

ENGINE

Teaching online electronics, microcontrollers and programming in Higher Education

**Output 2: Online Course for Microcontrollers:
syllabus, open educational resources**

Practice leaflet: Module_1-3 Communication and
ADC

Lead Partner: International Hellenic University (IHU)

Δήλωση

Αυτό το αρχείο συντάχθηκε στο πλαίσιο του έργου ENGINE. Όπου έχουν χρησιμοποιηθεί άλλα δημοσιευμένα και αδημοσίευτα υλικά, αυτά έχουν αναγνωρισθεί.

Πνευματική ιδιοκτησία

© Copyright 2021 - 2023 the [ENGINE](#) Consortium

Warsaw University of Technology (Poland)

International Hellenic University (IHU) (Greece)

European Lab for Educational Technology- EDUMOTIVA (Greece)

University of Padova (Italy)

University of Applied Sciences in Tarnow (Poland)

Όλα τα δικαιώματα διατηρούνται.



Αυτό το έγγραφο έχει άδεια Creative Commons Attribution-NonCommercial- NoDerivatives 4.0 International License.

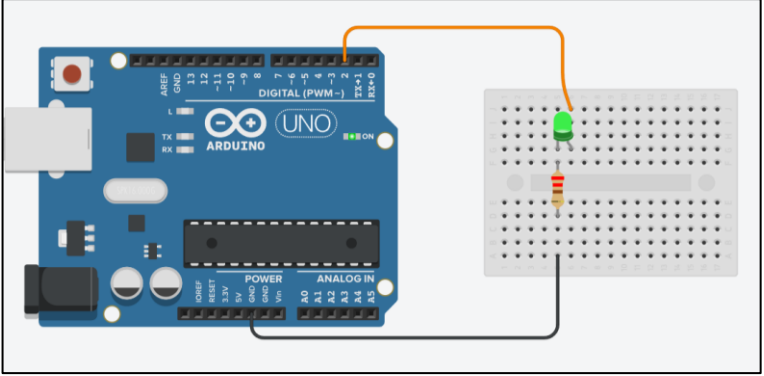
Αυτό το έργο έχει χρηματοδοτηθεί με την υποστήριξη της Ευρωπαϊκής Επιτροπής. Αυτή η έκθεση αντικατοπτρίζει μόνο τις απόψεις του συγγραφέα και η Επιτροπή δεν μπορεί να θεωρηθεί υπεύθυνη για οποιαδήποτε χρήση των πληροφοριών που περιέχονται σε αυτήν.

Πίνακας Περιεχομένων

Δραστηριότητα 1. Serial και LED	1
Δραστηριότητα 2. Analog to Digital Converter	4
Ανακεφαλαίωση.....	5

Δραστηριότητα 1. Serial και LED

Αυτή η δραστηριότητα χρησιμοποιεί σειριακή επικοινωνία μεταξύ του Arduino Uno και του monitor.

<p>Δραστηριότητα 1a (15 λεπτά)</p>	<p>Σε αυτό το μέρος στόχος είναι το Arduino Uno να αναβοσβήνει ένα LED κάθε 5 δευτερόλεπτα. Κάθε φορά που αλλάζει κατάσταση το LED το Arduino Uno στέλνει αντίστοιχο μήνυμα στην σειριακή επικοινωνία.</p> <ul style="list-style-type: none">• Τα δεδομένα που στέλνει το Arduino Uno μέσω της σειριακής θύρας φαίνονται στο monitor <p>Βήμα 1. Υλοποιείται το κύκλωμα Βήμα 2. Γράφεται ο κώδικας του μικροελεγκτή Βήμα 3. Γίνεται έλεγχος λειτουργίας του κυκλώματος</p>
<p>Βήμα 1 (5 λεπτά)</p>	<p>Να υλοποιηθεί το επόμενο κύκλωμα.</p>  <p>Εικόνα.1.1 LED και serial communication</p>

