

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

# **ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ:** ΕΚΠΟΜΠΗ ΚΑΙ ΛΗΨΗ ΣΗΜΑΤΩΝ ΜΕ ΤΗ ΒΟΗΘΕΙΑ ΜΙΚΡΟΫΠΟΛΟΓΙΣΤΩΝ



**ТZONI ПІРРА** АМ:15237

Υπεύθυνος καθηγητής: Σπυρίδων Χρονόπουλος

Άρτα, Σεπτέμβριος 2018

ΤΕΙ ΗΠΕΙΡΟΥ 2018

# TRANSMITTING AND RECEIVING SIGNALS USING MICROCOMPUTERS

**TEI HITEIPOY 2018** 



#### Εγκρίθηκε από τριμελή εξεταστική επιτροπή

Τόπος, Ημερομηνία:....

### ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

- Επιβλέπων καθηγητής
   Όνομα Επίθετο,
   τίτλος, βαθμίδα
- 2. Μέλος επιτροπής
   Όνομα Επίθετο,
   τίτλος, βαθμίδα
- 3. Μέλος επιτροπής
   Όνομα Επίθετο,
   τίτλος, βαθμίδα

Ο/Η Προϊστάμενος/η του Τμήματος Όνομα Επίθετο, τίτλος, βαθμίδα ΤΕΙ ΗΠΕΙΡΟΥ 2018

© Πίρρα Τζόνι 2018. Με επιφύλαξη παντός δικαιώματος. All rights reserved.



•

## Δήλωση μη λογοκλοπής

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα πτυχιακή εργασία είναι εξ ολόκληρο αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία

Πίρρα Τζόνι

Υπογραφή

ΤΕΙ ΗΠΕΙΡΟΥ 2018

#### ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τους γονείς μου για την υποστήριξη που μου δώσανε σε όλα τα χρόνια της σταδιοδρομίας μου.

Ευχαριστώ τον υπεύθυνο καθηγητή μου κ. Σπυρίδωνα Χρονόπουλο για όλη την υποστήριξη και καθοδήγηση που μου παρείχε σε όλο το διάστημα των σπουδών μου.

Έπειτα θα ήθελα να ευχαριστήσω τον κ. Παναγή Νικολάτο επιβλέποντα υπεύθυνο της πρακτικής μου άσκησης, για την βοήθεια και στήριξή του κατά την υλοποίηση της πτυχιακής μου εργασίας.

ΤΕΙ ΗΠΕΙΡΟΥ 2018



## ΠΕΡΙΛΗΨΗ

Στη παρούσα εργασία παρουσιάζονται και συγκρίνονται αρχικά, διάφορα συστήματα μικροϋπολογιστών. Στη συνέχεια παρουσιάζεται ένα από τα μελετημένα συστήματα το οποίο και υλοποιήθηκε στην πράξη. Το προτεινόμενο υλοποιημένο σύστημα εκπέμπει και λαμβάνει ασύρματα, σήματα πληροφορίας θερμοκρασίας και υγρασίας τα οποία ανιχνεύονται από κατάλληλο αισθητήρα. Περαιτέρω προσθήκες στο σύστημα εκπομπής και λήψεως είναι η μετάδοση των δεδομένων που εισάγονται από ένα Arduino σε ένα άλλο μέσω ασύρματης ζεύξης WiFi με χρησιμοποίηση σημείου πρόσβασης cAP lite.



ΤΕΙ ΗΠΕΙΡΟΥ 2018



#### ABSTRACT

In this work, various microcomputer systems are presented and compared between them. Then one of the studied systems is presented as it has been implemented in practice. The proposed implemented system transmits and receives wirelessly temperature and humidity information signals which are detected by a selected sensor. Further additions to the broadcasting and receiving system are the transmission of data imported from one Arduino to another through a wireless WiFi connection using a cAP lite access point.



ΤΕΙ ΗΠΕΙΡΟΥ 2018



# Πίνακας περιεχομένων

ΕY	XAPI	ΙΣΤΙΕΣ	. 9
ПВ	ΡΙΛΙ	ΗΨΗ	11
AE	STR	ACT	13
1.	Δι	ιάφορα συστήματα μικροϋπολογιστών	19
		Σχήμα 1. Εμφάνιση πλακέτας μικροϋπολογιστή [2]	. 19
	1.1	Raspberry Pi	20
		Σχήμα 2. Εμφάνιση συστήματος Raspberry Pi [4]	. 20
	1.2	BeagleBoard	21
		Σχήμα 3. Εμφάνιση συστήματος BeagleBoard [6]	. 21
	1.3	Banana Pi	22
		Σχήμα 4. Εμφάνιση συστήματος Banana Pi [8]	. 22
	1.4	Intel Galileo	23
		Σχήμα 5. Εμφάνιση συστήματος Intel Galileo [10]	. 23
2.	Ai	rduino	24
	2.1	Τι είναι το Arduino;	24
		Σχήμα 6. Εμφάνιση συστήματος Arduino [12]	. 24
	2.2	Λόγος χρησιμοποίησης Arduino	25
		Σχήμα 7. Περιφερειακά, βοηθητικά εξαρτήματα, καλώδια [13]	. 25
	2.3	Προϊόντα Arduino	27
	2.	3.1 Επίπεδο εισόδου	27
		Σχήμα 8. Διάφορα συστήματα Arduino [14]	. 27
	2.	3.2 Ενισχυμένα χαρακτηριστικά	28
		Σχήμα 9. Ενισχυμένα χαρακτηριστικά Arduino [14]	. 28
	2.	3.3 Internet of Things (IoT)	29
		Σχήμα 10. Διασυνδεσιμότητα [14]	. 29
		Σχήμα 11. Περιφερειακές πλακέτες συστήματος [14]	. 30
3.	Eγ	γκατάσταση του IDE του Arduino Desktop	31
	3.1	Εγκατάσταση του λογισμικού Arduino (IDE) σε υπολογιστές με Windows	31
		Σχήμα 12. Εικόνα οδηγού εγκατάστασης Windows και επιλογή των στοιχείων που θα εγκατασταθού <sup>.</sup> [16].	v .31
		Σχήμα 13. Εικόνα οδηγού εγκατάστασης Windows και επιλογή του καταλόγου εγκατάστασης (συνήθη η προεπιλεγμένη) [16]	ως . 32
		Σχήμα 14. Εικόνα οδηγού εγκατάστασης Windows - Κατά τη διαδικασία εγκατάστασης θα εξαχθούν κ θα εγκατασταθούν όλα τα απαιτούμενα αρχεία για τη σωστή εκτέλεση του λογισμικό Arduino (IDE) [16]	και . 32
	3.2	Εγκατάσταση του λογισμικού Arduino (IDE) στο OS Χ	33
		Σχήμα 15. Εικόνα οδηγού εγκατάστασης OS X [17].	. 33

	Σχήμα 16. Εικόνα οδηγού εγκατάστασης OS Χ: Αντιγραφή της εφαρμογής Arduino στο φάκελο "Εφαρμογές" (ή οπουδήποτε αλλού στην αποθηκευτική μονάδα του υπολογιστού) [17]	33
3.3	3 Εγκατάσταση του λογισμικού Arduino (IDE) στο Linux	34
	Σχήμα 17. Εικόνα οδηγού εγκατάστασης Linux [18]	34
	Σχήμα 18. Εικόνα οδηγού εγκατάστασης Linux: Το αρχείο είναι συμπιεσμένο και θα πρέπει να γ εξαγωγή σε έναν κατάλληλο φάκελο, υπενθυμίζοντας την εκτέλεσή του από εκεί.[18]	ίνει 34
	Σχήμα 19. Εικόνα οδηγού εγκατάστασης Linux [18].	35
4.	Κώδικας	37
4.2	1 Γλώσσα Προγραμματισμού	37
	Σχήμα 20. Λογότυπο γλώσσας προγραμματισμού C++ [20]	
4.2	2 Γενική εικόνα	37
	Σχήμα 21. Προσχεδιασμός [21]	
4.3	3 Προεπεξεργασία	38
	Σχήμα 22. Από τον κώδικα στη πλακέτα [22]	
5.	Ζεύξη	41
5.2	1 MikroTik	41
	Σχήμα 23. Λογότυπο εταιρείας MikroTik [23]	41
5.2	2 cAP lite	42
	Σχήμα 24. Εμφάνιση κεραίας cAP [24]	42
5.3	3 RB260GSP	42
	Σχήμα 25. Εμφάνιση Switch RB260GSP [25]	42
5.4	4 Γεφύρωση (Bridging)	43
	Σχήμα 27. Μοντελοποίηση Γεφύρωσης [27]	43
6.	Αναλυτική περιγραφή της υλοποίησης του ασύρματου συστήματος	45
	Σχήμα 28. Αποτελέσματα Web Client	45
	Σχήμα 29. Ολοκληρωμένο σύστημα	46
6.2	1 Εξυπηρετητής ιστού (Web Server)	46
	Σχήμα 30. Σύστημα Web Server	47
6.2	2 Πελάτης ιστού (Web Client)	48
	Σχήμα 31. Σύστημα Web Client	48
6.3	3 Αισθητήρας υγρασίας και θερμοκρασίας	49
	Σχήμα 32. Μοντέλο Αισθητήρα [28]	49
6.4	4 Ethernet Shield	49
	Σχήμα 33. Εμφάνιση Ethernet Shield [29]	50
7.	Συμπεράσματα - Μελλοντικοί Στόχοι	50
8.	ΒΙΒΛΙΟΓΡΑΦΙΑ (REFERENCES)	51
ПАРА	ΑΡΤΗΜΑ ΠΗΓΑΙΟΥ ΚΩΔΙΚΑ	55

Π1. Server	55
Π2. Client	



ΤΕΙ ΗΠΕΙΡΟΥ 2018



### 1. Διάφορα συστήματα μικροϋπολογιστών

Ένας μικροελεγκτής είναι ένας μικρός υπολογιστής που περιλαμβάνεται σε ένα και μόνο ολοκληρωμένο κύκλωμα. Στη σύγχρονη ορολογία, αναφέρεται ως «σύστημα σε ένα chip» (System On Chip ή SoC). Ένα SoC μπορεί να περιλαμβάνει έναν μικροελεγκτή ο οποίος μπορεί να περιέχει έναν ή περισσότερους επεξεργαστές (πυρήνες επεξεργαστών) μαζί με μνήμη και προγραμματιζόμενα περιφερειακά εισόδου / εξόδου. Η μνήμη προγραμμάτων με τη μορφή μνήμης RAM, NOR flash ή OTP ROM συμπεριλαμβάνεται επίσης συχνά στο τσιπ, καθώς και μια μικρή ποσότητα μνήμης RAM. Οι μικροελεγκτές είναι σχεδιασμένοι για ενσωματωμένες εφαρμογές, σε αντίθεση με τους μικροεπεξεργαστές που χρησιμοποιούνται σε προσωπικούς υπολογιστές ή άλλες εφαρμογές γενικού σκοπού [1].



