



**Α.Τ.Ε.Ι. ΠΕΙΡΑΙΑ**  
**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ**  
**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΙΑΣ**

**“ΣΧΕΔΙΑΣΗ ΚΑΙ ΚΑΤΑΣΚΕΥΗ ΔΙΑΤΑΞΗΣ  
ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΗΛΙΟΥ”**

**Επιβλέπων Καθηγητής:** Κος ΚΩΝ/ΝΟΣ  
ΨΩΜΟΠΟΥΛΟΣ

**Σπουδαστής:**  
ΓΕΩΡΓΙΟΣ  
ΜΠΑΓΙΑΡΤΑΚΗΣ &  
ΝΙΚΟΛΑΟΣ ΜΟΣΧΟΣ

ΑΜ: 36905, 37479

**ΑΘΗΝΑ**

**ΜΑΙΟΣ – 2017**

Copyright © Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Πειραιά

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή της για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν το συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Ανώτατου Τεχνολογικού Εκπαιδευτικού Ίδρυματος Πειραιά.

# **ΕΥΧΑΡΙΣΤΙΕΣ**

# ΠΕΡΙΕΧΟΜΕΝΑ

Ευχαριστίες .....	iii
Περιεχόμενα .....	iv
Λίστα σχημάτων .....	v
Λίστα πινάκων .....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.
Summary .....	vii
Πρόλογος .....	1
Εισαγωγή.....	2
<b>1ο Κεφάλαιο “Φωτοβολταϊκή Τεχνολογία”</b>	
1.1.Εισαγωγή.....	1
1.2 Η εξέλιξη της τεχνολογίας .....	1
1.3 Οι εφαρμογές και το κόστος της φ/β τεχνολογία .....	2
1.4 Το φωτοβολταϊκό φαινόμενο .....	4
<b>σελιδοδείκτης.</b>	
1.5 Κατηγορίες φωτοβολταϊκών στοιχείων.....	7
1.6 Η ηλιακή ακτινοβολία.....	19
1.6.1. Η συμβατική ακτινοβολία του ενός πλήρους ήλιου.....	20
1.6.2 Η θέση του ήλιου στον ορίζοντα.....	21
1.6.3 Η βέλτιστη κλίση του συλλέκτη.....	23
1.6.4 Η ένταση της ηλιακής ακτινοβολίας.....	24
1.6.5 Η επίδραση των νεφών.....	26
1.7 Τα ηλεκτρικά χαρακτηριστικά των φ/β στοιχείων.....	26
1.8 Από τι αποτελείται ένα φωτοβολταϊκό σύστημα.....	30
<b>2ο Κεφάλαιο “Θεωρία Φ/Β Ηλιοστάτη”</b>	
2.1 Εισαγωγή.....	33
2.2 Η εύρεση του μεγίστου της έντασης της ηλιακής ακτινοβολίας.....	34
2.3 Η επιλογή του φορτίου και η χρήση της ηλεκτρικής ισχύος για την εύρεση του μεγίστου.....	35
<b>3ο Κεφάλαιο “Arduino”</b>	
3.1Εισαγωγή.....	38
3.2 Ιστορικό .....	38
3.3 Εκδόσεις.....	39
3.4 Πλατφόρμα .....	39
3.4.1Υλικό (Hardware) .....	40
3.4.2Επίσημες πλακέτες .....	41
3.4.3Shields.....	41
3.5Λογισμικό.....	43
3.6Ανάπτυξη.....	44
<b>4ο Κεφάλαιο “Η κατασκευή” .....</b>	<b>44</b>
<b>Βιβλιογραφία .....</b>	<b>56</b>
Παράρτημα 1 .....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.
Παράρτημα 2 .....	Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.

## ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ

Σχήμα 1.....	3
Σχήμα 2.....	3
Σχήμα 3.....	4
Σχήμα 4.....	6
Σχήμα 5.....	7
Σχήμα 6.....	8
Σχήμα 7.....	9
Σχήμα 8.....	15
Σχήμα 9.....	16
Σχήμα 10.....	18
Σχήμα 11.....	20
Σχήμα 12.....	21
Σχήμα 13.....	21
Σχήμα 14.....	22
Σχήμα 15.....	23
Σχήμα 16.....	24
Σχήμα 17.....	25
Σχήμα 18.....	26
Σχήμα 19.....	27
Σχήμα 20.....	27
Σχήμα 21.....	28
Σχήμα 22.....	29
Σχήμα 23.....	32
Σχήμα 24.....	32
Σχήμα 25.....	35



## SUMMARY

This thesis called Design and Creation of solar tracker , is composed by two step motors, a solar panel and an Arduino.It's function is to follow the sun all days of the year. In the first chapter we mention on everything about solar panel that is how this phenomenon works , its evolution through the time , the cost of this system and about the characteristics of solar panel. We also mention on the solar radiation and the best position that a solar panel must take to have the best result.

In the second chapter we mention on the practice of the theory that the first chapter told us. We analyze the ways and the algorithms that we must use to find the best position for a solar panel.

In the third chapter we mention on the Arduino , its evolution through the time how it works and how it is programmed.

In the fourth chapter we mention on our construction , how we created it , its parts and the program of the Arduino.

We also mention on the ways that we can control the solar panels.

**Keywords:** Solar tracker, Arduino, step motors

# ΠΡΟΛΟΓΟΣ

Η παρούσα εργασία ονομάζεται Σχεδίαση και Κατασκευή Διάταξης παρακολούθησης του ήλιου και περιγράφει μία διάταξη αποτελούμενη από δύο βηματικούς κινητήρες ένα φωτοβολταϊκό panel και ένα Arduino η οποία έχει σκοπό να ακολουθάει τον ήλιο όλες τις μέρες του χρόνου. Στο πρώτο κεφάλαιο αναλύουμε τα πάντα για την φωτοβολταϊκή τεχνολογία δηλαδή για το φωτοβολταϊκό φαινόμενο πως λειτουργεί την εξέλιξη της φωτοβολταϊκής τεχνολογίας μέσα στο χρόνο, το κόστος εφαρμογής της , από τι αποτελείται ένα φωτοβολταϊκό σύστημα, τα ηλεκτρικά χαρακτηριστικά των φωτοβολταϊκών panel.Επίσης αναφερόμαστε στην ηλιακή ακτινοβολία και στην βέλτιστη κλίση του ηλιακού συλλέκτη για την μέγιστη απόδοση αλλά και στο πως επηρεάζει το νέφος στο φαινόμενο αυτό.

Το δεύτερο κεφάλαιο αναφέρεται στην πρακτική της θεωρίας πουμίλαγε το πρώτο κεφάλαιο δηλαδή τους τρόπους και τους αλγόριθμους που θα χρησιμοποιηθούν για να βρούμε την βέλτιστη κλίση του συλλέκτη για να έχουμε την βέλτιστη απόδοση.

Στο τρίτο κεφάλαιο αναφερόμαστε στο Arduino, στην εξέλιξή του μέσα στο χρόνο, τον τρόπο λειτουργίας του, στην ανάπτυξη του λογισμικού του και στον τρόπο προγραμματισμού του.

Στο τέταρτο κεφάλαιο μιλάμε για την κατασκευή μας τον τρόπο που την κατασκευάσαμε από τι αποτελείται και για το πρόγραμμα που βάλουμε στο Arduino για να ελέγχει την κίνηση των step motors ώστε να παρακολουθούμε τον ήλιο όλες τις μέρες του χρόνου.

Τέλος στην αρχή της εργασίας μας αναφέρουμε τα είδη και τους τρόπους ελέγχου των solar tracker.

Λέξεις κλειδιά:Φωτοβολταϊκό, βηματικός κινητήρας



## “ΕΙΣΑΓΩΓΗ ”

### ΕΙΔΗ ΚΑΙ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΦΩΤΟΒΟΛΤΑΪΚΩΝ ΗΛΙΟΣΤΑΤΩΝ

Με τη ραγδαία πρόοδο της τεχνολογίας στους υπολογιστές και στα συστήματα ελέγχου κατά τις τελευταίες δεκαετίες, πολλά εξελιγμένα συστήματα παρακολούθησης του ήλιου, τα οποία έχουν σχεδιαστεί για να μεγιστοποιούν την απόδοση των θερμικών και ηλιακών φωτοβολταϊκών συστημάτων, περιλαμβάνονται πλέον στη βιβλιογραφία. Τα συστήματα αυτά ταξινομούνται ανάλογα με τον τρόπο λειτουργίας τους σε συστήματα κλειστού και ανοιχτού βρόγχου.

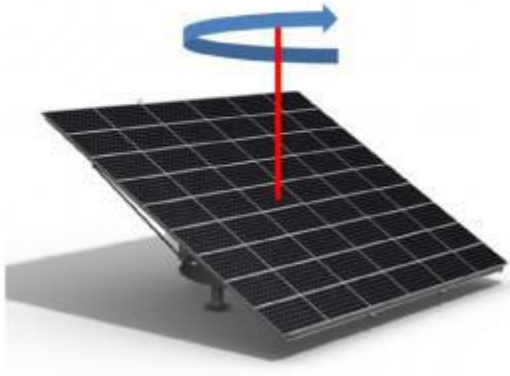
Τα συστήματα κλειστού βρόχου βασίζονται στην αρχή της επίβλεψης και της ανατροφοδότησης. Με λίγα λόγια, σε αυτά τα συστήματα υπάρχει ένα πλήθος από αισθητήρες οι οποίοι μετρούν διάφορες ποσότητες και στη συνέχεια μεταδίδουν τα δεδομένα αυτά σε έναν ελεγκτή, ο οποίος με τη σειρά του τα επεξεργάζεται και λαμβάνει απόφαση για το ποια θα είναι η επόμενη κίνηση που θα ακολουθήσουν οι συλλέκτες.

Τα συστήματα ανοιχτού βρόχου υπολογίζουν την ενέργεια που θα κάνουν, μόνο με βάση την παρούσα κατάσταση και έναν αλγόριθμο. Σε αυτά τα συστήματα δε χρησιμοποιείται ανατροφοδότηση ώστε να έχουν τη δυνατότητα να διαπιστώσουν αν έχουν πετύχει την επιθυμητή ενέργεια. Είναι πιο απλά και φθηνά συστήματα σε σχέση με αυτά του κλειστού βρόχου. Ουσιαστικά δεν έχουν τη δυνατότητα να παρατηρούν το αποτέλεσμα της ενέργειας τους και γι' αυτό το λόγο δεν είναι σε θέση να αντιληφθούν πιθανά λάθη που μπορεί συμβαίνουν κατά την λειτουργία τους. Τα συστήματα ανοιχτού βρόχου χρησιμοποιούν ένα μοντέλο το οποίο μπορεί να υπολογίζει την κίνηση του ήλιου.

Ένας άλλος τρόπος ταξινόμησης των φωτοβολταϊκών ηλιοστατών είναι με βάση το είδος της κίνησης που εκτελούν. Χωρίζονται σε δύο βασικές κατηγορίες, στα συστήματα μονού άξονα και στα συστήματα διπλού άξονα.