3. Communication and ADC - Practice leaflet

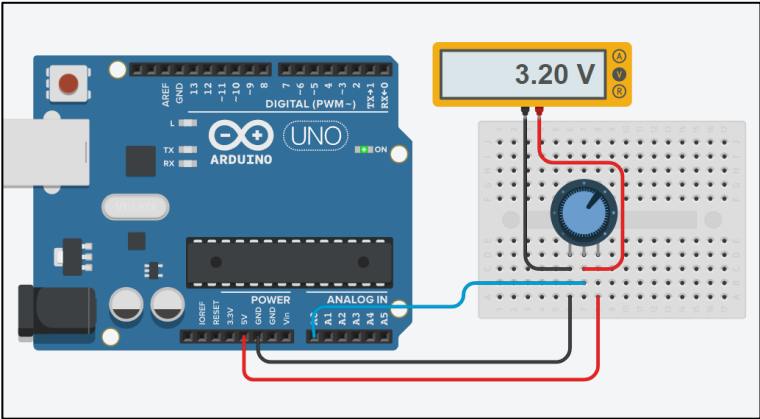
<p>Βήμα 2 (8 λεπτά)</p>	<p>Να μελετηθεί ο κώδικας, να συμπληρωθεί και να γραφτεί στον μικροελεγκτή:</p> <hr/> <pre>/* Blinking a LED and print to Serial Συνδεσμολογία κυκλώματος: PIN_2 => LED_Anode - LED_Cathode => Αντίσταση 220Ω => Gnd PIN_0 => Serial_RX PIN_1 => Serial_TX */ //Η συνάρτηση setup() εμπεριέχει αρχικοποιήσεις //Εκτελείται μόνο μία φορά κατά την τροφοδοσία του Arduino Uno void setup() { //Ρυθμίζεται το PIN_2 ως έξοδος pinMode(2, OUTPUT); //άνοιγμα της σειριακής και αρχικοποίηση στα 9600 bps Serial.begin(9600); } //Αυτή η συνάρτηση εκτελείται διαρκώς void loop() { digitalWrite(2, HIGH); //Γράψε HIGH (5V) στο pin 2 Serial.println("LED is ON");//αποστολή δεδομένων «LED is ON» delay(5000); // Αναμονή 5 δευτερολέπτων digitalWrite(2, LOW); //Γράψε LOW (0V) στο pin 2 //αποστολή δεδομένων «LED is OFF» delay(5000); // Αναμονή 5 δευτερολέπτων }</pre> <hr/>								
<p>Βήμα 3 (2 λεπτά)</p>	<p>Έλεγχος σωστής λειτουργίας του κυκλώματος.</p> <p>Tip. Ανοίξτε το monitor του Arduino Uno ώστε να φαίνονται τα μηνύματα που στέλνει.</p>								
<p>Δραστηριότητα 1b (10 λεπτά)</p>	<p>Σε αυτό το μέρος στόχος είναι το Arduino Uno</p> <ul style="list-style-type: none"> να ανάβει ή να σβήνει ένα LED σύμφωνα με τις εντολές που διαβάζει από την σειριακή επικοινωνία <p>Πίνακας 1. Πιθανές εντολές από την σειριακή</p> <table border="1" data-bbox="635 1697 1189 1861"> <thead> <tr> <th>Εντολή</th> <th>Ρύθμιση</th> </tr> </thead> <tbody> <tr> <td>“on”</td> <td>LED on</td> </tr> <tr> <td>“off”</td> <td>LED off</td> </tr> <tr> <td colspan="2">Οτιδήποτε άλλο θεωρείται λάθος εντολή</td> </tr> </tbody> </table> <p>**Το κύκλωμα είναι ίδιο με την δραστηριότητα 1a**</p> <p>Βήμα 1. Γράφεται ο κώδικας του μικροελεγκτή</p>	Εντολή	Ρύθμιση	“on”	LED on	“off”	LED off	Οτιδήποτε άλλο θεωρείται λάθος εντολή	
Εντολή	Ρύθμιση								
“on”	LED on								
“off”	LED off								
Οτιδήποτε άλλο θεωρείται λάθος εντολή									

3. Communication and ADC - Practice leaflet

	<p>Βήμα 2. Γίνεται έλεγχος λειτουργίας του κυκλώματος Βήμα 3. Τροποποιήσεις και συζήτηση</p>
<p>Βήμα 2 (7 λεπτά)</p>	<p>Να μελετηθεί ο κώδικας και να γραφτεί στον μικροελεγκτή.</p> <hr/> <p>/* ON/OFF a LED by Serial communication</p> <p>Συνδεσμολογία κυκλώματος: PIN_2 => LED_Anode - LED_Cathode => Αντίσταση 220Ω => Gnd PIN_0 => Serial_RX PIN_1 => Serial_TX */</p> <p>String input; //για αποθήκευση δεδομένων από την σειριακή</p> <p>//Η συνάρτηση setup() περιέχει αρχικοποιήσεις //Εκτελείται μόνο μία φορά κατά την τροφοδοσία του Arduino Uno void setup() { //Ρυθμίζεται το PIN_2 ως έξοδος pinMode(2, OUTPUT); //άνοιγμα της σειριακής και αρχικοποίηση στα 9600 bps Serial.begin(9600); } //Αυτή η συνάρτηση εκτελείται διαρκώς void loop() { //έλεγχος για δεδομένα στην σειριακή if (Serial.available() > 0){ //ανάγνωση και αποθήκευση δεδομένων input = Serial.readString(); //έλεγχος δεδομένων if(input == "on"){ //LED is on digitalWrite(2, HIGH); // Γράψε HIGH (5V) στο pin 2 } else if(input=="off"){ //LED is off digitalWrite(2, LOW); // Γράψε LOW (0V) στο pin 2 } else{ Serial.println("Wrong command"); } } } }</p>
<p>Βήμα 3 (2 λεπτά)</p>	<p>Έλεγχος σωστής λειτουργίας του κυκλώματος</p>
<p>Βήμα 4 (1 λεπτά)</p>	<p>Προτεινόμενες τροποποιήσεις και συζήτηση:</p> <ul style="list-style-type: none"> • θα μπορούσαν εξαρτήματα να συνδεθούν στα PIN_0, PIN_1;