Σχήμα 1. Εμφάνιση πλακέτας μικροϋπολογιστή [2]

Οι μικροελεγκτές χρησιμοποιούνται σε αυτοματισμούς, όπως συστήματα ελέγχου κινητήρα αυτοκινήτων, εμφυτεύσιμα ιατρικά βοηθήματα, τηλεχειριστήρια, μηχανές γραφείου, συσκευές, ηλεκτρικά εργαλεία, παιχνίδια και άλλα ενσωματωμένα συστήματα. Με τη μείωση του μεγέθους και του κόστους σε σύγκριση με ένα σχέδιο που χρησιμοποιεί ξεχωριστό μικροεπεξεργαστή, μνήμη και συσκευές εισόδου / εξόδου, οι μικροελεγκτές καθιστούν οικονομικό τον ψηφιακό έλεγχο ακόμα περισσότερων συσκευών και διαδικασιών. Οι μικροελεγκτές μικτού σήματος είναι συνηθισμένοι, ενσωματώνοντας αναλογικά στοιχεία που απαιτούνται για τον έλεγχο μη ψηφιακών ηλεκτρονικών συστημάτων.[1]

Ορισμένοι μικροελεγκτές μπορούν να χρησιμοποιούν λέξεις τεσσάρων δυαδικών ψηφίων και να λειτουργούν σε συχνότητες όπως 4 kHz, για χαμηλή κατανάλωση ενεργείας (milliwatts ή microwatts). Έχουν γενικά τη δυνατότητα να διατηρούν τη stand-by λειτουργικότητά τους ενόσω περιμένουν ένα συμβάν για να ενεργοποιηθούν, όπως ένα πάτημα κουμπιού ή άλλη διακοπή. Η κατανάλωση ενέργειας σε κατάσταση αδρανείας ανέρχεται σε μερικά nanowatts, καθιστώντας πολλούς από αυτούς κατάλληλους για εφαρμογές βασιζόμενες σε μακρά διάρκεια μπαταριών. Άλλοι μικροελεγκτές μπορεί να εξυπηρετούν εφαρμογές όπου η επίδοση είναι κρίσιμη όπως η ψηφιακή επεξεργασία σήματος (DSP), όπου απαιτούνται υψηλότερες ταχύτητες ρολογιού και άρα υψηλότερη κατανάλωση ενέργειας [1].

#### 1.1 Raspberry Pi

Το Raspberry Pi είναι μια σειρά μικροϋυπολογιστών που περιορίζονται σε μία και μόνο πλακέτα κυκλώματος (single-board-computers ή sbc). Αναπτύχθηκαν στο Ηνωμένο Βασίλειο από το ίδρυμα Raspberry Pi Foundation για να προωθήσουν τη διδασκαλία της επιστήμης των υπολογιστών στα σχολεία και στις αναπτυσσόμενες χώρες. Το αρχικό μοντέλο έγινε πολύ πιο δημοφιλές από ό,τι αναμενόταν ώστε να χρησιμοποιείται και σε εφαρμογές (π.χ. ρομποτική) πέραν εκείνων που ανήκαν στην στοχευόμενη αγορά. Δεν περιλαμβάνει περιφερειακά (όπως πληκτρολόγιο, ποντίκι υπολογιστού και άλλα). Ωστόσο, ορισμένα εξαρτήματα έχουν συμπεριληφθεί σε διάφορες επίσημες και ανεπίσημες εκδόσεις [3].



Σχήμα 2. Εμφάνιση συστήματος Raspberry Pi [4].

#### 1.2 BeagleBoard

To BeagleBoard είναι ένας υπολογιστής ανοικτής αρχιτεκτονικής, χαμηλής ισχύος που κατασκευάζεται από την Texas Instruments σε συνεργασία με τις εταιρείες Digi-Key και Newark14. Οι σχεδιαστές του BeagleBoard είχαν κατά νου για την εφαρμογή του σε ένα σύστημα που θα εξυπηρετούσε την ανάπτυξη λογισμικού ανοιχτού κώδικα όπως και την παρουσίαση των δυνατοτήτων του συστήματος OMAP3530 της Texas Instruments (στο σύστημα αυτό βασίστηκε και το BeagleBoard). Το BeagleBoard αναπτύχθηκε από μια μικρή ομάδα μηχανικών για να χρησιμοποιηθεί σε κολέγια σε όλο τον κόσμο που ενδιαφέρονται να διδάξουν τις δυνατότητες υλισμικού και λογισμικού ανοιχτού κώδικα. Επίσης πωλείται στο κοινό με την άδεια Creative Commons share-alike. Για τη σχεδίαση των κυκλωμάτων του χρησιμοποιήθηκε από τους κατασκευαστές του το Cadence OrCAD και το Cadence Allegro για την κατασκευή των τυπωμένων κυκλωμάτων [5].



Σχήμα 3. Εμφάνιση συστήματος BeagleBoard [6].



#### 1.3 Banana Pi

Ως Banana Pi ονομάζεται μια σειρά υπολογιστών χαμηλού κόστους, με μέγεθος πιστωτικής κάρτας και παράγονται από την κινεζική εταιρεία Shenzhen SINOVOIP Co. Ltd. Οι σχεδιαστές του υλικού των υπολογιστών Banana Pi επηρεάστηκαν από το Raspberry Pi [7].

Η πλακέτες του Banana Pi είναι συμβατή με τις πλακέτες του Raspberry Pi. Το Banana Pi μπορεί επίσης να χρησιμοποιηθεί στα λειτουργικά συστήματα NetBSD, Android, Ubuntu, Debian, Arch Linux, Raspbian, αν και η CPU συμμορφώνεται με τις απαιτήσεις της θύρας armhf του Debian. Χρησιμοποιεί το Allwinner SoC (σύστημα στο τσιπ) και ως εκ τούτου καλύπτεται από τη πλακέτα linux-sunxi.[7]. Επίσης, το Banana Pi είναι το λογισμικό ανοιχτού κώδικα και η πλατφόρμα λογισμικού η οποία έχει σχεδιαστεί ώστε να πλαισιώνει την banana-pi.org.[7].



Σχήμα 4. Εμφάνιση συστήματος Banana Pi [8].



#### 1.4 Intel Galileo

Το Intel Galileo είναι το πρώτο σε μια σειρά από πλατφόρμες ανάπτυξης με την πιστοποίηση της Arduino που βασίζεται στην αρχιτεκτονική Intel x86 και έχει σχεδιαστεί για τους κατασκευαστές και για διάφορες εκπαιδευτικές κοινότητες προσανατολισμένες στη χρήση μικροελεγκτών και μικροεπεξεργαστών. Η Intel κυκλοφόρησε δύο εκδόσεις του Galileo, που αναφέρονται ως Gen 1 και Gen 2. Αυτές οι πλακέτες ανάπτυξης ονομάζονται μερικές φορές ως «Breakout boards». Όμως, η εταιρεία διέκοψε την παραγωγή της πλακέτας στις 19 Ιουνίου 2017 [9].



Σχήμα 5. Εμφάνιση συστήματος Intel Galileo [10].



#### 2. Arduino

#### 2.1 Τι είναι το Arduino;

Το Arduino είναι μια ηλεκτρονική πλατφόρμα ανοικτού κώδικα βασισμένη σε εύχρηστο υλικό και λογισμικό. Οι πλακέτες Arduino είναι σε θέση να διαβάζουν εισόδους (π.χ. πρόσπτωση φωτός σε αισθητήρα), το πάτημα ενός κουμπιού ή ένα μήνυμα Twitter και να το μετατρέπουν σε έξοδο ενεργοποιώντας ένα μοτέρ, ενεργοποιώντας ένα LED ή και δημοσιεύοντας προκαθορισμένες πληροφορίες στο διαδίκτυο. Η λειτουργία της πλακέτας ρυθμίζεται στέλνοντας ένα σύνολο οδηγιών μέσω κώδικα στον μικροελεγκτή της. Για να επιτευχθεί ο προαναφερθέντας στόχος, χρησιμοποιείται η γλώσσα προγραμματισμού Arduino (βασισμένη στην καλωδίωση), και το λογισμικό Arduino (IDE) [11].

Με τα χρόνια το Arduino αποτέλεσε την καρδιά πολλών εφαρμογών που περιλάμβαναν από καθημερινές ασχολίες έως και σύνθετες επιστημονικές μετρήσειςεφαρμογές. Μια παγκόσμια κοινότητα δημιουργών (φοιτητές, χομπίστες, καλλιτέχνες, προγραμματιστές και επαγγελματίες) έχει επικεντρωθεί σε αυτή την πλατφόρμα ανοιχτού κώδικα, όπου οι συνεισφορές τους έχουν συνδράμει στη δημιουργία μίας τεράστια ποσότητας προσβάσιμης γνώσης που μπορεί να αποτελέσει μεγάλη βοήθεια όχι μόνο για τους αρχάριους αλλά και για ειδικούς [11].