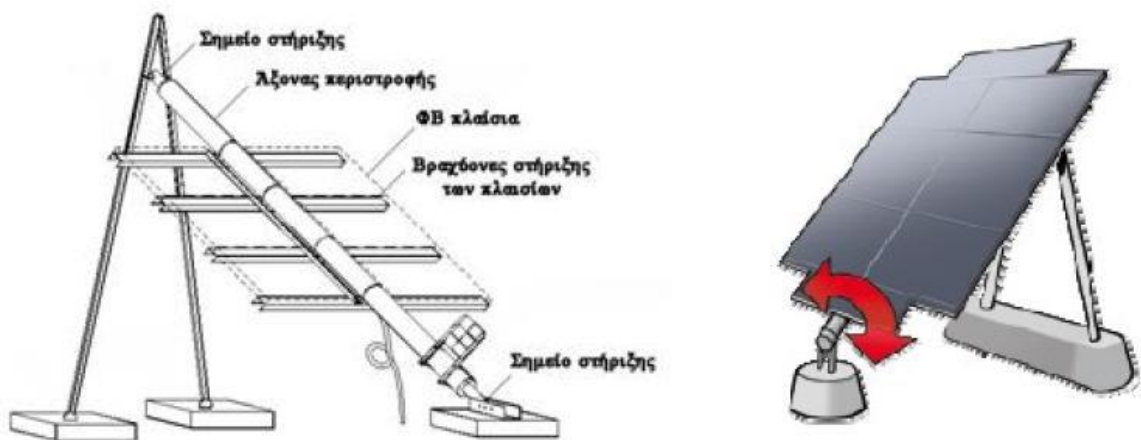
Τα συστήματα μονού άξονα είναι συστήματα στα οποία η κίνηση των πάνελ γίνεται μόνο σε έναν άξονα, αυτόν της Ανατολής-Δύσης. Τα συστήματα αυτά διακρίνονται σε δύο υποκατηγορίες, το αζιμουθιακό ηλιοτρόπιο και το πολικό ηλιοτρόπιο.

Το σύστημα αζιμουθιακού ηλιοτροπίου περιστρέφεται ως προς κατακόρυφο άξονα, με σκοπό οι ακτίνες του ήλιου να προσπίπτουν κάθετα στο συλλέκτη. Παρακολουθεί με λίγα λόγια, την αζιμουθιακή κίνηση του ήλιου. Οι συλλέκτες τοποθετούνται με συγκεκριμένη αλλά και σταθερή γωνία η οποία επιλέγεται βάση του γεωγραφικού πλάτους όπως φαίνεται και στο σχήμα 1.



Σχήμα 1. Φωτοβολταϊκο panel σε περιστροφικό άξονα

Το σύστημα πολικού ηλιοτροπίου περιστρέφεται ως προς άξονα τοποθετημένο σε γωνία η οποία είναι ίση με το γεωγραφικό πλάτος του τόπου στον οποίο είναι εγκατεστημένο και παράλληλα με τον πολικό άξονα της γης όπως φαίνεται στο σχήμα 2. Έτσι, ο ήλιος βρίσκεται σε μόνιμη βάση στο επίπεδο που είναι κάθετο στο συλλέκτη.



Σχήμα 2. Σύστημα πολικού ηλιοτροπίου

Τα συστήματα διπλού άξονα είναι συστήματα στα οποία η ρύθμιση της κλίσης των πάνελ είναι δυνατό να ρυθμιστεί και ως προς τον οριζόντιο άξονα όπως φαίνεται και στο σχήμα 3. Η δυνατότητα αυτή αυξάνει θεαματικά την απόδοση του ηλιοστάτη μεταξύ 35%-45% σε σύγκριση με τα συστήματα σταθερών βάσεων.



Σχήμα 3. Σύστημα διπλού άξονα

Η κίνηση των φωτοβολταϊκών ηλιοστατών γίνεται συνήθως με ηλεκτρο-μηχανικά ή ηλεκτρο-υδραυλικά μέσα. Αυτό έχει σα συνέπεια, όλα τα συστήματα αυτά να χαρακτηρίζονται από ιδιοκαταναλώσεις, οι οποίες είναι μικρές αφού η κίνηση είναι περιοδική (μία ανά δέκα λεπτά περίπου). Ωστόσο, καλό είναι η ενέργεια αυτή να προέρχεται από το δίκτυο της ΔΕΗ και όχι από τα φωτοβολταϊκά πάνελ λόγω της διαφοράς στην τιμή.

## ΤΕΧΝΙΚΕΣ ΑΝΙΧΝΕΥΣΗΣ ΒΕΛΤΙΣΤΗΣ ΑΠΟΔΟΣΗΣ

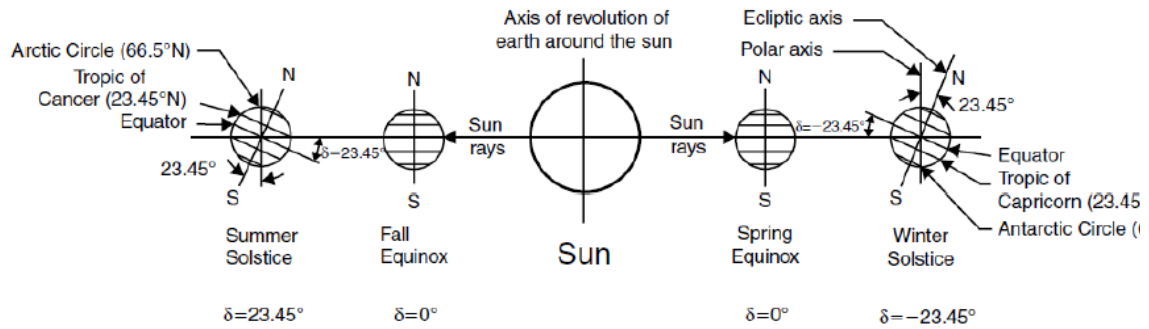
Όπως έχει αναφερθεί παραπάνω, στόχος των φωτοβολταϊκών ηλιοστατών είναι να βρίσκουν τη θέση του ήλιου στον ουρανό με σκοπό να μεγιστοποιούν την ποσότητα της ηλιακής ακτινοβολίας η οποία προσπίπτει στην επιφάνεια του φωτοβολταϊκού πάνελ. Η ανίχνευση της εκάστοτε θέσης του ήλιου μπορεί να επιτευχθεί με δύο

τρόπους. Ο πρώτος τρόπος είναι με χρήση ηλιακών αισθητήρων, οι οποίοι είναι σε θέση να αντιλαμβάνονται τη θέση του ήλιου ανά πάσα στιγμή. Ο δεύτερος τρόπος χρησιμοποιεί ένα μοντέλο της κίνησης του ήλιου το οποίο εξαρτάται από τις γεωγραφικές συντεταγμένες της περιοχής και είναι αποθηκευμένο στη μνήμη ενός υπολογιστή. Είναι απαραίτητο επίσης ένα πρόγραμμα το οποίο είναι υπεύθυνο να εκτελεί έναν αλγόριθμο που προβλέπει τις διαδοχικές θέσεις του ήλιου με βάση τη μέγιστη ένταση της ηλιακής ακτινοβολίας, ενεργοποιώντας στη συνέχεια τα μοτέρ κίνησης.

Η παρούσα εργασία βασίζεται στον δεύτερο τρόπο εύρεσης του ήλιου με τη βοήθεια ενός Arduino, μιας ηλεκτρονικής πυξίδας και ενός timer. Το πρόγραμμα που έχουμε περάσει στον Arduino είναι μία φόρμουλα κίνησης του ήλιου η οποία βασίζεται στην παρακάτω λογική.

Η γη περιστρέφεται γύρω από τον ήλιο όπως είναι γνωστό σε μια ελλειπτική τροχιά. Το επίπεδο στο οποίο περιέχεται αυτή η τροχιά ονομάζεται ελλειπτικό επίπεδο και ο χρόνος που χρειάζεται η γη για να ολοκληρώσει αυτήν την τροχιά καθορίζει το γήινο έτος.

Ταυτόχρονα, η γη περιστρέφεται γύρω από τον κεντρικό της άξονα ο οποίος ονομάζεται πολικός άξονα και η κίνηση αυτή καθορίζει τη γήινη ημέρα. Ο πολικός άξονας έχει μία τροχιά γύρω από τον ήλιο, διατηρώντας πάντα σταθερή γωνία  $23.45^\circ$  με το ελλειπτικό επίπεδο, με τρόπο ώστε η γωνία που σχηματίζεται μεταξύ του ισημερινού επιπέδου και μιας ευθείας γραμμής η οποία εκτείνεται από το κέντρο της γης μέχρι το κέντρο του ήλιου να αλλάζει συνεχώς. Αυτή η γωνία καλείται ηλιακή απόκλιση  $\delta$  και μπορεί να θεωρηθεί σχεδόν σταθερή κατά τη διάρκεια μιας οποιασδήποτε ημέρας. Η μέγιστη μεταβολή της γωνίας  $\delta$  μέσα σε 24 ώρες είναι λιγότερο από  $0.5^\circ$ . Στο σχήμα 4 φαίνεται η θέση του άξονα της γης σε σχέση με τον ήλιο ανάλογα με τις εποχές.



Σχήμα 4. Η θέση του άξονα της γης σε σχέση με τον ήλιο ανάλογα με τις εποχές

Ο υπολογισμός της γωνίας ηλιακής απόκλισης  $\delta$  μπορεί να γίνει με μεγάλη ακρίβεια για οποιαδήποτε ημέρα του έτους χρησιμοποιώντας την παρακάτω σχέση :

$$\delta = [0,006918 - 0,399912 * \cos(\Gamma) + 0,070257 * \sin(\Gamma) + 0,006758 * \cos(2\Gamma) + 0,000907 * \sin(2\Gamma) - 0,002697 * \cos(3\Gamma) + 0,00148 * \sin(3\Gamma)] * 180/\pi$$

Όπου  $\Gamma$  (σε rad) δίνεται από τη σχέση :

$$\Gamma = 2 * \pi * (dn - 1) * 365$$

όπου  $dn$  ο αριθμός της μέρας αρχίζοντας τη μέτρηση από την αρχή του έτους. Δηλαδή το  $dn$  παίρνει τιμές από 1 στις 1 Ιανουαρίου ως 365 στις 31 Δεκεμβρίου. Σύμφωνα με την ελλειπτική τροχιά της γης η απόσταση μεταξύ ήλιου και γης κατά τη διάρκεια του έτους παίρνει τιμές σύμφωνα με την παρακάτω σχέση :

$$Eo = \frac{Ro}{R} = 1,0011 + 0,034221 * \cos(\Gamma) + 0,001280 * \sin(\Gamma) + 0,000719 * \cos(2\Gamma) + 0,000077 * \sin(2\Gamma)$$

όπου:  $r$  είναι η απόσταση μεταξύ ήλιου και γης και  $ro$  η μέση τιμή του  $r$  που είναι ίση με 149.600.000Km.

