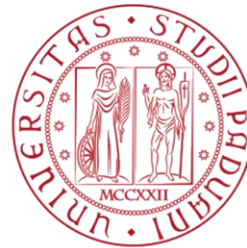
---

## ENGINE Partnership

- Warsaw University of Technology (PL) - *coordinator*
- IHU - International Hellenic University (GR)
- EDUMOTIVA - European Lab for Educational Technology (GR)
- University of Padova (IT)
- University of Applied Sciences in Tarnow (PL)



INTERNATIONAL  
HELLENIC  
UNIVERSITY



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



## CONTACT:



[www.engined.eu](http://www.engined.eu)



[angelika.tefelska@pw.edu.pl](mailto:angelika.tefelska@pw.edu.pl)



[@projectENGINE1](https://twitter.com/projectENGINE1)



[@EUprojectEngine](https://www.facebook.com/EUprojectEngine)



# Erasmus+

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.