Σχήμα 6. Εμφάνιση συστήματος Arduino [12].



Το Arduino δημιουργήθηκε στο ινστιτούτο Ivrea Interaction Design ως ένα εύκολο εργαλείο για τη γρήγορη δημιουργία πρωτότυπων ιδεών, και που θα απευθύνονταν σε φοιτητές με λιγοστές γνώσεις σε ό,τι αφορά ηλεκτρονικά κυκλώματα και προγραμματισμό. Όταν δε κοινοποιήθηκε σε μια ευρύτερη κοινότητα ηλεκτρονικών και προγραμματιστών, η πλακέτα του Arduino τροποποιήθηκε καταλλήλως ώστε να προσαρμοστεί στις νέες ανάγκες και προκλήσεις. Αποτέλεσμα των προαναφερθέντων ήταν η συμπερίληψή του από απλές κάρτες 8-bit μέχρι και σε εφαρμογές ΙοΤ, φορητές συσκευές, τρισδιάστατους εκτυπωτές κλπ. Όλες οι πλακέτες Arduino είναι πλήρους ανοιχτού κώδικα, επιτρέποντας στους χρήστες να τις προγραμματίζουν και να τις προσαρμόζουν σύμφωνα με τις ανάγκες τους [11].

#### 2.2 Λόγος χρησιμοποίησης Arduino

Η ανοικτή αρχιτεκτονική του Arduino το καθιστά εύκολα προσβάσιμο και γρήγορο στην ανάπτυξη πρωτότυπων κατασκευών και για το λόγο αυτό έχει χρησιμοποιηθεί σε αναρίθμητες εφαρμογές. Το λογισμικό Arduino είναι εύκολο στη χρήση για αρχάριους και αρκετά ευέλικτο για τους προχωρημένους χρήστες. Εκτελείται σε Mac, Windows και Linux.



Σχήμα 7. Περιφερειακά, βοηθητικά εξαρτήματα, καλώδια [13].

Οι εκπαιδευτικοί και οι φοιτητές το χρησιμοποιούν για να κατασκευάζουν επιστημονικά όργανα χαμηλού κόστους, για εφαρμογές στη χημεία και στη φυσική, ή για να εξοικειωθούν με τον προγραμματισμό και τη ρομποτική. Οι σχεδιαστές και οι αρχιτέκτονες κατασκευάζουν διαδραστικά πρωτότυπα, οι μουσικοί και οι καλλιτέχνες το χρησιμοποιούν για τις εγκαταστάσεις τους και το πειραματισμό τους με νέα μουσικά όργανα. Οι κατασκευαστές, φυσικά, το χρησιμοποιούν για να δημιουργήσουν πολλά από τα έργα που π.χ. εκτίθενται στο Maker Faire. Το Arduino είναι ένα βασικό εργαλείο για την εκμάθηση νέων αντικειμένων μελέτης. Το Arduino, επίσης, απλοποιεί τη διαδικασία εργασίας με τους μικροελεγκτές, καθώς προσφέρει πλεονεκτήματα στους εκπαιδευτικούς, στους σπουδαστές και στους ενδιαφερόμενους ερασιτέχνες σε σύγκριση με άλλα συστήματα [11]:

- Είναι φθηνή αρχιτεκτονική Οι πλακέτες Arduino είναι σχετικά φθηνές σε σύγκριση με άλλες πλατφόρμες μικροελεγκτών. Η λιγότερο δαπανηρή έκδοση ενός μοντέλου Arduino μπορεί να συναρμολογηθεί πολύ εύκολα και ακόμη και οι προ-συναρμολογημένες μονάδες Arduino κοστίζουν λιγότερο από 50 ευρώ [6].
- Έχει διαλειτουργικότητα (Cross-platform) Το λογισμικό Arduino (IDE) λειτουργεί σε λειτουργικά συστήματα Windows, Macintosh OSX και Linux. Τα περισσότερα συστήματα μικροελεγκτών περιορίζονται στα Windows.
- Παρέχει απλό και σαφές περιβάλλον προγραμματισμού Το λογισμικό Arduino (IDE) είναι εύκολο στη χρήση από αρχάριους, αλλά αρκετά ευέλικτο για να επωφεληθούν και οι προηγμένοι χρήστες όπως προαναφέρθηκε. Σε ό,τι αφορά τους εκπαιδευτικούς, είναι κατάλληλα βασισμένο στο περιβάλλον προγραμματισμού διαδικασιών, έτσι ώστε οι μαθητές που καλούνται να μάθουν προγραμματισμό σε αυτό το περιβάλλον να είναι εξοικειωμένοι με τον τρόπο λειτουργίας του IDE του Arduino.
- Είναι Λογισμικό ανοιχτού κώδικα και είναι επεκτάσιμο Το λογισμικό Arduino είναι γνωστό ως εργαλείο ανοικτού κώδικα, διαθέσιμο για επέκταση από έμπειρους προγραμματιστές. Η γλώσσα μπορεί να επεκταθεί μέσω των βιβλιοθηκών C ++ και οι ενδιαφερόμενοι που θέλουν να κατανοήσουν τις τεχνικές λεπτομέρειες μπορούν να μεταπηδήσουν από το Arduino στη γλώσσα προγραμματισμού AVR-C στην οποία και βασίζεται. Ομοίως, μπορεί να προστεθεί και ο κώδικας AVR-C απευθείας στα προγράμματα Arduino, αν αυτό είναι επιθυμητό.
- Ανοιχτή σχεδίαση υλισμικού και επεκτασιμότητα Τα σχέδια των πλακετών του Arduino δημοσιεύτηκαν με άδεια δημιουργού κοινής χρήσης. Έτσι οι

έμπειροι σχεδιαστές κυκλωμάτων μπορούν να δημιουργήσουν τη δική τους έκδοση της πλακέτας-μονάδας επεξεργασίας, να την επεκτείνουν και να την βελτιώσουν. Ακόμη και σχετικά άπειροι χρήστες μπορούν να δημιουργήσουν την έκδοση breadboard του ανοικτού σχεδίου για να καταλάβουν πώς λειτουργεί και να εξοικονομήσουν χρήματα.

#### 2.3 Προϊόντα Arduino

#### 2.3.1 Επίπεδο εισόδου

Σε ό,τι αφορά την χρησιμοποίηση Arduino σε επίπεδο entry-level, το σύστημα είναι εύκολο στη χρήση και να υποστηρίξει τις πρώτες παραγωγικές προσπάθειες. Αυτές οι πλακέτες είναι οι καλύτερες για την εκκίνηση της εκμάθησης και του παιχνιδιού με τα ηλεκτρονικά και την κωδικοποίηση. Το StarterKit περιλαμβάνει ένα βιβλίο με 15 μαθήματα που προσφέρουν καθοδήγηση σε βασικές ενότητες λειτουργίας [14].



Σχήμα 8. Διάφορα συστήματα Arduino [14].



#### 2.3.2 Ενισχυμένα χαρακτηριστικά

Για τη δημιουργία πιο σύνθετων εφαρμογών και με πιο γρήγορες λειτουργίες αρκεί να χρησιμοποιηθούν πλακέτες με πρόσθετα χαρακτηριστικά που φαίνονται παρακάτω [14].



Σχήμα 9. Ενισχυμένα χαρακτηριστικά Arduino [14]



#### 2.3.3 Internet of Things (IoT)

To Internet of Things (IoT) είναι το δίκτυο των φυσικών συσκευών, των οχημάτων, των οικιακών συσκευών και άλλων αντικειμένων τα οποία έχουν ενσωματωμένα σε αυτά, ηλεκτρονικά, λογισμικό, αισθητήρες και άλλες περιφερειακές συσκευές που επιτρέπουν τη διασυνδεσιμότητα και την ανταλλαγή δεδομένων, κάνοντας εφικτή την πιο άμεση ενσωμάτωση του «φυσικού κόσμου» σε σύστημα του «ψηφιακού κόσμου», με αποτέλεσμα τη βελτίωσης της αποτελεσματικότητας, τις οικονομικές απολαβές και την επίτευξη αυτοματισμού [15].



Σχήμα 10. Διασυνδεσιμότητα [14]

Με την βοήθεια των Ethernet shields και άλλων επεκτάσιμων μικροπλακετών μπορεί να συνδεθεί η κύρια πλακέτα με το διαδίκτυο και με αυτό τον τρόπο να επεκτείνεται η δημιουργικότητα λόγω των δυνατοτήτων του παγκόσμιου ιστού [14].



Σχήμα 11. Περιφερειακές πλακέτες συστήματος [14].



## 3. Εγκατάσταση του IDE του Arduino Desktop

# 3.1 Εγκατάσταση του λογισμικού Arduino (IDE) σε υπολογιστές με Windows

Η απόκτηση της πιο πρόσφατης έκδοσης μπορεί να γίνει από την επίσημη ιστοσελίδα. Μπορεί να επιλεχθεί μεταξύ του πακέτου Installer (.exe) και του συμπιεσμένου αρχείου Zip. Προτείνεται να χρησιμοποιηθεί η πρώτη, η οποία εγκαθιστά άμεσα όλα όσα χαπαιτούνται για να χρησιμοποιηθεί το λογισμικό Arduino (IDE) μέσα από την επίσημη ιστοσελίδα του Arduino, συμπεριλαμβανομένους τους οδηγούς του λογισμικού. Με το συμπιεσμένο πακέτο Zip μπορούν να εγκατασταθούν οι οδηγοί των προγραμμάτων με μη αυτόματο τρόπο. Το αρχείο συμπιεσμένο Zip είναι επίσης χρήσιμο εάν χρειαστεί να δημιουργηθεί μια φορητή εγκατάσταση [16].

Όταν ολοκληρωθεί η λήψη, συνεχίζεται η εγκατάσταση του λογισμικού και ειδικότερα εκτελείται η διαδικασία εγκατάστασης του οδηγού του προγράμματος όταν λάβει την κατάλληλη έγκριση από το λειτουργικό σύστημα [16].