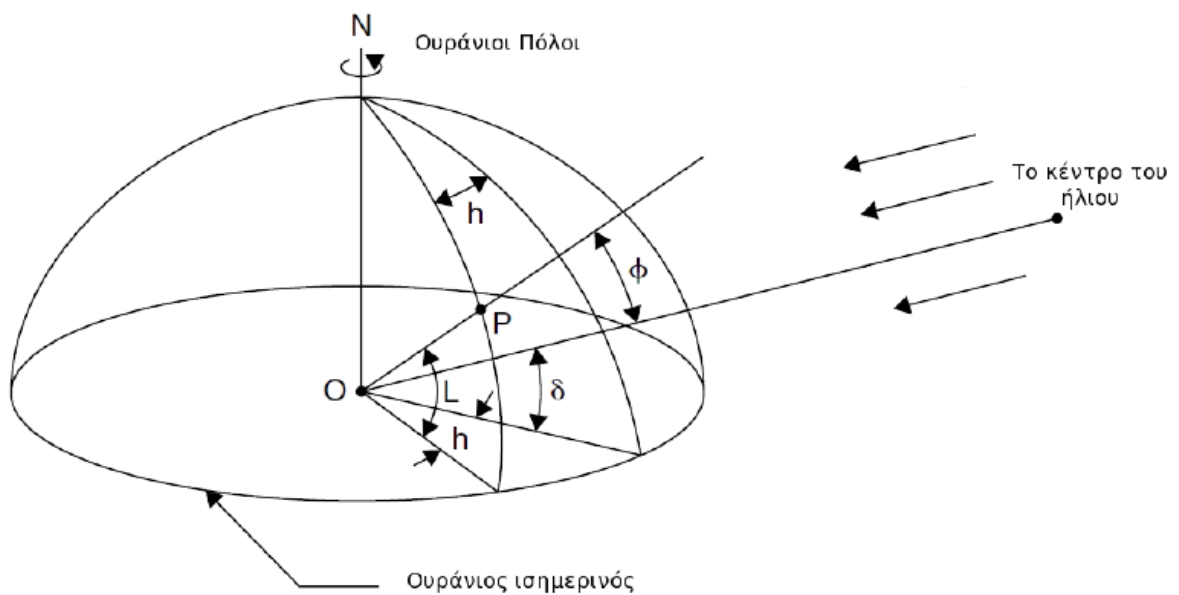
Οι σχέσεις (1) και (3) λαμβάνουν υπόψη τη γωνιακή ταχύτητα της γης. Σε πολλές εφαρμογές της εφαρμοσμένης μηχανικής η γωνιακή ταχύτητα είναι δυνατό να θεωρηθεί σταθερή, προσεγγιστικά. Η θεώρηση αυτή είναι σχετικά ακριβής και

συνάμα βολική. Σε αυτή τη περίπτωση οι εξισώσεις (1) και (3) αντικαθίστανται από τις παρακάτω :

$$\delta = 23,45 * \sin\left(\frac{360}{365} * (dn + 284)\right)$$

$$Eo = 1 + 0,033 * \cos\left(360 * \frac{dn}{284}\right)$$

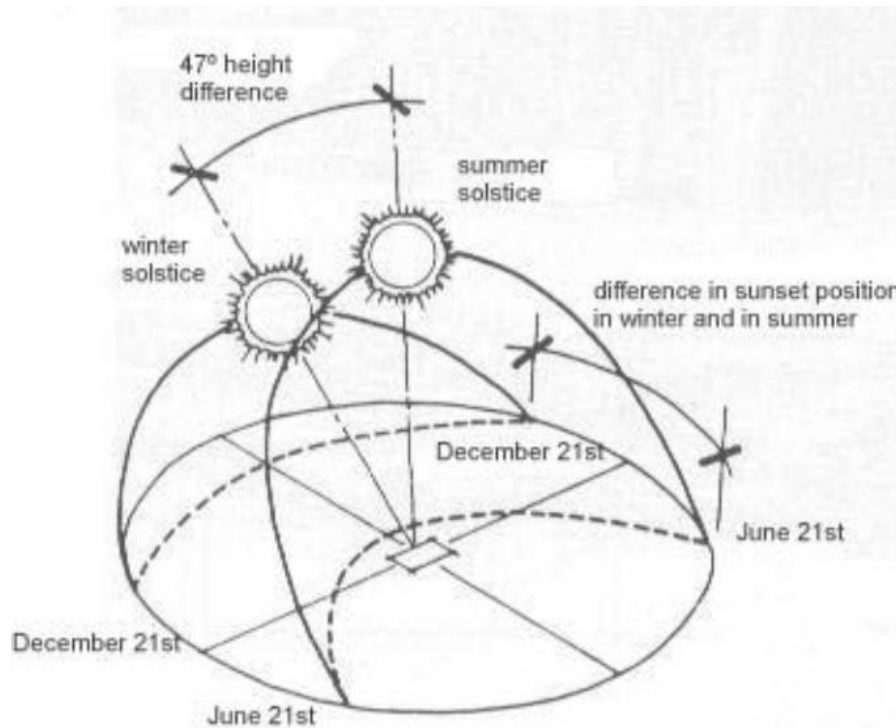
Ένας πολύ κλασικός τρόπος για την αναπαράσταση του ουρανού είναι ως μία σφαίρα η οποία κεντροθετείται σε ένα καθορισμένο σημείο τη γης, όπως παρουσιάζεται και στο σχήμα 5. Αυτό το μοντέλο αποκαλείται «ουράνια σφαίρα». Κάθε σημείο του, αντιπροσωπεύει μια πορεία με κατεύθυνση προς τον ουρανό όπως αυτή φαίνεται από τη γη.



Σχήμα 5. Αναπαράσταση του ουρανού

Αν χρησιμοποιηθεί το συγκεκριμένο μοντέλο αναπαράστασης τότε η μετακίνηση της γης γύρω από τον ήλιο θα μπορεί να μετασχηματιστεί ουσιαστικά σε μια μετακίνηση του ήλιου ως προς τη γη η οποία λαμβάνεται σαν σταθερή όπως φαίνεται και στο

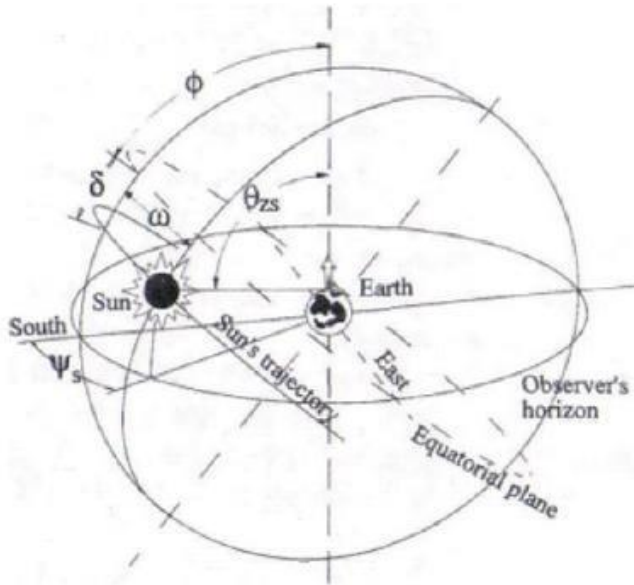
σχήμα 6. Σε αυτή τη περίπτωση ο ήλιος ονομάζεται ελλειπτικός και σχηματίζει γωνία  $23,45^\circ$  με τον ουράνιο ισημερινό. Ο ήλιος ολοκληρώνει έναν τέτοιο κύκλο μία φορά το χρόνο ενώ η ουράνια σφαίρα περιστρέφεται γύρω από τη γη (που θεωρείται σταθερή) μία φορά την ημέρα. Κατά τον τρόπο αυτό, ο ήλιος διαγράφει έναν κύκλο γύρω από τη γη. Η διάμετρος αυτού του κύκλου αλλάζει μέρα με τη μέρα, φτάνοντας το μέγιστο στις ισημερίες και το ελάχιστο στα ηλιοστάσια.



Σχήμα 6. Η κίνηση του ήλιου σε σχέση με τη γη

Εάν από οποιοδήποτε σημείο της γης τραβηχτεί μία κατακόρυφος, αυτή θα τέμνει την ουράνια σφαίρα σε σημεία γνωστά ως ζενίθ και ναδίρ. Η γωνία που σχηματίζεται μεταξύ της κατακόρυφου αυτής και του πολικού άξονα ορίζει το γεωγραφικό πλάτος  $\phi$  το οποίο θεωρείται θετικό στο βόρειο ημισφαίριο και αρνητικό στο νότιο ημισφαίριο. Ο τοπικός μεσημβρινός πρόκειται για τον μεγάλο κύκλο στη γη ο οποίος περιέχει τους πόλους, ζενίθ και ναδίρ και παρουσιάζεται στο σχήμα 7.

## Zenith



## Nadir

Σχήμα 7. Οι πόλοι Zenith και Nadir

Για τον καθορισμό της θέσης ενός σημείου όπως για παράδειγμα του ηλίου στην ουράνια σφαίρα σε σχέση με ένα σημείο στη γη γίνεται χρήση δύο συντεταγμένων. Η πρώτη είναι η γωνία ζενίθ,  $\theta Z$ , η οποία σχηματίζεται μεταξύ της κατακόρυφου και μιας γραμμής από τη γη στο συγκεκριμένο σημείο. Η δεύτερη είναι το αζιμούθιο  $\Psi S$ , που σχηματίζεται μεταξύ των μεσημβρινών του σημείου και της θέσης του το συμπλήρωμα της γωνίας  $\theta Z$  είναι γνωστό ως ανύψωση  $\gamma$ . Το αζιμούθιο θεωρείται θετικό προς τη δύση και αρνητικό προς την ανατολή. Ο κύκλος ο οποίος αντιστοιχεί στο γεωμετρικό τόπο των σημείων που έχουν  $\gamma = 0$  ονομάζεται ορίζοντας. Η γωνία ζενίθ ( $\theta Z$ ) και το αζιμούθιο ( $\Psi S$ ) σε οποιαδήποτε δεδομένη στιγμή, υπολογίζονται από τις σχέσεις :

$$\cos(\theta Z) = \sin\delta * \sin\varphi * \cos\delta * \cos\varphi * \cos\omega = \sin\gamma$$

$$\cos(\Psi) = (\sin\gamma * \sin\varphi - \sin\delta) / (\cos\gamma * \cos\varphi)$$



όπου το  $\omega$  είναι μια ποσότητα που καλείται πραγματικός ηλιακός χρόνος ή ηλιακή ώρα. Στο μεσημέρι κάθε ημέρας το  $\omega$  παίρνει την τιμή 0, το πρωί λαμβάνεται ως αρνητικό και το απόγευμα ως θετικό.