Δραστηριότητα 2. Analog to Digital Converter

Αυτή η δραστηριότητα χρησιμοποιεί τον ενσωματωμένο μετατροπέα αναλογικού σήματος σε ψηφιακό του Arduino Uno.

<p>Δραστηριότητα 2 (25 λεπτά)</p>	<p>Στόχος είναι το Arduino Uno</p> <ul style="list-style-type: none"> • να διαβάσει την αναλογική τάση ενός ποτενσιόμετρου • να την στέλνει στην σειριακή επικοινωνία <p>Στο κύκλωμα έχει προστεθεί ένα βολτόμετρο για τον έλεγχο της τάσης.</p> <p>Βήμα 1. Υλοποιείται το κύκλωμα Βήμα 2. Γράφεται ο κώδικας του μικροελεγκτή Βήμα 3. Γίνεται έλεγχος λειτουργίας του κυκλώματος Βήμα 4. Τροποποιήσεις και συζήτηση</p>
<p>Βήμα 1 (5 λεπτά)</p>	<p>Να υλοποιηθεί το επόμενο κύκλωμα.</p>  <p>Εικόνα.1.2 ADC και serial</p>
<p>Βήμα 2 (10 λεπτά)</p>	<p>Να μελετηθεί ο κώδικας και να γραφτεί στον μικροελεγκτή:</p> <hr/> <pre> /* ADC and serial Συνδεσμολογία κυκλώματος: Potensiometer_Terminal_1 => Gnd Potensiometer_Wiper => A0 Potensiometer_Terminal_2 => Vcc PIN_0 => Serial_RX PIN_1 => Serial_TX */ #define pot_pin A0 //ονομάζουμε "pot_pin" το PIN_A0 //για αποθήκευση δεδομένων από τον ADC int adc_value; //εύρος αριθμών 0~1023 float voltage; //μεταβλητή για την αναλογική τάση </pre>

3. Communication and ADC - Practice leaflet

	<pre>//Η συνάρτηση setup() εμπεριέχει αρχικοποιήσεις //Εκτελείται μόνο μία φορά κατά την τροφοδοσία του Arduino Uno void setup() { Serial.begin(9600); } //Αυτή η συνάρτηση εκτελείται διαρκώς void loop() { //διαβάζει την αναλογική τιμή και επιστρέφει αριθμό adc_value = analogRead(pot_pin); //υπολογισμός της αναλογικής τάσης από τον αριθμό του ADC voltage = float(adc_value)/1023*5; //τύπωση στην σειριακή Serial.print("ADC number: "); Serial.println(adc_value); Serial.print("Voltage = "); Serial.print(voltage); Serial.println("V"); //αναμονή για 5 δευτερόλεπτα delay(5000); }</pre>
Βήμα 3 (5 λεπτά)	Έλεγχος σωστής λειτουργίας του κυκλώματος
Βήμα 4 (5 λεπτά)	Προτεινόμενες τροποποιήσεις και συζήτηση: <ul style="list-style-type: none">• Ποιος είναι ο μέγιστος αριθμός μερών που μπορούμε να χωρίσουμε τις τιμές του ποτενσιόμετρου;• Ποια είναι η μέγιστη αναλογική τάση που μπορεί να μετρήσει ο ADC του Arduino Uno

Ανακεφαλαίωση

Χρησιμοποιήθηκαν βασικές εντολές προγραμματισμού του Arduino, όπως:

- delay()
- analogWrite()
- digitalWrite()
- analogRead()
- Serial.available()

Μέσα από τις δραστηριότητες αξιοποιήθηκαν

- pins του Arduino Uno ως αναλογικοί είσοδοι
- αμφίδρομη σειριακή επικοινωνία