💿 Arduino Setup: Installation O	ptions	_		$\times$		
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.						
Select components to install: Install Arduino software Install USB driver Create Start Menu shortcut Create Desktop shortcut Associate .ino files						
Space required: 392.7MB						
Cancel Nullsoft Install S	iystem v2.46	< <u>B</u> ack	Next	t>		

Σχήμα 12. Εικόνα οδηγού εγκατάστασης Windows και επιλογή των στοιχείων που θα εγκατασταθούν [16].

🥺 Arduino Setup: Installation Folder 🧼 —	□ ×
Setup will install Arduino in the following folder. To install folder, click Browse and select another folder. Click Insta installation.	in a different Il to start the
Destination Folder	
C:\Program Files (x86)\Arduino\	Browse
Space required: 392.7MB	
Space available: 24.6GB	
Cancel Nullsoft Install System v2.46 < Back	Install

Σχήμα 13. Εικόνα οδηγού εγκατάστασης Windows και επιλογή του καταλόγου εγκατάστασης (συνήθως η προεπιλεγμένη) [16].

💿 Arduino Setup: Installing	_		$\times$
Extract: c++.exe			
Show details			
Cancel Nullsoft Install System v2.46	< <u>B</u> ack	⊆lose	

Σχήμα 14. Εικόνα οδηγού εγκατάστασης Windows - Κατά τη διαδικασία εγκατάστασης θα εξαχθούν και θα εγκατασταθούν όλα τα απαιτούμενα αρχεία για τη σωστή εκτέλεση του λογισμικό Arduino (IDE) [16].

#### 3.2 Εγκατάσταση του λογισμικού Arduino (IDE) στο OS Χ

Επίσης, η απόκτηση της ενημερωμένης έκδοσης μπορεί να γίνει από την επίσημη σελίδα του λογισμικού. Το αρχείο είναι σε μορφή συμπίεσης Zip. Αν χρησιμοποιηθεί το Safari αυτό θα αποσυμπιεστεί αυτόματα. Εάν χρησιμοποιηθεί διαφορετικό πρόγραμμα περιήγησης, ίσως χρειαστεί να γίνει εξαγωγή με μη αυτόματο τρόπο [17].

	0	Downloads		
$\langle \langle \rangle \rangle$			Q Search	
Favorites	Name	Size	Kind Date Added	ł
All My Files	© Arduino	426,6 MB	Application Today 15:0	4
🛆 iCloud Drive				
Applications				
Desktop				
Documents				
O Downloads				
Devices				
Macintosh HD				
Tags				
e Red				
😑 Orange				
Yellow				
- Create				

Σχήμα 15. Εικόνα οδηγού εγκατάστασης OS X [17].

< >	Appli 	cations 🔹 🗸 🚹 🔲	Q Search
Favorites		$\overline{\circ}$	Ĩ
Applications	App Store	Arduino	Automator
Desktop Documents Downloads		30L 17	3
Devices	Calculator	Calendar	Chess
Shared		÷ ° °	Aa
Fags ● Red	Contacts	Dashboard	Dictionary
	(max)		

Σχήμα 16. Εικόνα οδηγού εγκατάστασης OS X: Αντιγραφή της εφαρμογής Arduino στο φάκελο "Εφαρμογές" (ή οπουδήποτε αλλού στην αποθηκευτική μονάδα του υπολογιστού) [17].

#### 3.3 Εγκατάσταση του λογισμικού Arduino (IDE) στο Linux

Η έκδοση για Linux του λογισμικού Arduino (IDE) αποτελεί ένα πακέτο που δεν απαιτεί ειδική διαδικασία εγκατάστασης για τις διάφορες διανομές που διατίθενται για Linux. Η μόνη σχετική προσοχή πρέπει να δοθεί στην επιθυμητή έκδοση εγκατάστασης 32 ή 64 bit του λειτουργικού συστήματος [18].

Όπως και στα Windows, παρομοίως και στο OS X λειτουργικό σύστημα, η απόκτηση της πιο πρόσφατης έκδοση μπορεί να γίνει από την επίσημη σελίδα. Μπορεί να επιλεχθεί μια από τις διατιθέμενες εκδόσεις 32, 64 και ARM. Είναι πολύ σημαντικό να επιλεχθεί η σωστή έκδοση για διανομή Linux [18].

🔞 💿 Opening arduino-1.6.10-linux32.ta	ar.xz	
You have chosen to open:		
arduino-1.6.10-linux32.tar.xz		
which is: XZ archive (94.9 MB) from: https://downloads.arduino.cc		
Would you like to save this file?		
	Cancel	Save File

Σχήμα 17. Εικόνα οδηγού εγκατάστασης Linux [18].

🕲 🖨 💿 arduino-1.6.10-li Extract +	nux32.ta	ər.xz		۹ ≡
< > 🟠 Location:	<b>i</b> /			
Name		Size	Туре	Modified
arduino-1.6.10		440.6 MB	Folder	26 July 2016, 14:16
Ext Please	racting x32.tar	the files fi .xz"	rom "arduin	10-1.6.10-
	_	Cancel		

Σχήμα 18. Εικόνα οδηγού εγκατάστασης Linux: Το αρχείο είναι συμπιεσμένο και θα πρέπει να γίνει εξαγωγή σε έναν κατάλληλο φάκελο, υπενθυμίζοντας την εκτέλεσή του από εκεί.[18] Εν συνεχεία των προαναφερθέντων, ανοίγοντας τον φάκελου arduino-1.6.x που μόλις δημιουργήθηκε από τη διαδικασία εξαγωγής, είναι εμφανές το αρχείο install.sh. Πατώντας δεξί κλικ πάνω του, μετά, Επιλογή και Εκτέλεση στο τερματικό από το μενού, η διαδικασία εγκατάστασης θα ολοκληρωθεί γρήγορα και θα πρέπει να εμφανιστεί ένα νέο εικονίδιο στην επιφάνεια εργασίας [18].

Αν δεν βρεθεί η επιλογή εκτέλεσης της δέσμης ενεργειών από το μενού, θα πρέπει να ανοιχτεί ένα νέο παράθυρο του τερματικού και να μετακινηθεί στο φάκελο arduino-1.6.x. Πληκτρολογώντας την εντολή ./install.sh θα υπάρξει ένας χρόνος αναμονής για την ολοκλήρωση της διαδικασίας. Στη συνέχεια θα εμφανιστεί ένα νέο εικονίδιο στην επιφάνεια εργασίας [18].



Σχήμα 19. Εικόνα οδηγού εγκατάστασης Linux [18].



ΤΕΙ ΗΠΕΙΡΟΥ 2018



### 4. Κώδικας

#### 4.1 Γλώσσα Προγραμματισμού

Η γλώσσα Arduino είναι απλώς ένα σύνολο λειτουργιών C / C ++ που μπορούν να καλούνται από τον κώδικα προγραμματισμού. Στη συνέχεια εκτελούνται διάφορες αλλαγές (π.χ. αυτόματη δημιουργία πρωτοτύπων λειτουργίας) και στη συνέχεια το πρόγραμμα διαβιβάζεται απευθείας σε έναν μεταγλωττιστή C / C ++ (avr-g ++). Όλες οι τυπικές δομές C και C ++ που υποστηρίζονται από το avr-g ++ θα πρέπει να λειτουργούν στο Arduino [19].



Σχήμα 20. Λογότυπο γλώσσας προγραμματισμού C++ [20].

#### 4.2 Γενική εικόνα

Προκειμένου να προγραμματιστεί η πλακέτα Arduino, πρέπει να ακολουθηθεί μία διαδικασία. Καταρχάς, το περιβάλλον Arduino εκτελεί προεπεξεργασία ώστε να εκτελέσει το πρόγραμμα C++. Ακολούθως αυτό διαβιβάζεται σε έναν μεταγλωττιστή (avr-gcc), ο οποίος μετατρέπει τον αναγνώσιμο κώδικα (από τον προγραμματιστή) σε οδηγίες αναγνώσιμες από μηχανή (ή αρχεία αντικειμένων). Εν συνέχεια, ο κώδικας συνδυάζεται με βιβλιοθήκες του Arduino που παρέχουν βασικές λειτουργίες όπως το digitalWrite () ή το Serial.print (). Το αποτέλεσμα είναι ένα ενιαίο αρχείο Intel hex, το οποίο περιέχει τα bytes που πρέπει να εγγραφούν στη μνήμη προγραμμάτων του chip που βρίσκεται πάνω στην πλακέτα του Arduino. Αυτό το αρχείο φορτώνεται στην πλακέτα ως εξής: Μεταδίδεται μέσω USB ή σειριακής σύνδεσης χρησιμοποιώντας τον bootloader που βρίσκεται ήδη στο τσιπ είτε με εξωτερική μονάδα προγραμματισμού [19].



Σχήμα 21. Προσχεδιασμός [21].

#### 4.3 Προεπεξεργασία

Το περιβάλλον Arduino εκτελεί διάφορες αλλαγές στον κώδικα προτού τον μεταβιβάσει στον μεταγλωττιστή avr-gcc:[19].

Όλα τα αρχεία .ino στο φάκελο προγράμματος (εμφανίζονται στο IDE ως καρτέλες χωρίς επέκταση) συνδέονται μεταξύ τους ξεκινώντας από το αρχείο που αντιστοιχεί στο όνομα του φακέλου που ακολουθείται από τα υπόλοιπα με αλφαβητική σειρά καθώς και η επέκταση .cpp προστίθεται στο όνομα αρχείου [19].

Εάν δεν υπάρχει ήδη, το #include <Arduino.h> τότε εκείνο προστίθεται στο σχέδιο. Αυτό το αρχείο κεφαλίδας (που βρίσκεται στον κεντρικό φάκελο για την τρέχουσα επιλεγμένη πλακέτα) έχει άμεση σχέση με τον βασικό πυρήνα Arduino [19].