Το σύστημα που μόλις περιγράφηκε, δίνει τις γωνίες  $\theta Z$  και  $\Psi S$  σε σχέση με μια οριζόντια επιφάνεια. Στην πραγματικότητα όμως, οι περισσότερες πρακτικές εφαρμογές απαιτούν τον καθορισμό της θέσης του ήλιου σχετικά με ένα κεκλιμένο επίπεδο. Ο προσανατολισμός μιας επιφάνειας μπορεί να περιγραφεί γενικά από την κλίση  $\beta$  η οποία είναι η γωνία που διαμορφώνεται με το οριζόντιο επίπεδο και το αζιμούθιο  $\alpha$  της κάθετης στην επιφάνεια. Η γωνία της ηλιακής πρόσπτωσης μεταξύ των ακτινών του ήλιου και της κάθετης στην επιφάνεια μπορεί είναι δυνατό να υπολογιστεί από τη σχέση :

$$\begin{aligned} \mathit{Cos}(\theta Z) = & \mathit{sin}\delta * \mathit{sin}\varphi * \mathit{cos}\beta * \mathit{cos}\varphi - \mathit{sin}\delta * \mathit{cos}\varphi * \mathit{sin}\beta * \mathit{cos}\alpha + \mathit{cos}\delta \\ & * \mathit{cos}\varphi * \mathit{cos}\beta * \mathit{cos}\omega + \mathit{cos}\delta * \mathit{cos}\alpha * \mathit{cos}\omega * \mathit{sin}\varphi * \mathit{sin}\beta \\ & + \mathit{sin}\alpha * \mathit{sin}\omega * \mathit{sin}\beta * \mathit{cos}\delta \end{aligned}$$

Η σχέση 8 πρόκειται για μία πολύ χρήσιμη σχέση η οποία σε πολλές περιπτώσεις απλοποιείται σημαντικά όπως για παράδειγμα στην περίπτωση επιφανειών προσανατολισμένων προς το νότο που παίρνει την παρακάτω μορφή :

$$\mathit{Cos}(\theta Z) = \mathit{sin}\delta * \mathit{sin}(\varphi - \beta) + \mathit{cos}\delta * \mathit{cos}(\varphi - \beta) * \mathit{cos}\omega$$

Με τη βοήθεια του timer μέσω του διαδικτύου λαμβάνουμε τιμές για την τρέχουσα ημερομηνία και ώρα που μπαίνει σε λειτουργία το Arduino και με τη βοήθεια της ηλεκτρονικής πυξίδας βρίσκουμε πάντα την ανατολή.

## **ΚΕΦΑΛΑΙΟ 1**

### **Φωτοβολταϊκή Τεχνολογία**

#### **1.1 Εισαγωγή**

Στην σημερινή εποχή με το έντονο ενεργειακό πρόβλημα λόγω της πετρελαϊκής κρίσης είναι μείζονος σημασίας η ανάπτυξη και ευρεία χρήση εναλλακτικών πηγών παραγωγής ηλεκτρικής ενέργειας. Αυτές οι μορφές ενέργειας πρέπει να είναι φιλικές προς το περιβάλλον καθώς η παραγωγή ηλεκτρικής ενέργειας ενοχοποιείται σε πολύ μεγάλο ποσοστό για την επιβάρυνση του περιβάλλοντος, αλλά και να μην εξαντλούν τους φυσικούς πόρους της γης. Μια μορφή από τις ονομαζόμενες Ανανεώσιμες Πηγές Ενέργειας είναι και παραγωγή με χρήση της φωτοβολταϊκής τεχνολογίας. Η φωτοβολταϊκή τεχνολογία χρησιμοποιεί την ηλιακή ακτινοβολία για την παραγωγή ηλεκτρικού ρεύματος. Το πλεονέκτημα της χρήσης της είναι εμφανές: Η ηλιακή ακτινοβολία είναι πολύ σταθερή και ανεξάντλητη για τον σχετικό χρόνο της παρουσίας του ανθρώπου στην γη. Αξίζει λοιπόν να ασχοληθούμε περισσότερο με αυτήν την μορφή ενέργειας, προσπαθώντας να την βελτιώσουμε και να την αναπτύξουμε.

Προς αυτήν την κατεύθυνση κινείται και η παρούσα εργασία. Σκοπός της είναι η κατασκευή και η μελέτη της απόδοσης μιας διάταξης για την φ/β τεχνολογία γνωστής και ως ηλιοστάτης/παρακολουθητής του ήλιου (διεθνής ορολογία : solar tracker/heliostat). Η διάταξη αυτή ανιχνεύει την θέση του ήλιου στον ουρανό και προσανατολίζει τον φωτοβολταϊκό συλλέκτη κατάλληλα με σκοπό την μεγιστοποίηση της ενέργειας που παράγει. Ο τρόπος υλοποίησης της μεθόδου ανεύρεσης του ήλιου και η διαδικασία κατασκευής ακολουθούν στην συνέχεια της εργασίας. Πριν όμως είναι πολύ σημαντικό να αναφερθούν τα βασικά χαρακτηριστικά των φωτοβολταϊκών συστημάτων, γιατί πάνω στην θεωρία αυτή θα βασιστεί η υλοποίηση του solar tracker.

#### **1.2 Η εξέλιξη της τεχνολογίας**

Ο όρος φωτοβολταϊκή ενέργεια αναφέρεται στην ενέργεια που προερχόμενη από την ηλιακή ακτινοβολία μετατρέπεται σε ηλεκτρική μορφή μέσω του φωτοβολταϊκού φαινομένου. Η ιστορία της φωτοβολταϊκής ενέργειας ξεκινάμε την παρατήρηση του

φαινομένου σε μεταλλικά ηλεκτρόδια (Pt,Ag) βυθισμένα σε ηλεκτρολύτες από τον Becquerel το 1839. Η κατασκευή του πρώτου ηλιακού στοιχείου ήρθε έναν αιώνα περίπου αργότερα το 1937 από τους Fischer και Godden. Η εμπορική παραγωγή ηλιακών στοιχείων ξεκίνησε το 1956, τρία χρόνια μετά την ανακάλυψη της μεθόδου σχηματισμού ενώσεων p-n με διάχυση προσμίξεων. Με την ανάπτυξη νέων τεχνικών κατασκευής των ηλιακών στοιχείων αυξήθηκε σταδιακά και η απόδοση τους στην μετατροπή της ενέργειας από ηλιακή σε ηλεκτρική. Από 1% απόδοση των στοιχείων από Se το 1939 περνάμε σε απόδοση 6% το 1954 με στοιχεία πυριτίου (Si) ένωσης p-n με την μέθοδο διάχυσης προσμίξεων και φτάνουμε στο 14% το 1972 (Lindmayer & Allison). Παράλληλα γίνονται χρήσεις και άλλων υλικών και ενώσεων. Χαρακτηριστικό παράδειγμα είναι η κατασκευή στοιχείου από αρσενικούχο γάλλιο (GaAs) απόδοσης 16% το 1976 ενώ γίνεται επίσης προσπάθεια μείωσης του υψηλού κόστους με χρήση φθηνής πρώτης ύλης, όπως άμορφου πυριτίου με σημαντικά αποτελέσματα: Το 1984 βγαίνει στην παραγωγή φωτοβολταϊκό στοιχείο άμορφου πυριτίου με απόδοση 5% στην Ιαπωνία.

Στις μέρες μας νέα υλικά είναι υπό έρευνα, άλλα πολύπλοκων ενώσεων (όπως Cadmium telluride και Copper indium diselenide), άλλα οργανικής βάσης ενώ αναπτύσσονται τεχνικές συγκέντρωσης της ηλιακής ακτινοβολίας, όπως θα αναλυθεί και στα επόμενα κεφάλαια .

### **1.3 Οι εφαρμογές και το κόστος της φ/β τεχνολογίας**

Η χρήση της φωτοβολταϊκής τεχνολογίας έχει μερικά πολύ σημαντικά πλεονεκτήματα που την κάνουν ιδιαίτερα ελκυστική εναλλακτική μορφή ενέργειας. Αυτός είναι και ο λόγος που δημιουργήθηκαν κίνητρα στην φωτοβολταϊκή αγορά για αύξηση της παραγωγής (βέβαια με παράλληλη βελτίωση των τεχνολογιών), έχοντας ως αποτέλεσμα την ραγδαία πτώση των τιμών τους. Έτσι μπορούμε εμείς σήμερα να μιλάμε για ευρεία χρήση των φωτοβολταϊκών σε εφαρμογές υψηλής ισχύος (οικιακή και βιομηχανική χρήση) ενώ παλιότερα εύρισκαν εφαρμογή σε πολύ χαμηλές απαιτήσεις ισχύος (ηλεκτρονικά ρολόγια , υπολογιστές τσέπης κ.τ.λ.). Χαρακτηριστικό της πτώσης των τιμών των ηλιακών στοιχείων είναι ότι το κόστος ανά  $W_p$  (μονάδα μέτρησης της ισχύος στα φ/β ανάλογη του Watt στην ηλεκτρική ισχύ-θα αναλυθεί εκτενώς στην συνέχεια) το έτος 1956 ήταν περίπου στα 1000\$ ανά  $W_p$  (την εποχή που χρησιμοποιούνταν μόνο σε διαστημικές εφαρμογές). Σήμερα

πλέον μιλάμε για κόστος περί των 5 \$ ανά Wp, παρέχοντας την δυνατότητα ανάπτυξης ολοκληρωμένων λύσεων σε περιοχές δυσπρόσιτες, εκτός δικτύου ή σε περιοχές και περιπτώσεις που το κόστος της επέκτασης του δικτύου είναι μεγαλύτερο από την πλήρη εγκατάσταση ενός αυτόνομη φ/β συστήματος. Παραδείγματα τέτοιων εφαρμογών είναι :

- Ηλεκτροδότηση εξοχικών κατοικιών χαμηλών ηλεκτρικών απαιτήσεων
- Εγκαταστάσεις αφαλάτωσης (σε νησιά) και αντλιοστάσια υδροδότησης
- Φάροι, κεραιές κινητής τηλεφωνίας, κεραιές τηλεόρασης και ραδιόφωνου στην ύπαιθρο
- Αγροτικές γεωτρήσεις, συστήματα ανίχνευσης κινδύνου πυρκαγιάς στα δάση
- Παροχή ισχύος στα συστήματα ασφαλείας και πυρανίχνευσης σε μη ηλεκτροδοτημένους αποθηκευτικούς χώρους
- Οδικός φωτισμός επαρχιακών περιοχών μακριά απ' το δίκτυο
- Φόρτιση μπαταριών υψηλής χωρητικότητας (π.χ. για ηλεκτροκίνητα οχήματα)
- Βοηθητική ηλεκτροδότηση σε πλοία
- Παροχή ηλεκτρικής ενέργειας σε διαστημικές εφαρμογές
- Εν γένει απομονωμένες εγκαταστάσεις και πολλά άλλα

Τα φωτοβολταϊκά συστήματα παρουσιάζουν τα εξής πλεονεκτήματα :

- Τεχνολογία φιλική στο περιβάλλον : δεν προκαλούνται ρύποι κατά την παραγωγή ηλεκτρικής ενέργειας
- Η ηλιακή ενέργεια είναι αποκεντρωμένο «καύσιμο», διατίθεται παντού και δεν στοιχίζει απολύτως τίποτα
- Αθόρυβη λειτουργία
- Σχεδόν μηδενικές απαιτήσεις συντήρησης (για τα σταθερά)
- Μεγάλη διάρκεια ζωής: οι κατασκευαστές εγγυώνται χαρακτηριστικά απόδοσης για 20-30 χρόνια λειτουργίας
- Εύκολη επεκτασιμότητα