Εν συνεχεία, δημιουργούνται πρωτότυπα αρχεία .ino για όλους τους ορισμούς των λειτουργιών που δεν έχουν ήδη συσχετισμένα αρχεία. Σε ορισμένες σπάνιες περιπτώσεις η παραγωγή πρωτοτύπου μπορεί να αποτύχει για ορισμένες λειτουργίες. Για εύρεση λύσης πρέπει να διεξαχθεί επεξεργασία των αρχείων συμπληρώνοντας ή διορθώνοντας τα πρωτότυπα για αυτές τις λειτουργίες [19]. Οι #line οδηγίες έχουν προστεθεί για να κάνουν τα μηνύματα προειδοποίησης ή λάθους να αντιστοιχούν στην αρχική διάταξη κώδικα-σχεδιασμού [19].

Δεν γίνεται προ-επεξεργασία στα αρχεία ενός σχεδιασμού (project) με οποιαδήποτε επέκταση εκτός από εκείνη σε .ino. Επιπλέον, τα αρχεία .h δεν είναι αυτομάτως #included από το κύριο αρχείο σχεδίων. Επίσης, αν πρέπει να καλεστούν λειτουργίες που έχουν οριστεί σε ένα αρχείο .c από ένα αρχείο .cpp (όπως εκείνο που δημιουργείται από το σχέδιο του προγραμματιστού), θα πρέπει να ολοκληρωθούν οι δηλώσεις του σε ένα 'εξωτερικό" C "{} 'μπλοκ που ορίζεται μόνο μέσα στα αρχεία C++ [19].



Σχήμα 22. Από τον κώδικα στη πλακέτα [22].

ΤΕΙ ΗΠΕΙΡΟΥ 2018



# Ζεύξη 5.1 MikroTik

Η MikroTik είναι μια Λετονική εταιρεία που ιδρύθηκε το 1996 για την ανάπτυξη δρομολογητών και ασύρματων συστημάτων ISP. Η MikroTik παρέχει τώρα υλικό και λογισμικό για σύνδεση στο Internet στις περισσότερες χώρες του κόσμου. Σημαντικό βήμα για την επιτυχία της ήταν η διαρκής ενασχόληση της, από το 1997, για την ανάπτυξη ενός λειτουργικού συστήματος του RouterOS που έχει εφαρμογή σε συσκευές δικτύου (routers, switches κ.λπ). Το RouterOS είναι κλειστό λειτουργικό σύστημα που όμως βασίζεται στον πυρήνα του Linux και παρέχει εκτεταμένη σταθερότητα, ελέγχους και ευελιξία για όλα τα είδη διεπαφών δεδομένων και δρομολόγησης. Χρησιμοποιήθηκε αρχικά πάνω σε τυποποιημένο υλικό PC, μέχρι που το 2002 η Mikrotik αποφάσισε να κατασκευάσει το δικό της υλικό που είναι η σειρά προϊόντων RouterBOARD. Παράλληλα, με την άνοδο της ζήτησης των προϊόντων ασύρματης επικοινωνίας, η Mikrotik έκανε μια δυναμική είσοδο στην αγορά αυτή, παρουσιάζοντας προϊόντα ασύρματης δικτύωσης με διαφόρους τύπους πρόσβασης (WiFi WiMAX, GSM, LTE) βασισμένων σε ολοκληρωμένα κυκλώματα γνωστών κατασκευαστών όπως η Atheros [23].



Σχήμα 23. Λογότυπο εταιρείας MikroTik [23].



#### 5.2 cAP lite

Το cAP lite είναι μια μικρή σε διαστάσεις συσκευή σημείου πρόσβασης, ιδανική για δημόσιες τοποθεσίες και επιχειρήσεις φιλοξενίας όπως ξενοδοχεία, αεροδρόμια, εμπορικά κέντρα κλπ. Το cAP lite υποστηρίζει 802.11b/g/n και μπορεί να τροφοδοτείται με ρεύμα μέσω PoE (Power over Ethernet). Συνεργάζεται με το MikroTik CAPsMAN (ελεγχόμενος διαχειριστής συστήματος AP), που επιτρέπει τον έλεγχο όλων των συσκευών από μια κεντρική τοποθεσία. Το cAP lite μπορεί επίσης να χρησιμοποιηθεί ως αυτόνομο σημείο πρόσβασης με δυνατότητες του RouterOS [24].



Σχήμα 24. Εμφάνιση κεραίας cAP [24].

#### 5.3 RB260GSP

To RB260GSP είναι ένας μικρός σε διαστάσεις προγραμματιζόμενος μεταγωγέας δικτύου (network switch), ο οποίος διαθέτει έξοδο ισχύος στις θύρες ethernet του. Οι θύρες 2-5 παρέχουν τάση τροφοδοσίας στις συσκευές που είναι συνδεδεμένες σε αυτές ακολουθώντας το πρότυπο παθητικού PoE (Power over Ethernet) Η συσκευή διαθέτει λειτουργικό σύστημα σχεδιασμένο ειδικά για μοντέλα MikroTik Switch – SwOS [25].



*Σχήμα 25. Εμφάνιση Switch RB260GSP [25].* 

#### 5.4 Γεφύρωση (Bridging)

Στη μηχανική των δικτύων επικοινωνίας διακρίνουμε δύο μηχανισμούς που εξασφαλίζουν τη μεταφορά της πληροφορίας από το ένα άκρο στο άλλο. Εφόσον οι συσκευές που διασυνδέονται είναι μέλη του ίδιου δικτύου τότε ο μηχανισμός που χρησιμοποιούμε ονομάζεται γεφύρωση. Σε αντίθετη περίπτωση, εφόσον οι συσκευές που διασυνδέονται είναι μέλη διαφορετικών δικτύων χρησιμοποιούμε τον όρο "δρομολόγηση" για να υποδηλώσουμε την μεταξύ τους επικοινωνία [26].

Αναλυτικά, στο μοντέλο OSI, η γεφύρωση γίνεται στο στρώμα ζεύξης δεδομένων (layer 2). Εάν ένα ή περισσότερα τμήματα του γεφυρωμένου δικτύου είναι ασύρματα, η συσκευή είναι γνωστή ως ασύρματη γέφυρα και η λειτουργία ως ασύρματη γεφύρωση. Αντίθετα η δρομολόγηση στο μοντέλο OSI βρίσκεται στο στρώμα δικτύου (layer 3) [26].



Σχήμα 27. Μοντελοποίηση Γεφύρωσης [27].

ΤΕΙ ΗΠΕΙΡΟΥ 2018



# Αναλυτική περιγραφή της υλοποίησης του ασύρματου συστήματος

Η παρούσα εργασία ασχολείται με σύστημα Arduino το οποίο λαμβάνει μετρήσεις υγρασίας και θερμοκρασίας με τη βοήθεια ενός αισθητήρα (ο οποίος ανιχνεύει τις συνθήκες περιβάλλοντος) και τις αποθηκεύει στη μνήμη της πλακέτας. Έπειτα αποστέλλονται τα πακέτα με DHCP (Server). Αυτό γίνεται με τη χρήση Ethernet Shield όπου συνδέει τη πλακέτα με το LAN δίκτυο μέσω ασύρματης ζεύξης δυο κεραιών περνώντας πρώτα από δυο Switch (οπού συνδέεται πάνω τους η κάθε κεραία ξεχωριστά) και καταλήγουν σε άλλο Arduino, λαμβάνοντας τα δεδομένα επίσης με DHCP (Client). Τα δεδομένα τα οποία αποστέλλονται από το πρώτο Arduino στο δεύτερο εμφανίζονται στην οθόνη συνδέοντας τη πλακέτα του δεύτερου Arduino με το πρόγραμμα Arduino IDE, το οποίο είναι εγκατεστημένο σε υπολογιστή. Επιπροσθέτως, τα αποτελέσματα του πρώτου Arduino δεν τα διαβάζουμε διαμέσω μόνο του δεύτερου Arduino αλλά και μέσω σύνδεσης με κινητό ή υπολογιστή στο δίκτυο κάλυψης της κεραίας εκπομπής πληκτρολογώντας στον Browser την IP διεύθυνση του WebServer.

	COM4 (Arc	tuino/Genuino Unaj
client Sµέva c	Γίνεται σύνδεση HTTP/1.1 200 OK Content-Type: text/html Connection: close	
iet	Refresh: 5	
)18	<pre><html> ****Data Transmission &amp; Reception Using Microcomputers**** *Creduction project by Direct Johny DegNet 15237, Son 2018*</html></pre>	Ş
19	<pre>*Graduation project by Pirra Johny Regult: 1923/ Sep. 2018* **** Dr.Spiros Chronopoulos : supervising proffessor ***** Arduino WebService running at: 192.168.168.5   Temperature &amp; Humidity Update No: 124  </pre>	
:.h>	Current Temperature is: 27.00C Humidity is: 54.00% Aυτόματη κύλιση Show timestamp	ntml>
lent; / Connec ig post	// Ορισμός της μεταβλητής που θα υλοπο.ήσει την επικοινωνία. :tionTime = 0; // Ο χρόνος που έγινε η τελευταία σύνδσεη στον :ingInterval = 10 * 1000; // χρόνος επανάληψης, σε milliseconds	WebServer, or millisecond

Σχήμα 28. Αποτελέσματα Web Client.



Σχήμα 29. Ολοκληρωμένο σύστημα.

#### 6.1 Εξυπηρετητής ιστού (Web Server)

Στη συνέχεια παρουσιάζεται η υλοποίηση ενός Web Server που μεταδίδει τιμές Θερμοκρασίας και Υγρασίας. Καταρχάς, γίνεται αρχικοποίηση της υπηρεσίας Web Server στην καθορισμένη θύρα (τυπική θύρα για μεταφορές HTTP είναι η TCP 80). Στη συνέχεια γίνεται αρχικοποίηση της σειριακής επικοινωνίας για την προβολή της κονσόλας όπου έχουμε και αναμονή μέχρι να γίνει διαθέσιμη η σειριακή θύρα. Ύστερα εκτελείται έλεγχος από το σύστημα για την παρουσία του κατάλληλου υλικού. Εάν δεν βρεθεί κατάλληλο υλισμικό, τότε έχουμε ατέρμων βρόγχο αναμονής. Εάν βρεθεί το υλισμικό, θέλουμε χρόνο περίπου ενός δευτερολέπτου για την πλήρη εκκίνησή του. Σχετικά με το δίκτυο, αρχικά, ορίζουμε την φυσική διεύθυνση (MAC address) του μικροελεγκτή. Επίσης ορίζουμε μια διεύθυνση IP από την περιοχή διευθύνσεων του τοπικού μας δικτύου για την περίπτωση που δεν λειτουργεί σε αυτό DHCP Server καθώς και διεύθυνση DNS. Έπειτα, γίνεται η εκκίνηση της σύνδεσης Ethernet και της υπηρεσίας Web. Εν συνεχεία αποστέλλουμε το header τμήμα μιας τυπικής απόκρισης πρωτόκολλου http. Επίσης, η σύνδεση θα διακοπεί στην ολοκλήρωση της απάντησης και γίνεται αυτόματη ανανέωση της σελίδας κάθε 5 δευτερόλεπτα.

Τα δεδομένα που εμφανίζονται στην οθόνη είναι ένα αρχικό κείμενο με στοιχεία σχετικά με την σύνδεση και σύντομες πληροφορίες επικεφαλίδας. Στη συνέχεια οι τιμές θερμοκρασίας και υγρασίας στην οθόνη του απομακρυσμένου πελάτη.

Τέλος, έχουμε ακρόαση για εισερχόμενες συνδέσεις και κάθε αίτηση http τερματίζεται με κενή γραμμή. Να επισημάνουμε ότι η λήψη του χαρακτήρα '\n' σηματοδοτεί το τέλος της τρέχουσας γραμμής. Εφόσον δε η επόμενη γραμμή είναι κενή, η αίτηση http έχει ολοκληρωθεί και μπορούμε να στείλουμε την απάντηση. Ύστερα έχουμε έναρξη νέας γραμμής και έναν χαρακτήρα στην τρέχουσα γραμμή. Μετέπειτα έχουμε καθυστέρηση μέχρι να ολοκληρωθεί η λήψη από τον πελάτη και τερματισμό της σύνδεσης.



Σχήμα 30. Σύστημα Web Server.



#### 6.2 Πελάτης ιστού (Web Client)

Σε αυτή την ενότητα παρουσιάζεται ο Web Client που συνδέεται με τον Web Server και λαμβάνει δεδομένα σε τακτά χρονικά διαστήματα. Αρχικά, έχουμε τον ορισμό της μεταβλητής που θα υλοποιήσει την επικοινωνία, το χρόνο που έγινε η τελευταία σύνδεση στον WebServer σε milliseconds και το χρόνο επανάληψης σε milliseconds. Έπειτα έχουμε αναμονή μέχρι να γίνει διαθέσιμη η σειριακή θύρα.

Σχετικά με τη σύνδεση του δικτύου, αρχικά ορίζουμε την φυσική διεύθυνση (MAC address) για τον μικροελεγκτή, μετά ορίζουμε μια διεύθυνση IP από την περιοχή διευθύνσεων του τοπικού μας δικτύου για την περίπτωση που δεν λειτουργεί σε αυτό DHCP Web Server καθώς και τη διεύθυνση DNS. Στη συνέχεια γίνεται έλεγχος για την παρουσία του κατάλληλου υλισμικού και στην περίπτωση που δεν βρεθεί κατάλληλο υλισμικό και στην περίπτωση που δεν βρεθεί κατάλληλο υλισμικό θα υφίσταται ατέρμων βρόχος αναμονής. Στην περίπτωση αποτυχίας του DHCP κάνουμε χρήση στατικών παραμέτρων και τελικώς χρειαζόμαστε περίπου χρόνο ενός δευτερολέπτου για την πλήρη εκκίνηση του υλικού. Εν συνεχεία αρχικοποιούμε την σειριακή επικοινωνία για την δυνατότητα προβολής στην κονσόλα και μετά έχουμε την εκκίνηση της σύνδεσης Ethernet. Εν τέλει επαναλαμβάνουμε την http αίτηση μόλις λήξει ο χρόνος επανάληψης. Η μέθοδος αυτή καθιστά τη σύνδεση HTTP με τον Web Server καθώς και απαιτείται να συμβαδίζει με τον Host στην HTTP GET αίτηση. Τελικά, κάνουμε έλεγχο για επιτυχή ή μη επιτυχή σύνδεση, διαμορφώνουμε μια τυπική HTTP GET αίτηση και καταγράφουμε το χρόνο που έγινε η σύνδεση. Στη περίπτωση που αποτύχει η σύνδεση εμφανίζεται μήνυμα προειδοποίησης.



Σχήμα 31. Σύστημα Web Client



#### 6.3 Αισθητήρας υγρασίας και θερμοκρασίας

Ο αισθητήρας υγρασίας και θερμοκρασίας DHT11 καθιστά αρκετά εύκολο να συμπεριληφθεί η μέτρηση δεδομένων υγρασίας και θερμοκρασίας σε DIY ηλεκτρονικά έργα. Είναι ιδανικό για απομακρυσμένους μετεωρολογικούς σταθμούς, οικιακά συστήματα περιβαλλοντικού ελέγχου και συστήματα παρακολούθησης αγροκτήματος ή κήπου [28].





Σχήμα 32. Μοντέλο Αισθητήρα [28].

#### 6.4 **Ethernet Shield**

Το W5100 είναι ένα ευέλικτο τσιπ μίας διεπαφής δικτύου με ενσωματωμένο ελεγκτή Ethernet 10 / 100 Mbs. Χρησιμοποιείται κυρίως σε ένα ενσωματωμένο σύστημα, υψηλής σταθερότητας, απόδοσης και χαμηλού κόστους. Το W5100 επιτρέπει τη σύνδεση στο Internet χωρίς σύστημα λειτουργίας. Είναι επίσης συμβατό με ΙΕΕΕ802.3 10BASE-T και 802.3u 100BASE-TX. Το W5100 ενσωματώνει μια στοίβα πρωτοκόλλου TCP / IP πλήρους υλικού, το επίπεδο ελέγχου MAC (MAC) και το φυσικό επίπεδο (PHY). Η στοίβα πρωτοκόλλου TCP / IP υποστηρίζει τα ακόλουθα πρωτόκολλα: TCP, UDP, IPV4, ICMP, ARP, IGMP και PPoE, τα οποία έχουν δοκιμαστεί από την αγορά εδώ και χρόνια σε πολλά πεδία εφαρμογής. Επιπλέον, το

Signal Vcc (+)

Not used

Ground (-



W5100 ενσωματώνει μνήμη 16KB για τη μετάδοση δεδομένων. Για το W5100, δεν χρειάζεται να εξεταστεί ο έλεγχος του Ethernet. Αυτό που απαιτείται είναι ο προγραμματισμός υποδοχής. Το W5100 έχει τρεις διεπαφές: απευθείας παράλληλο δίαυλο, έμμεσο παράλληλο δίαυλο και δίαυλο SPI. Είναι εύκολο να συνδεθεί με ένα MCU, όπως και να έχει πρόσβαση σε εξωτερική μνήμη. Εδώ χρησιμοποιούμε τη βιβλιοθήκη Ethernet για την εφαρμογή του W5100 στο σύστημά μας.



Σχήμα 33. Εμφάνιση Ethernet Shield [29].

## 7. Συμπεράσματα - Μελλοντικοί Στόχοι

Κατά την υλοποίηση της παρούσης εργασίας, αναφερθήκαμε στην επικοινωνία ενός Web Server με άλλους Web Clients. Ο κάθε πελάτης έχει τη δυνατότητα να βλέπει τα πακέτα που στέλνει ο Server συνδέοντας τη συσκευή του στο δίκτυο και πληκτρολογώντας στο Browser την διεύθυνση του Server. Για λόγους απλότητας και για να επιτευχθεί η ταχύτερη δυνατή λειτουργία της επικοινωνίας χρησιμοποιήθηκε η μέθοδος της γεφύρωσης μεταξύ κεραιών. Μέσω αυτής της υλοποίησης κατανοήθηκε η δυνατότητα του προτεινόμενου συστήματος να δρα αυτόνομα ακόμα και σε ευρύτερα δίκτυα όπως εκείνο του διαδικτύου. Τέλος, σε περίπτωση πρόσθεσης και άλλων αισθητήρων μπορούμε να έχουμε τη διαχείριση μεγάλου όγκου δεδομένων (Big Data), το οποίο και απαιτεί ιδιαίτερη εξέλιξη του εν λόγω συστήματος. Επίσης, το σύστημα αυτό μπορεί να ενσωματωθεί και σε συστήματα εξελιγμένων ασύρματων επικοινωνιών [30], εξελιγμένων ανιχνευτών [31-32] που μπορεί να βασίζονται σε Turbo κώδικες [33-34] και OFDM μεταδόσεις [35-37]. Επίσης, με την κατάλληλη επεξεργασία σήματος θα μπορούσε μέχρι και να κάνει προβλέψεις βασιζόμενο στις προηγούμενες μετρήσειςκαταστάσεις (π.χ. περιβαλλοντικές μετρήσεις).

#### 8. ΒΙΒΛΙΟΓΡΑΦΙΑ (REFERENCES)

1. En.wikipedia.org. (2018). *Microcontroller*. Available at: <u>https://en.wikipedia.org/wiki/Microcontroller</u> [Accessed 1 Sep . 2018].

2. Industrialtrainingjalandhar.net. (2018). *pic microcontroller training in jalandhar*. [online] Available at: <a href="https://industrialtrainingjalandhar.net/embedded-systems/pic-microcontroller-training-in-jalandhar">https://industrialtrainingjalandhar.net/embedded-systems/pic-microcontroller-training-in-jalandhar</a> [Accessed 2 Sep. 2018].