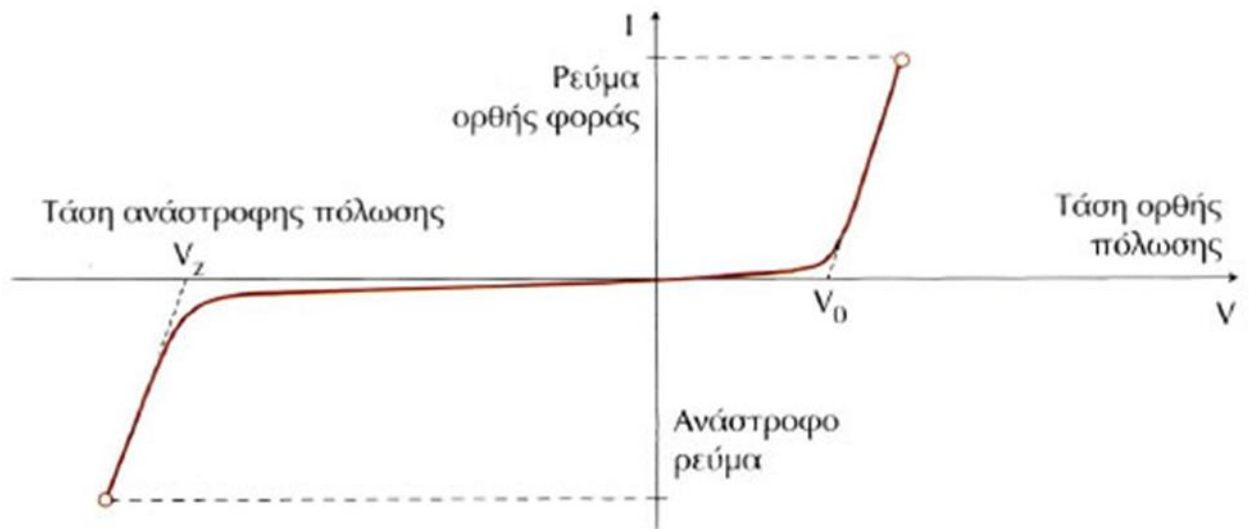
Το βασικό μειονέκτημα της φωτοβολταϊκής τεχνολογίας παραμένει το κόστος της. Παρά την ραγδαία μείωση των τιμών εξακολουθεί να παραμένει ακριβή συγκριτικά με τις παραδοσιακές μεθόδους παραγωγής ηλεκτρισμού. Όμως οι τελευταίες έχουν μερίδιο ευθύνης για εκπομπή αερίων του θερμοκηπίου και ως εκ τούτου εναλλακτικές 'πράσινες' μορφές ενέργειας πρέπει να προωθηθούν. Τέλος αξίζει να αναφερθεί ότι στην εποχή μας με το σοβαρό ενεργειακό πρόβλημα να προέρχεται

κυρίως από την μονομερή εξάρτηση της παγκόσμιας αγοράς από το μονοπώλιο του πετρελαίου, είναι εμφανής η ανάγκη να απαγκιστρωθούμε από την εξάρτηση αυτή. Η λύση είναι η εκμετάλλευση των εναλλακτικών μορφών ενέργειας, με σαφές πλεονέκτημα την φωτοβολταϊκή στην Ελλάδα λόγω της υψηλής μέσης ηλιακής ακτινοβολίας που χαίρει σε σχέση με άλλες ευρωπαϊκές χώρες .

#### **1.4 Το φωτοβολταϊκό φαινόμενο**

Τα φωτοβολταϊκά στοιχεία κατασκευάζονται από ημιαγώγιμα υλικά όπως το πυρίτιο (Si) το γερμάνιο(Ge) καθώς και από ενώσεις όπως αρσενικούχο γάλλιο (GaAs) και θειούχο κάδμιο (CdS). Το κυρίαρχο εξ αυτών είναι το πυρίτιο, κατέχοντας το μεγαλύτερο μερίδιο της παραγωγής (όπως και στην παραγωγή των περισσότερων ηλεκτρονικών εξαρτημάτων). Στο πυρίτιο (τετρασθενές ημιαγώγιμο υλικό) όταν απορροφηθεί ενέργεια υπό μορφή ακτινοβολίας ή θερμότητας προκαλείται διάσπαση ηλεκτρονιακών δεσμών, ελευθερώνοντας έτσι ηλεκτρόνια στο σώμα του ημιαγωγού. Αυτό συμβαίνει και σε συνθήκες θερμοκρασίας δωματίου (η αναλογία των ελεύθερων-δεσμευμένων ηλεκτρονίων σε συνθήκες θερμοκρασίας δωματίου είναι της τάξης του  $10^{-13}$  ). Οι θέσεις από όπου φεύγουν τα ηλεκτρόνια καλούνται οπές (holes). Τα ελεύθερα ηλεκτρόνια και οι οπές είναι υπεύθυνα για την ηλεκτρική αγωγιμότητα των ημιαγωγών . Τα παραπάνω ισχύουν στους λεγόμενους ενδογενείς ή καθαρούς ημιαγωγούς. Σε αυτούς γίνεται πρόσμιξη πεντασθενών (αρσενικό) ή τρισθενών (βάριο) στοιχείων που λειτουργούν ως δότες και αποδέκτες ηλεκτρονίων αντίστοιχα. Έτσι δημιουργούνται οι ημιαγωγοί πρόσμιξης τύπου n , στους οποίους κύριοι φορείς του ηλεκτρικού ρεύματος είναι τα αρνητικά φορτισμένα ελεύθερα ηλεκτρόνια και οι ημιαγωγοί πρόσμιξης τύπου p , η αγωγιμότητα των οποίων οφείλεται κυρίως στις οπές (θεωρούνται θετικά φορτισμένες). Σε αυτούς τους φορείς αναφερόμαστε ως φορείς πλειονότητας, σε αντίθεση με τους μειονότητας που οφείλονται στην αγωγιμότητα του ημιαγωγού. Βασικός νόμος για τους ημιαγωγούς είναι ο νόμος δράσης των μαζών που λέει : (συγκέντρωση ελεύθερων ηλεκτρονίων)\*(συγκέντρωση οπών)= (ενδογενής συγκέντρωση)<sup>2</sup>. Για να εκδηλωθούν οι σημαντικές ιδιότητες των ημιαγωγών πρέπει να δημιουργηθεί μια ένωση p-n με τα φαινόμενα διάχυσης που τη συνοδεύουν. Εμφανίζεται τότε μια περιοχή αραίωσης στην ένωση των δυο περιοχών και αναλόγως την πολικότητα της τάσης που

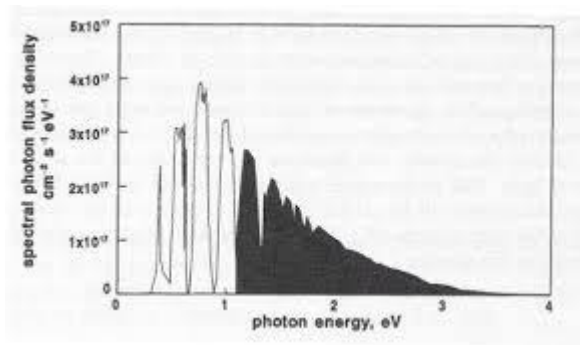
εφαρμόζουμε στα άκρα της διάταξης παίρνουμε το παρακάτω διάγραμμα (έχουμε μια ημιαγωγική δίοδο) :



*Σχήμα 8 Η χαρακτηριστική καμπύλη έντασης(I)-τάσης(V) μιας διόδου p-n*

Στο παραπάνω σχήμα διακρίνουμε τις δυο περιοχές ορθής και ανάστροφης πόλωσης καθώς το λεγόμενο ανάστροφο ρεύμα κόρου που εμφανίζεται για αρνητικές τιμές της επιβαλλόμενης τάσης . Τα φωτοβολταϊκά στοιχεία που απασχολούν την εργασία αυτή δεν είναι τίποτε άλλο από ημιαγωγικές διόδους που λειτουργούν υπό συνθήκες ορθής πόλωσης, των οποίων όμως τα ηλεκτρικά χαρακτηριστικά εξαρτώνται και από το ανάστροφο ρεύμα κόρου. Πώς γίνεται όμως η μετατροπή της ενέργειας από ηλεκτρομαγνητική (ηλιακή) σε ηλεκτρική (ρεύμα); Η μετατροπή βασίζεται στην

κβαντική φύση του φωτός: Κβάντα φωτός (φωτόνια) με ενέργεια  $(h\nu) / E = hc / \lambda$  = το καθένα προσπίπτουν στην γήινη επιφάνεια με ρυθμό  $17,4 \cdot 10^{17}$  κάθε δευτερόλεπτο μιας ηλιόλουστης ημέρας. Προσπίπτοντας τα φωτόνια στην επιφάνεια του φ/β στοιχείου, μπορούν (κάποια εξ αυτών με ενέργεια τουλάχιστον ίση με αυτή του ενεργειακού διακένου του ημιαγωγού) να διεγείρουν ένα ηλεκτρόνιο (το καθένα) από την ζώνη σθένους στην ζώνη αγωγιμότητας δημιουργώντας ένα ελεύθερο ηλεκτρόνιο και μια οπή. Αυτή η κίνηση των φορτίων (ελεύθερου ηλεκτρονίου και οπής) δημιουργούνται ηλεκτρικό ρεύμα που λέμε ότι παράγει το φ/β στοιχείο (ουσιαστικά μετατρέπει). Είναι προφανές ότι ένα φ/β στοιχείο δεν μπορεί να απορροφήσει όλα τα φωτόνια που προσπίπτουν σε αυτό παρά μόνο ένα μέρος του ηλιακού φάσματος, για τον απλό λόγο ότι δεν ταιριάζουν όλες οι ενέργειες των φωτονίων με το ενεργειακό διάκενο του ημιαγωγού. Για να γίνει αυτό κατανοητό ακολουθεί το παρακάτω σχήμα 9 που δείχνει ποιο ποσοστό της ακτινοβολίας απορροφάται :



Σχήμα 9. Η φασματική πυκνότητα ροής φωτονίων που εκμεταλλεύεται ένας συλλέκτης

Η τάση που αναπτύσσει η φ/β κυψέλη υπό συνθήκες ακτινοβολίας έχει άνω όριο την τιμή  $V_{oc} = \frac{q}{e} \ln \left( \frac{I_{ph}}{I_0} + 1 \right)$ , στην πράξη όμως εμφανίζονται μικρές διαφοροποιήσεις.