3. En.wikipedia.org. (2018). *Raspberry Pi*. [online] Available at:

https://en.wikipedia.org/wiki/Raspberry Pi [Accessed 1 Sep. 2018].

4. Raspberry Pi. (2018). *Products Archive - Raspberry Pi*. [online] Available at:

https://www.raspberrypi.org/products/ [Accessed 2 Sep. 2018].

5. En.wikipedia.org. (2018). *BeagleBoard*. [online] Available at:

https://en.wikipedia.org/wiki/BeagleBoard [Accessed 1 Sep. 2018].

Inc., P. (2018). BEAGLEBOARD X15 Precision Technology Inc. DigiKey. [online] Digikey.jp. Available
 at: https://www.digikey.jp/product-detail/ja/precision-technology-inc/BEAGLEBOARD-X15/1777-1001 ND/6834154 [Accessed 2 Sep. 2018].

7. En.wikipedia.org. (2018). *Banana Pi*. [online] Available at: <u>https://en.wikipedia.org/wiki/Banana Pi</u> [Accessed 1 Sep. 2018].

8. (webmaster@reichelt.de), r. (2018). BANANA PI - Banana Pi, 1 GHz Dual Core, 1 GB DDR3, SATA.
[online] Elektronik und Technik bei reichelt elektronik günstig bestellen. Available at:

https://www.reichelt.de/banana-pi-1-ghz-dual-core-1-gb-ddr3-sata-banana-pi-p144326.html [Accessed 2 Sep. 2018].

9.En.wikipedia.org. (2018). *Intel Galileo*. [online] Available at: <u>https://en.wikipedia.org/wiki/Intel\_Galileo</u> [Accessed 1 Sep. 2018].

Ürün, Ü. and TAŞÇENE, O. (2018). Buy Intel® Galileo Gen 2 with cheap price. [online]
 https://www.robotistan.com/. Available at: https://www.robotistan.com/intel-galileo-gen-2 [Accessed 2 Sep.
 2018].

11. Arduino.cc. (2018). *Arduino - Introduction*. Available at:

https://www.arduino.cc/en/Guide/Introduction [Accessed 1 Sep. 2018].

12. Pololu.com. (2018). Pololu - Arduino Leonardo. [online] Available at:

https://www.pololu.com/product/2192 [Accessed 2 Sep. 2018].

13. Toebehoren, M. (2018). *bol.com | Starter Kit voor Arduino & Raspberry Pi*. [online] Bol.com. Available at: https://www.bol.com/nl/p/starter-kit-voor-arduino-raspberry-pi/920000080287231/ [Accessed 2 Sep. 2018].

14. Arduino.cc. (2018). *Arduino - Products*. [online] Available at:

https://www.arduino.cc/en/Main/Products [Accessed 1 Sep. 2018].

15. En.wikipedia.org. (2018). *Internet of things*. [online] Available at:

https://en.wikipedia.org/wiki/Internet\_of\_things [Accessed 3 Sep. 2018].

16. Arduino.cc. (2018). Arduino - Windows. [online] Available at:

https://www.arduino.cc/en/Guide/Windows [Accessed 1 Sep. 2018].

17. Arduino.cc. (2018). *Arduino - MacOSX*. [online] Available at: https://www.arduino.cc/en/Guide/MacOSX [Accessed 1 Sep. 2018].



 Arduino.cc. (2018). Arduino - Linux. [online] Available at: <u>https://www.arduino.cc/en/Guide/Linux</u> [Accessed 7 Sep. 2018].

19. GitHub. (2018). *arduino/Arduino*. [online] Available at: <u>https://github.com/arduino/Arduino/wiki/Build-</u> <u>Process</u> [Accessed 1 Sep. 2018].

20. Eduonix Blog. (2018). *Learn the basic Syntax in C++ Programming Language - Eduonix Blog*. [online] Available at: https://blog.eduonix.com/system-programming/learn-the-basic-syntax-in-c-programminglanguage/ [Accessed 2 Sep. 2018].

Towards Data Science. (2018). Data Preprocessing in Python – Towards Data Science. [online]
 Available at: https://towardsdatascience.com/data-preprocessing-in-python-6f04e6c2cb70 [Accessed 2 Sep.
 2018].

22. Sharetechnote.com. (2018). *Arduino - Arduino Uno (R3) - ShareTechnote*. [online] Available at: http://www.sharetechnote.com/html/Arduino Programming.html [Accessed 2 Sep. 2018].

23. Mikrotik.com. (2018). *MikroTik*. [online] Available at: https://mikrotik.com/aboutus [Accessed 1 Sep. 2018].

24. Mikrotik.com. (2018). *MikroTik*. [online] Available at: <u>https://mikrotik.com/product/RBcAPL-2nD-</u> <u>307</u> [Accessed 1 Sep. 2018].

Mikrotik.com. (2018). *MikroTik*. [online] Available at: <u>https://mikrotik.com/product/RB260GS</u>
 [Accessed 1 Sep. 2018].

26. En.wikipedia.org. (2018). Bridging (networking). Available at:

https://en.wikipedia.org/wiki/Bridging\_(networking) [Accessed 1 Sep. 2018].

27. Docs.oracle.com. (2018). Bridging Overview - Managing Oracle Solaris 11.1 Network Performance.
 [online] Available at: https://docs.oracle.com/cd/E26502\_01/html/E28993/rbridgesoverview.html [Accessed 2 Sep.2018].

28. Basics, C. (2018). How to Set Up the DHT11 Humidity Sensor on an Arduino. [online] Circuit Basics. Available at: http://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/ [Accessed 2 Sep. 2018].

29. Arduino.cc. (2018). Arduino - ArduinoEthernetShield. [online] Available at:

https://www.arduino.cc/en/Guide/ArduinoEthernetShield [Accessed 2 Sep. 2018].

30. Chronopoulos, S., Kosma, E., Tafiadis, D., Dimopoulos, D., Raptis, V., C. Karvounis, E., Angelidis, P. and Kostarakis, P. (2018). Reduced Ecological Footprints of Modern Facilities Introducing the Implementation of Advanced Wireless Technologies, and Human Resources' Benefits. *Communications and Network*, 10(01), pp.11-29. DOI: <u>10.4236/cn.2018.101002</u>

31. Votis, C., Tatsis, G., Christofilakis, V., Chronopoulos, S., Kostarakis, P., Tritakis, V. and Repapis, C. (2018). A new portable ELF Schumann resonance receiver: design and detailed analysis of the antenna and the analog front-end. *EURASIP Journal on Wireless Communications and Networking*, 2018(1). DOI:10.1186/s13638-018-1157-7

32. Christofilakis, V., Tatsis, G., T. Votis, C., K. Chronopoulos, S., Kostarakis, P., J. Lolis, C. and Bartzokas, A. (2018). Rainfall Measurements Due to Radio Frequency Signal Attenuation at 2 GHz. *Journal of Signal and Information Processing*, 09(03), pp.192-201. DOI: <u>10.4236/jsip.2018.93011</u>

33. Chronopoulos, S., Tatsis, G. and Kostarakis, P. (2011). Turbo Codes—A New PCCC Design. *Communications and Network*, 03(04), pp.229-234. DOI: <u>10.4236/cn.2011.34027</u>

Chronopoulos, S.K., Tatsis, G., Raptis, V. and Kostarakis, P. (2012) A Parallel Turbo Encoder Decoder Scheme. Proceedings of the 2nd Pan-Hellenic Conference on Electronics and Telecommunications,
 Thessaloniki, 16-18 March 2012.

35. Chronopoulos, S.K., Votis, C., Raptis, V., Tatsis, G. and Kostarakis, P. (2010) In Depth Analysis of Noise Effects in Orthogonal Frequency Division Multiplexing Systems, Utilising a Large Number of Subcarriers. AIP Conference Proceedings, 1203, 967-972. DOI: <u>10.1063/1.3322592</u>

36. Chronopoulos, S.K., Tatsis, G. and Kostarakis, P. (2012). Turbo Coded OFDM with Large Number of Subcarriers. *Journal of Signal and Information Processing*, 03(02), pp.161-168. DOI: <u>10.4236/jsip.2012.32021</u>

Chronopoulos, S., Christofilakis, V., Tatsis, G. and Kostarakis, P. (2015). Performance of Turbo
 Coded OFDM Under the Presence of Various Noise Types. *Wireless Personal Communications*, 87(4), pp.1319 1336. DOI: <u>10.1007/s11277-015-3055-1</u>



ΤΕΙ ΗΠΕΙΡΟΥ 2018



# ΠΑΡΑΡΤΗΜΑ ΠΗΓΑΙΟΥ ΚΩΔΙΚΑ Π1. Server

/\*

TH-Web Server

Ένας απλός web server που μεταδίδει τιμές Θερμοκρασίας και Υγρασίας.

Κύκλωμα:

Στοιχείο Ethernet

Αισθητήρας Θερμοκρασίας & Υγρασίας DHT11 στο όριο 7

Αργοστόλι 2/9/2018

Πίρρα Τζονι

\*/

#include <SPI.h>

#include <Ethernet.h>

#include <dht.h>

#define DHT11\_PIN 7

// Αρχικοποίηση της υπηρεσίας WebServer

// στην καθορισμένη θύρα (τυπική θύρα για μεταφορές ΗΤΤΡ είναι η TCP 80)



TEI HITEIPOY 2018

```
EthernetServer myServer(80);
```

```
unsigned long upd = 0;
```

void setupSerial()

{

```
// Αρχικοποίηση της σειριακής επικοινωνίας για την προβολή της κονσόλας
Serial.begin(9600);
while (!Serial) {
    ; // Αναμονή μέχρι να γίνει διαθέσιμη η σειριακή θύρα
  }
}
```

void setupEthernet(byte mac[], IPAddress \*LOCip, IPAddress \*DNSip)

{

```
Ethernet.begin(mac, *LOCip, *DNSip);
```

```
if (Ethernet.hardwareStatus() == EthernetNoHardware) // Έλεγχος για την παρουσία του κατάλληλου υλικού
```

{

```
Serial.println("Δεν βρέθηκε υλικό Ethernet :(");
```

while (true) {

delay(1); // Ατέρμων βρόχος αναμονής αφού δεν υπάρχει το κατάλληλο υλικό

}

}

```
if (Ethernet.linkStatus() == LinkOFF) {
```

Serial.println("Αποσυνδέθηκε το καλώδιο Ethernet.");

}

delay(1000); // 1sec χρόνος για την πλήρη εκκίνηση του υλικού.

}

void setup() {

byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xF9, 0xED}; // Αρχικά ορίζουμε την φυσική διεύθυνση (MAC address) για τον μικροελεγκτή.

IPAddress LANip(192, 168, 168, 5); // Επίσης ορίζουμε μια διεύθυνση IP από την περιοχή διευθύνσεων του τοπικού μας δικτύου

IPAddress DNSip(192, 168, 168, 1); // για την περίπτωση που δεν λειτουργεί σε αυτό DHCP server καθώς και διεύθυνση DNS.

#### setupSerial();

setupEthernet(mac, &LANip, &DNSip); // Εκκίνηση της σύνδεσης Ethernet

myServer.begin(); // Εκκίνηση της υπηρεσίας Web

Serial.println("WebServer Θερμοκρασίας/Υγρασίας");

Serial.println("(c) 2018 Πίρρα Τζονι");

Serial.print("Η υπηρεσία λειτουργεί στην διεύθυνση : ");

Serial.println(Ethernet.localIP());

#### }

int getDHTreadings(int pin, dht \*DHT)

#### {

```
int chk = DHT->read11(pin);
```

Serial.print("Η Θερμοκρασία είναι : ");

Serial.print(DHT->temperature);

Serial.println("C");

Serial.print("Η Υγρασία είναι : ");

```
Serial.print(DHT->humidity);
```

```
Serial.println("%");
```

```
return (chk);
```

```
}
```

void HttpPost(EthernetClient \*cl, dht \*DHT)

#### {

cl->println("HTTP/1.1 200 OK"); // Αποστέλλουμε το header τμήμα μιας τυπικής απόκρισης πρωτόκολλου http

cl->println("Content-Type: text/html");

cl->println("Connection: close"); // Η σύνδεση θα διακοπεί στην ολοκλήρωση της απάντησης

cl->println("Refresh: 5"); // Αυτόματη ανανέωση της σελίδας κάθε 5 sec

```
cl->println();
```

cl->println("<!DOCTYPE HTML>");

cl->println("<html>");

cl->println("\*\*\*\*Data Transmission & Reception Using Microcomputers\*\*\*\*");

cl->println("\*Graduation project by Pirra Johnnie RegNo: 15237 Sep. 2018\*");

cl->println("\*\*\*\* Dr.Spiros Chronopoulos : supervising professor \*\*\*\*\*");

cl->println("<br>");

cl->print("Arduino WebService running at: ");

cl->print(Ethernet.localIP());

cl->println(" <br /><br>");

cl->print("Temperature & Humidity Update No: ");

cl->print(++upd);

cl->println(" <br /><br>");

cl->print("Current Temperature is: "); // Αποτύπωση των τιμών Θερμοκρασίας στον απομακρυσμένο πελάτη

```
cl->print(DHT->temperature);
```

cl->print("C ");

cl->print("Humidity is: "); // Αποτύπωση των τιμών υγρασίας στον απομακρυσμένο πελάτη

```
cl->print(DHT->humidity);
```

```
cl->println("% <br /></html>");
```

```
}
```

void loop() {

dht myDHT;

EthernetClient client = myServer.available(); // Ακρόαση για εισερχόμενες συνδέσεις

if (client) {

```
Serial.println("Νέος πελάτης!");
```

```
getDHTreadings(DHT11_PIN, &myDHT);
```

```
boolean currentLineIsBlank = true; // Κάθε αίτηση http τερματίζεται με κενή γραμμή
```

```
while (client.connected()) {
```

```
if (client.available()) {
```

```
char c = client.read();
```

```
Serial.write(c);
```

/\* Η λήψη του χαρακτήρα '\n' σηματοδοτεί το τέλος της τρέχουσας γραμμής

Εφόσον η επόμενη γραμμή είναι κενή η αίτηση http έχει ολοκληρωθεί,

```
και μπορούμε να στείλουμε την απάντηση. */
```

```
if (c == '\n' && currentLineIsBlank) {
```

```
HttpPost(&client, &myDHT);
```

break;

#### }

if (c = : n') {

// Ecoume énarch néas grammús

currentLineIsBlank = true;

```
} else if (c != '\r') {
```



```
// Έχουμε χαρακτήρα στην τρέχουσα γραμμή.
currentLineIsBlank = false;
}
}
// Καθυστέρηση μέχρι να ολοκληρωθεί η λήψη από τον πελάτη
delay(1);
// Τερματισμός της σύνδεσης
client.stop();
Serial.println("Ο πελάτης αποσυνδέθηκε");
```

}

}



#### П2. Client

/\*

Web Client

Ένας απλός web client που συνδέεται σε ένα Web Server

και παίρνει δεδομένα σε τακτικά χρονικά διαστήματα.

Κύκλωμα:

Στοιχείο Ethernet

Αργοστόλι 2/9/2018

Πίρρα Τζονι

\*/

#include <SPI.h>

#include <Ethernet.h>

EthernetClient client; // Ορισμός της μεταβλητής που θα υλοποιήσει την επικοινωνία.

unsigned long lastConnectionTime = 0; // Ο χρόνος που έγινε η τελευταία σύνδεση στον WebServer, σε milliseconds

const unsigned long posting Interval = 10 \* 1000; // χρόνος επανάληψης, σε milliseconds

```
void setupSerial()
```

{

```
Serial.begin(9600);
```

```
while (!Serial) {
```

; // Αναμονή μέχρι να γίνει διαθέσιμη η σειριακή θύρα

```
}
}
```

```
void setupEthernet()
```

{

byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xF7, 0xEA}; // Αρχικά ορίζουμε την φυσική διεύθυνση (MAC address) για τον μικροελεγκτή.

IPAddress LANip(192, 168, 168, 9); // Επίσης ορίζουμε μια διεύθυνση IP από την περιοχή διευθύνσεων του τοπικού μας δικτύου

IPAddress DNSip(192, 168, 168, 1); // για την περίπτωση που δεν λειτουργεί σε αυτό DHCP WebServer καθώς και διεύθυνση DNS.

Serial.println("Αναμονή παραμέτρων δικτύου από τον DHCP:");

```
if (Ethernet.begin(mac) == 0) {
```

Serial.println("Αποτυχία λήψης παραμέτρων δικτύου από τον DHCP");

// Check for Ethernet hardware present

```
if (Ethernet.hardwareStatus() == EthernetNoHardware) { // Έλεγχος για την παρουσία του κατάλληλου υλικού
```

Serial.println("Δεν βρέθηκε υλικό Ethernet :(");

while (true) {

delay(1); // Ατέρμων βρόχος αναμονής αφού δεν υπάρχει το κατάλληλο υλικό

}

}

if (Ethernet.linkStatus() == LinkOFF) {

Serial.println("Αποσυνδέθηκε το καλώδιο Ethernet.");

}

// Στην περίπτωση αποτυχίας του DHCP κάνουμε χρήση στατικών παραμέτρων

Ethernet.begin(mac, LANip, DNSip);

Serial.print("Η IP διεύθυνση είναι: ");

Serial.println(Ethernet.localIP());

} else {

Serial.print(" Αποδόθηκε IP διεύθυνση μέσω DHCP: ");

Serial.println(Ethernet.localIP());

}

delay(1000); // 1sec χρόνος για την πλήρη εκκίνηση του υλικού.

}

void setup() {



#### ΤΕΙ ΗΠΕΙΡΟΥ 2018

setupSerial(); // Αρχικοποίηση της σειριακής επικοινωνίας για την προβολή στην κονσόλα

```
setupEthernet(); // Εκκίνηση της σύνδεσης Ethernet
```

}

```
void loop() {
```

```
// Αποστέλλουμε στην σειριακή πόρτα ότι δεδομένα έχουμε λάβει
```

```
// apó to díktuo.
```

```
if (client.available()) {
```

```
char c = client.read();
```

```
Serial.write(c);
```

#### }

```
// Επαναλαμβάνουμε την http αίτηση μόλις λήξει ο χρόνος επανάληψης
if (millis() - lastConnectionTime > postingInterval) {
    httpRequest();
  }
}
```

```
// η μέθοδος αυτή κάνει μια σύνδεση ΗΤΤΡ με τον WebServer:
```

```
void httpRequest() {
```

//char WebServer[] = "www.th-WebServer.lan"; // Είναι απαραίτητο να συμβαδίζει με το Host στην HTTP GET αίτηση

IPAddress WebServer(192, 168, 168, 5);

// Diakóptoume kábe súndesh prin thn apostolh

```
// οποιασδήποτε νέας http αίτησης
```

client.stop();

if (client.connect(WebServer, 80)) { // Έλεγχος για επιτυχή ή μη σύνδεση

Serial.println("Γίνεται σύνδεση...");

// διαμορφώνουμε μια τυπική HTTP GET αίτηση

client.println("GET /latest.txt HTTP/1.1");

client.println("Host: 192.168.168.5");

client.println("User-Agent: arduino-ethernet");

```
client.println("Connection: close");
```

client.println();

lastConnectionTime = millis(); // Καταγράφουμε το χρόνο που έγινε η σύνδεση

} else {

// Αποτυχεία δημιουργίας σύνδεσης:

Serial.println("Αποτυχία σύνδεσης");

}

}