### 1.5 Κατηγορίες φωτοβολταϊκών στοιχείων

Τα βασικά στάδια κατασκευής ενός φ/β πάνελ (όπως συνηθίζεται να λέγεται) είναι τα εξής τέσσερα :

- Από άμμος σε καθαρό πυρίτιο (from sand to pure silicon)
- Από καθαρό πυρίτιο σε δίσκους & κρυστάλλους (from silicon feedstock to crystals & wafers)
- Από δίσκους σε κυψέλες (from silicon wafers to solar cells)

- Από κυψέλες σε μονάδα (from cells to modules)

Τα στάδια της επεξεργασίας είναι ιδιαίτερος πολύπλοκα και απαιτούν εξειζητημένες τεχνικές (βλέπε μέθοδος Czochralski σχήμα 1.4) που συνδυάζουν γνώση από διάφορα επιστημονικά πεδία (δυναμική των ρευστών, χημεία, ηλεκτροχημεία, ηλεκτρομαγνητισμό) και ως εκ τούτου γίνεται μια απλή αναφορά. Υπάρχουν διάφορες κατηγορίες φ/β στοιχείων ανάλογα με τα υλικά που χρησιμοποιούνται και αντίστοιχες τεχνολογίες που αυτά κατασκευάζονται. Αποτέλεσμα όμως της πληθώρας των διαφορετικών στοιχείων είναι η ποικιλία σε απόδοση και ηλεκτρικά χαρακτηριστικά των παραχθέντων στοιχείων. Για λόγους πληρότητας του κειμένου θα αναφερθώ επιγραμματικά στις κατηγορίες των στοιχείων καθ' ότι λεπτομερής ανάλυση αυτών δεν συνάδει με τον χαρακτήρα της εργασίας αυτής

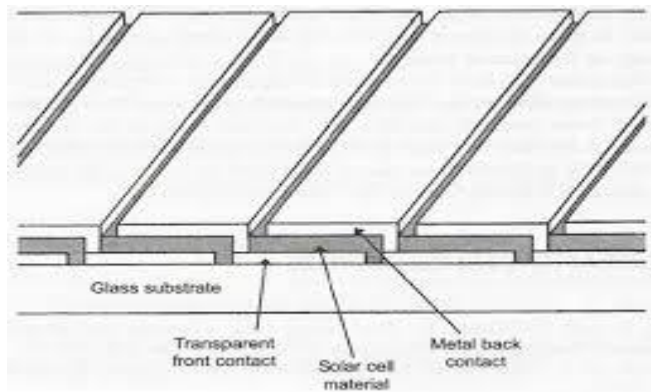
**. Οι βασικές κατηγορίες είναι :**

- Crystalline silicon solar cell technology (single crystalline or multicrystalline). Στα ελληνικά αποδίδεται με τον όρο μονοκρυσταλλικά και πολυκρυσταλλικά φωτοβολταϊκά στοιχεία. Πρόκειται για την πιο διαδεδομένη τεχνική επί του παρόντος έχοντας επιτύχει εργαστηριακές αποδόσεις κοντά στο 25%, αρκετά κοντά στο θεωρητικά επιτεύξιμο. Σε αυτό συνέβαλαν τα πολλά κοινά που παρουσιάζει η τεχνολογία αυτή με την επεξεργασία που γίνεται στο πυρίτιο από την βιομηχανία της μικροηλεκτρονικής. Αν και η δημιουργία υψηλής ποιότητας υλικών όπως οι μονοκρύσταλλοι πυριτίου απαιτεί πολύπλοκη τεχνολογία, εν τούτοις η βιομηχανία έχει αναπτύξει τους απαραίτητους μηχανισμούς, με αποτέλεσμα να βγαίνουν στην αγορά πλέον φ/β πάνελ με απόδοση 11%-16%. Έτσι από επίσημα στοιχεία, για το έτος 1998, το 83% των φ/β συστημάτων παγκοσμίως χρησιμοποιούσαν τεχνολογία πολυκρυστάλλων (44%) και μονοκρυστάλλων (39%), αφήνοντας μόλις το 13% για την τεχνολογία άμορφου πυριτίου (βλέπε παρακάτω) και ένα 4% για άλλες τεχνολογίες (όπως cells based on ribbon silicon, cadmium telluride και διαφόρων τεχνικών συγκεντρωτικών συστημάτων).

- Thin film solar cell technology (τεχνολογία κυψελών λεπτού στρώματος). Η τεχνολογία αυτή (υπάρχει ήδη στην αγορά φωτοβολταϊκών) υπόσχεται σημαντική μείωση των τιμών με χρήση λιγότερου υλικού και ενέργειας στην διαδικασία κατασκευής των κυψελών. Επιπλέον υπάρχει το σημαντικότερο πλεονέκτημα να κατασκευάζονται μονάδες (modules) σε ολοκληρωμένη μορφή χωρίς την απαίτηση



για δαπανηρές διασυνδέσεις των ανεξάρτητων κυψελών. . Ο τρόπος που γίνεται η ολοκληρωμένη διασύνδεση στη μονάδα τεχνολογίας λεπτού στρώματος φαίνεται στο παρακάτω σχήμα :



Σχήμα 10. Εσωτερικά διασυνδεδεμένη μονάδα

Στην κατηγορία της thin film solar cell technology ανήκουν μερικές ενδιαφέρουσες τεχνικές παραγωγής φ/β στοιχείων όπως :

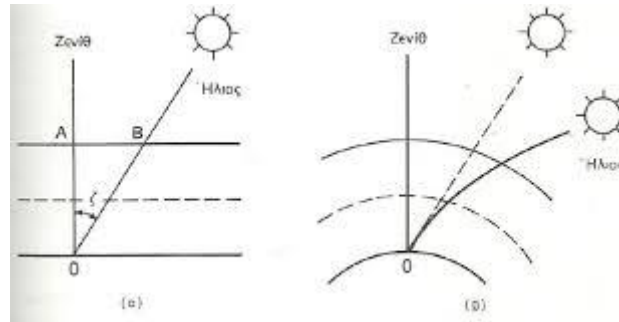
- Κυψέλες άμορφου πυριτίου (amorphous silicon cells). Εδώ το πυρίτιο διαφέρει από την κρυσταλλική μορφή στο ότι τα άτομα δεν βρίσκονται σ' αρκετά ακριβείς αποστάσεις μεταξύ τους και οι γωνίες των δεσμών πυριτίου δεν έχουν μοναδική τιμή αλλά παρουσιάζουν μια ποικιλία διαφορετικών τιμών σε αντίθεση με την αρκετά συγκεκριμένη δομή των κρυσταλλικών κυψελών
- Copper indium diselenide cells. Πρόκειται για κυψέλες από ένα υλικό που παρουσιάζει άμεση οπτική απορρόφηση με εξαιρετικά υψηλό συντελεστή απορρόφησης. Τα ηλεκτρικά χαρακτηριστικά του εξαρτώνται ιδιαίτερα από την αναλογία χαλκού/ίνδιου καθώς και από τον ακριβή έλεγχο της στοιχειομετρίας στην παραγωγή.
- Cadmium telluride cells. Οι κυψέλες αυτού του τύπου έχουν τα ιδανικά χαρακτηριστικά για ηλιακή απορρόφηση αλλά είναι ιδιαίτερος δύσκολες στην κατασκευή (η εναπόθεση του Cadmium telluride σε κενό).

- **Concentrator solar cell systems** : Τα συστήματα αυτά χρησιμοποιούν μια τεχνική συγκέντρωσης της ηλιακής ακτινοβολίας που λειτουργεί ως εξής : Παραβολικά κάτοπτρα συγκεντρώνουν την ακτινοβολία από μια μεγάλη περιοχή σε ένα μικρό κομμάτι φ/β στοιχείου πολύ υψηλής απόδοσης ( βλέπε GaAs ) και βέβαια υψηλότερου κόστους παραγωγής. Το χαμηλό όμως κόστος των παραβολικών σε σχέση με τα φ/β στοιχεία κάνουν οικονομικά συμφέρουσα την τεχνολογία αυτή, αν και παρουσιάζει κάποιες δυσκολίες όπως το ακριβές tracking του ήλιου σε οπτικό επίπεδο. Μπορεί να γίνει χρήση παραβόλων ή και φακών Fresnel. Τέλος για λόγους πληρότητας του κειμένου θα πρέπει να αναφερθούν και νέες υπό έρευνα κατηγορίες κυψελών όπως οι οργανικές και πλαστικές φ/β κυψέλες (organic and plastic solar cells) και οι οπτοηλεκτροχημικές (optoelectrochemical cells).

### **1.6 Η ηλιακή ακτινοβολία**

Μετά την ανάλυση του φωτοβολταϊκού φαινομένου και των χαρακτηριστικών που διέπουν την λειτουργία των φ/β στοιχείων, σειρά έχει η μελέτη της ισχύος της ακτινοβολίας που δέχονται στην επιφάνειά τους από τον ήλιο. Ο λόγος είναι απλός : Θέλουμε όσο το δυνατόν περισσότερη ηλιακή ακτινοβολία να προσπίπτει άμεσα στο φ/β πάνελ, για όσο το δυνατόν περισσότερο χρόνο κατά τη διάρκεια της ημέρας, αλλά και κατ' επέκταση καθ' όλη την διάρκεια του έτους.

Το πρόβλημα του σωστού προσανατολισμού της φ/β μονάδας σχετίζεται με την θέση της ως προς τον ορίζοντα, την θέση του ήλιου, το γεωγραφικό πλάτος της τοποθεσίας, το υψόμετρο, οι μετεωρολογικές συνθήκες, η περιεκτικότητα της ατμόσφαιρας σε υγρασία κ.ά. Συμβατικά, το μήκος της διαδρομής της ηλιακής ακτινοβολίας μέσα στην γήινη ατμόσφαιρα μέχρι την στάθμη της θάλασσας (sea level), που είναι άμεση συνέπεια της θέσης του ήλιου, χαρακτηρίζεται από μια κλίμακα μάζας αέρα (air mass) βαθμολογημένης με την τέμνουσα της ζενιθιακής απόστασης, δηλαδή της γωνίας ανάμεσα στην θέση του ήλιου και στην κατακόρυφο. Η κλίμακα αυτή δείχνει δηλαδή πόσες φορές είναι μεγαλύτερη η διαδρομή της ηλιακής ακτινοβολίας στην ατμόσφαιρα, σε σύγκριση με την κατακόρυφη διαδρομή , όπως φαίνεται στο παρακάτω σχήμα 11 :



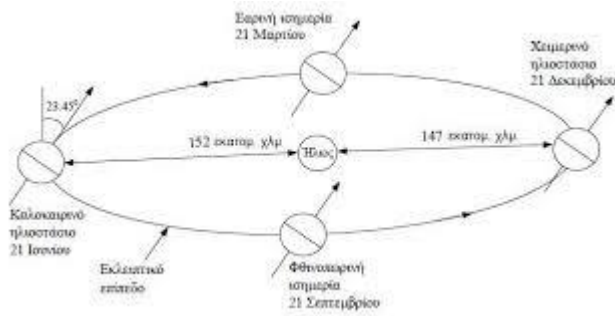
Σχήμα 11. Η εξάρτηση της κλίμακας μάζας αέρα (AM) από την θέση του ήλιου. Αριστερά απλοποιημένη απεικόνιση, δεξιά πραγματική λόγω διάθλασης στην ατμόσφαιρα

### **1.6.1. Η συμβατική ακτινοβολία του ενός πλήρους ήλιου**

Η ηλιακή ακτινοβολία Air Mass 1,5 έχει πυκνότητα ισχύος  $2\,935\text{ W m}^{-2}$  και αποτελεί μια αντιπροσωπευτική προσέγγιση της μέσης μέγιστης ισχύος που δέχεται μια επιφάνεια κάθετη προς τον ήλιο το καλοκαίρι με ευνοϊκές συνθήκες (μεσημέρι, χωρίς συννεφιά, χαμηλή ατμοσφαιρική υγρασία κ.τ.λ.) σε μεγάλο μέρος της γήινης επιφάνειας (η Ελλάδα ανήκει σε αυτό). Για ευκολία των μετρήσεων η πυκνότητα ισχύος αυτή στρογγυλοποιείται στα  $2100\text{ W m}^{-2}$  (ή αλλιώς  $2100\text{ mW cm}^{-2}$ ) και ονομάζεται συμβατικά ακτινοβολία ενός πλήρους ήλιου (προαιρετικό το πλήρης). Αξίζει να σημειωθεί ότι η απόδοση του φ/β στοιχείου εξαρτάται από τις συνθήκες Air Mass που επικρατούν: Σε συνθήκες AM2 ένα στοιχείο μπορεί να αποδίδει 16%, ενώ σε συνθήκες AM0 να φτάνει το 12%.

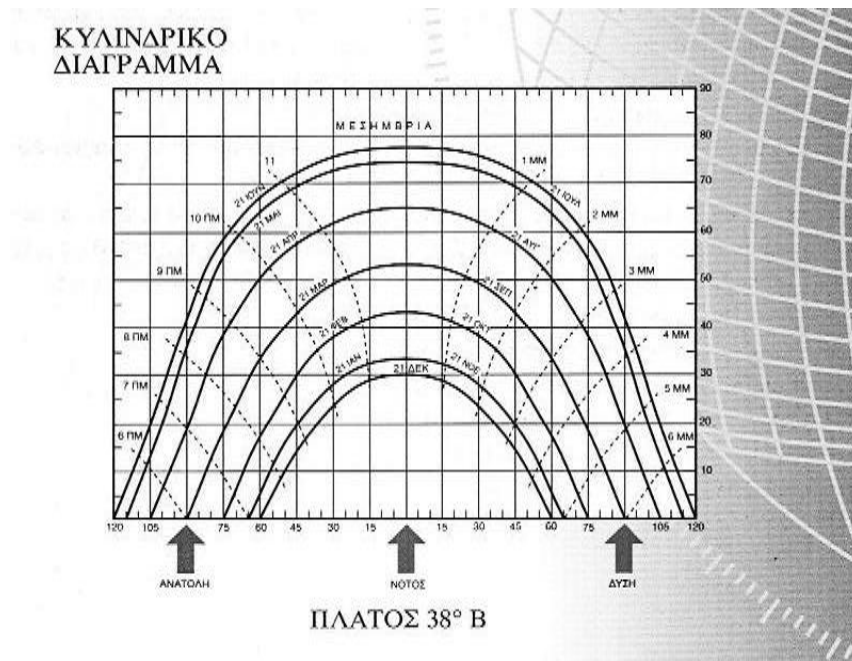
### **1.6.2 Η θέση του ήλιου στον ορίζοντα**

Ο σημαντικότερος και ταυτόχρονα πλήρως προβλεπόμενος παράγοντας που διαμορφώνει την ισχύ της ηλιακής ακτινοβολίας που προσπίπτει σε ένα φ/β πάνελ είναι η θέση του ήλιου αναφορικά με την τοποθεσία στη γη που βρισκόμαστε. Κατά τη διάρκεια του έτους η θέση του ήλιου αποκλίνει σημαντικά από την διάκεντρο γης – ήλιου. Μετρήσεις έχουν γίνει για να προκύψουν τα παρακάτω διαγράμματα που αναφέρονται σε τοποθεσία με γεωγραφικό πλάτος  $38^{\circ}$  βόρεια, πλάτος που αντιστοιχεί και στην περιοχή των Αθηνών που έλαβαν χώρα η κατασκευή και οι μετρήσεις της εργασίας αυτής. Στο παρακάτω διάγραμμα φαίνεται η ακραία απόκλιση του ήλιου κατά το θερινό και το χειμερινό ηλιοστάσιο:



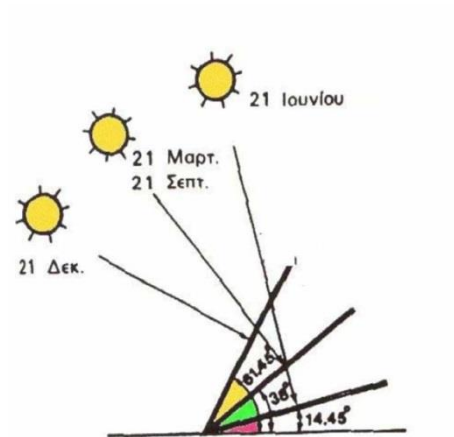
Σχήμα 11. Η ακραία απόκλιση του ήλιου κατά το θερινό και χειμερινό ηλιοστάσιο

Ακολουθεί ένα πολύ σημαντικό διάγραμμα της τροχιάς του ήλιου στον ουρανό κατά τη διάρκεια της ημέρας για διάφορες εποχές. Η αναφορά της ώρας δεν είναι το ωρολογιακό αλλά το ηλιακό μεσημέρι :



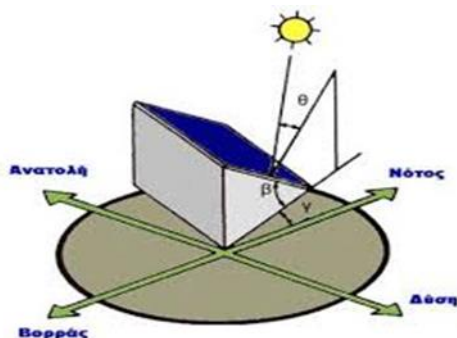
Σχήμα 12. Διάγραμμα της ακραίας απόκλισης του ήλιου

Τέλος, στο παρακάτω διάγραμμα φαίνεται η βέλτιστη κλίση του  $\phi/\beta$  συλλέκτη για τις διαφορετικές ημερομηνίες ως προς τον ορίζοντα :



Σχήμα 13 Βέλτιστη κλίση  $\Phi/\beta$  συλλέκτη

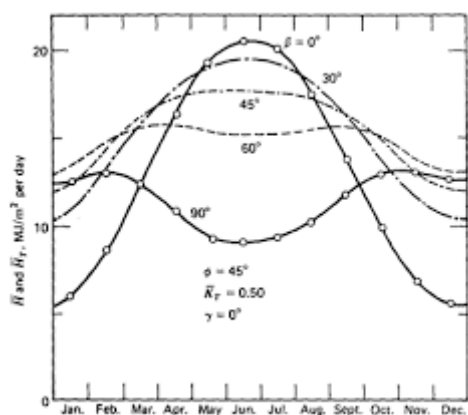
Συνέπεια της απόκλισης αυτής του ήλιου είναι να διαγράφονται ηλιακές τροχιές βορειότερα στον ουρανό το καλοκαίρι, με νωρίτερη ανατολή και αργότερη δύση στο βόρειο ημισφαίριο, ενώ αντίθετα συμβαίνει το χειμώνα. Για να έχουμε σωστή αναφορά της θέσης του ήλιου ορίζουμε δυο γωνίες : το ύψος ( $\beta$ ) και το αζιμούθιο ( $\theta$ ), όπως φαίνονται στο παρακάτω σχήμα. Μαζί φαίνεται και ο προσανατολισμός του φ/β συλλέκτη με τις χαρακτηριστικές του γωνίες κλίση ( $\beta$ ) και αζιμούθια γωνία συλλέκτη ( $\theta$ ) :



Σχήμα 14. Ύψος και αζιμούθιο για σωστή αναφορά της θέσης του ήλιου

### **1.6.3 Η βέλτιστη κλίση του συλλέκτη**

Αναφερόμενος στο συμβολισμό του παραπάνω σχήματος, η βέλτιστη κλίση του φ/β συλλέκτη ( $\beta$ ) στο βόρειο ημισφαίριο, για όλη τη διάρκεια του έτους είναι ίση με τον γεωγραφικό παράλληλο της τοποθεσίας, ενώ η βέλτιστη αζιμούθια γωνία είναι ίση με μηδέν. Παρατήρηση : Η άνω βέλτιστη κλίση του φ/β συλλέκτη ( $\beta$ ) αναφέρεται καθ' όλη την διάρκεια του έτους και εφαρμόζεται σε φ/β εγκαταστάσεις με σταθερά εγκατεστημένους τους συλλέκτες. Συνηθίζεται όμως στις εγκαταστάσεις (όπου είναι δυνατό) να είναι ρυθμιζόμενη αυτή η κλίση, και ο λόγος είναι ο εξής : Λόγω της μεταβολής της απόκλισης του ήλιου στη διάρκεια του έτους, η βέλτιστη κλίση των συλλεκτών είναι διαφορετική για κάθε εποχή και κατ' επέκταση για κάθε μήνα. Αυτό είναι φανερό στο παραπάνω σχήμα, που αναφέρεται σε περιοχή με γεωγραφικό πλάτος 45ο και διάφορες κλίσεις του συλλέκτη.



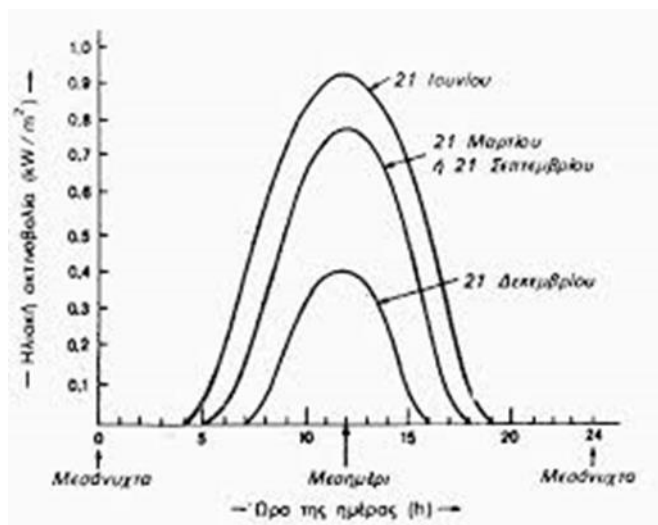
Σχήμα 15. Γεωγραφικό πλάτος

Στο παραπάνω σχήμα είναι φανερό ότι να μεν μια κλίση με το γεωγραφικό πλάτος είναι η βέλτιστη για σταθερούς συλλέκτες, αλλά αξίζει τον κόπο να ρυθμίζουμε την κλίση ανάλογα με την εποχή. Πώς; Μα από τα σχήματα 15 και 17 βλέπουμε την κατεύθυνση των αλλαγών. Συνηθίζεται δε σε συστήματα ρυθμιζόμενης κλίσης, μια κλίση 10ο έως 15ο μικρότερη από τον παράλληλο της τοποθεσίας για το καλοκαίρι, ενώ αντίστοιχα μεγαλύτερη για τον χειμώνα. Οι ακραίες θέσεις του ήλιου σε θερινό και χειμερινό ηλιοστάσιο είναι αυτές που φαίνονται στο σχήμα 13 με τις αντίστοιχες βέλτιστες θέσεις των συλλεκτών. Για να κλείσει αυτό το κομμάτι, αξίζει να σημειωθεί ότι η ακτινοβολία που τίθεται υπό συζήτηση μέχρι στιγμής είναι η άμεση από τον ήλιο (σαν δέσμη) και όχι το διάχυτο φως της ατμόσφαιρας. Για παράδειγμα σε περιοχές με πολύ υγρό κλίμα (τροπική ζώνη) λόγω της υψηλής συγκέντρωσης σταγονιδίων νερού στην ατμόσφαιρα, σημαντικό μέρος της ηλιακής ακτινοβολίας διαχέεται στην ατμόσφαιρα. Για τον λόγο αυτό μικρότερη κλίση των φ/β συλλεκτών, ώστε να αντικρίζουν περισσότερο τον ουρανό και να δέχονται αφθονότερα την διάχυτη ακτινοβολία.

#### **1.6.4 Η ένταση της ηλιακής ακτινοβολίας**

Ένα βασικό πρόβλημα της χρήσης της ηλιακής ακτινοβολίας για παραγωγή ηλεκτρικής ενέργειας είναι ότι παρουσιάζει έντονη διακύμανση και απότομες, απρόβλεπτες μεταβολές στην εξέλιξη του χρόνου σε σχέση με άλλα συστήματα μετατροπής ενέργειας. Παρακάτω φαίνεται ένα διάγραμμα της διακύμανσης της έντασης της

ηλιακής ακτινοβολίας σε μια οριζόντια επιφάνεια υπό καθαρό ουρανό για διαφορετικές ενδεικτικές ημερομηνίες ( ηλιοστάσια και ισημερίες ):



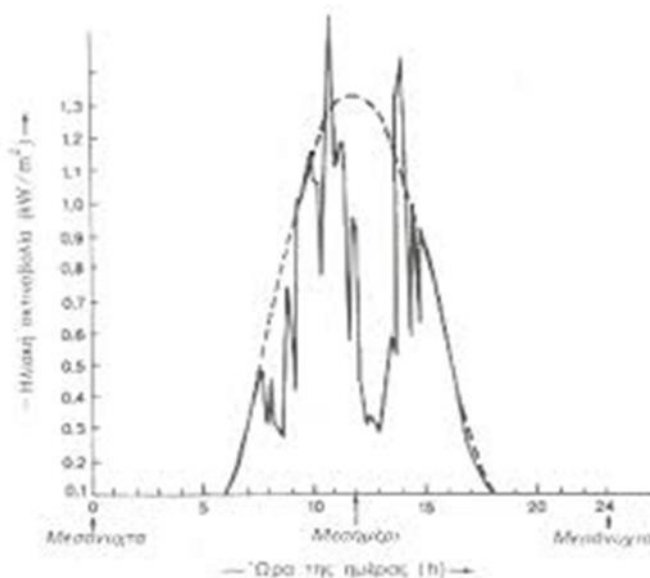
Σχήμα 16. Διάγραμμα της ηλιακής ακτινοβολίας

Ένας άλλος τρόπος να βελτιώσουμε την ένταση της ηλιακής ακτινοβολίας πουδέχεται μια επιφάνεια (εν προκειμένω η  $\phi/\beta$  μονάδα) είναι να την τοποθετήσουμε πάνω σε μια διάταξη κινητού πλαισίου που περιστρέφεται κατά διαστήματα, παρακολουθώντας την πορεία του ήλιου στον ουρανό. Με αυτό τον τρόπο εξασφαλίζουμε την κάθετη πρόσπτωση (σχετικής ακρίβειας) της ακτινοβολίας στην επιφάνεια του φωτοβολταϊκού, με αποτέλεσμα την αύξηση της παραγόμενης ηλεκτρικής ισχύος. Οι διατάξεις αυτές ονομάζονται στα ελληνικά  $\phi/\beta$  ηλιοστάτες (διεθνής όρος : sun trackers/solar trackers). Η ακριβής απόδοση του διεθνούς όρου είναι παρακολουθητής του ήλιου. Με την χρήση του solar tracker επιτυγχάνουμε από την μια αύξηση της παραγόμενης ηλεκτρικής ισχύος, από την άλλη όμως καταναλώνουμε ισχύ για την κίνηση της παρακολουθήσης. Ο υπολογισμός των ισχύων αυτών είναι που δείχνει αν η χρήση των trackers είναι ενεργειακά αποδοτική. Αυτή την μελέτη διαπραγματεύεται η εργασία αυτή για ένα φωτοβολταϊκό σύστημα μικρής ισχύος. Βέβαια υπάρχουν και άλλα σημαντικά θέματα για την επιλογή ενός συστήματος ηλιοστάτη όπως το κόστος αυτού, η ανάγκη συντήρησης και η δυνατότητα απόσβεσης του επιπλέον κόστους που αυτό επιφέρει στο φωτοβολταϊκό σύστημα. Με αυτά τα θέματα δεν ασχολείται η εργασία αυτή.



### 1.6.5 Η επίδραση των νεφών

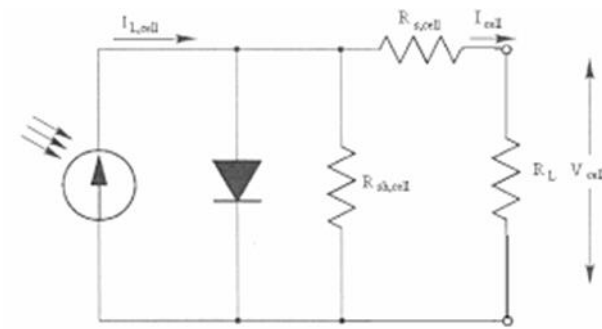
Η παρουσία νεφώσεων έχει άμεση επίδραση στην ένταση της ηλιακής ακτινοβολίας που φτάνει στην επιφάνεια της γης, εξασθενώντας την απευθείας ακτινοβολία και αυξάνοντας μερικώς την διάχυτη. Τα παραπάνω ισχύουν σε αραιές νεφώσεις, σε αντίθεση περίπτωση (πυκνή νέφωση) μπορεί να αποκοπεί το μεγαλύτερο μέρος της ακτινοβολίας. Η επίπτωση των νεφών στην ένταση της ακτινοβολίας φαίνεται στο παρακάτω σχήμα :



Σχήμα 17. Διάγραμμα με την επίπτωση των νεφών στην ηλιακή ακτινοβολία

### 1.7 Τα ηλεκτρικά χαρακτηριστικά των φ/β στοιχείων

Σε μια πρώτη προσέγγιση του φ/β στοιχείου μπορούμε να το θεωρήσουμε ως μια πηγή ρεύματος ελεγχόμενη από δίοδο. Προσθέτοντας όμως τις αναπόφευκτες αντιστάσεις  $R_s$  που παρεμβάλλονται στην κίνηση των φορέων μέσα στον ημιαγωγό και την παράλληλη αντίσταση  $R_{sh}$  λόγω διαρροών ρεύματος της διόδου, καταλήγουμε στο ισοδύναμο ηλεκτρικό κύκλωμα του φωτοβολταϊκού στοιχείου :

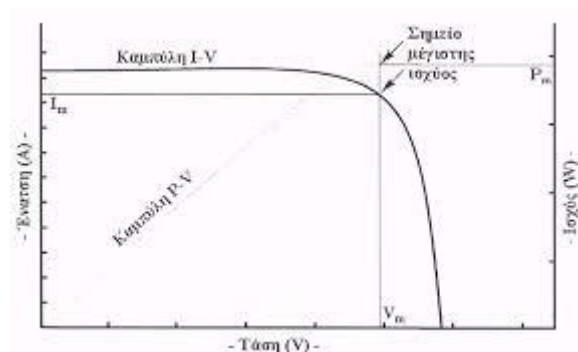


Σχήμα 18.Ισοδύναμο κύκλωμα Φ/Β στοιχείου

Από τα βασικά ηλεκτρικά χαρακτηριστικά των φωτοβολταϊκών στοιχείων είναι :

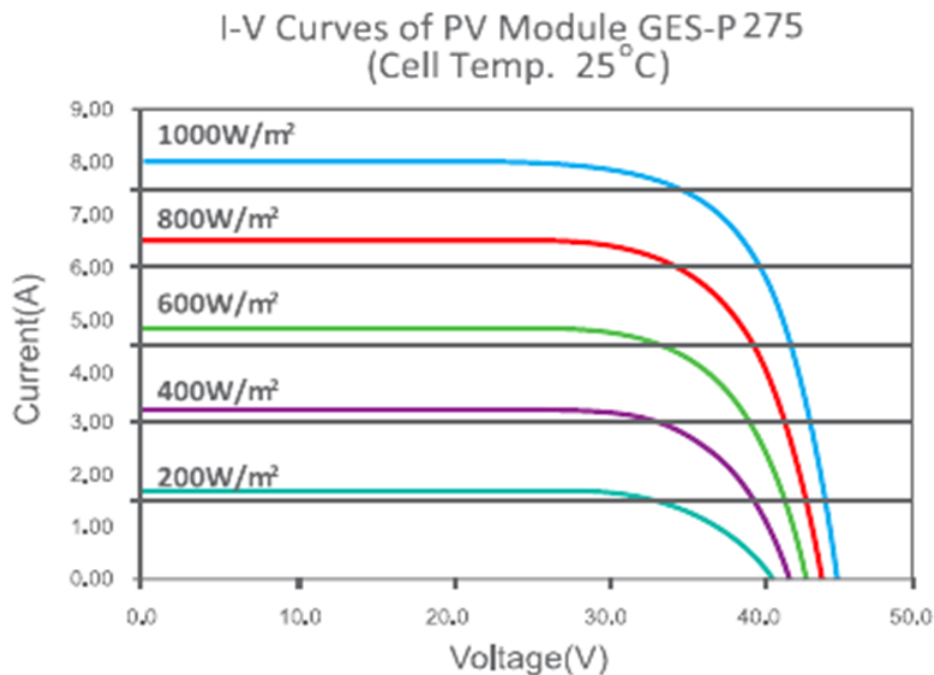
- A) Το ρεύμα βραχυκύκλωσης  $I_{sc}$  , όταν η τάση στα άκρα του φ/β μηδενιστεί
- B) Η τάση ανοικτοκύκλωσης  $V_{oc}$  , όταν το ρεύμα που διαρρέει το φ/β είναι μηδενικό
- Γ) Το ρεύμα και η τάση υπό φορτίο ( ,load load I V )
- Δ) Ο συντελεστής της θερμοκρασιακής διόρθωσης της απόδοσης (σθ)

Η βασική καμπύλη I-V χαρακτηριστικών μιας φωτοβολταϊκής μονάδας υπό συνθήκες δεδομένης ακτινοβολίας καθώς και η καμπύλη P-V φαίνονται στο παρακάτω σχήμα :



Σχήμα 19.Καμπύλες I-V και P-V μιας φωτοβολταϊκής μονάδας

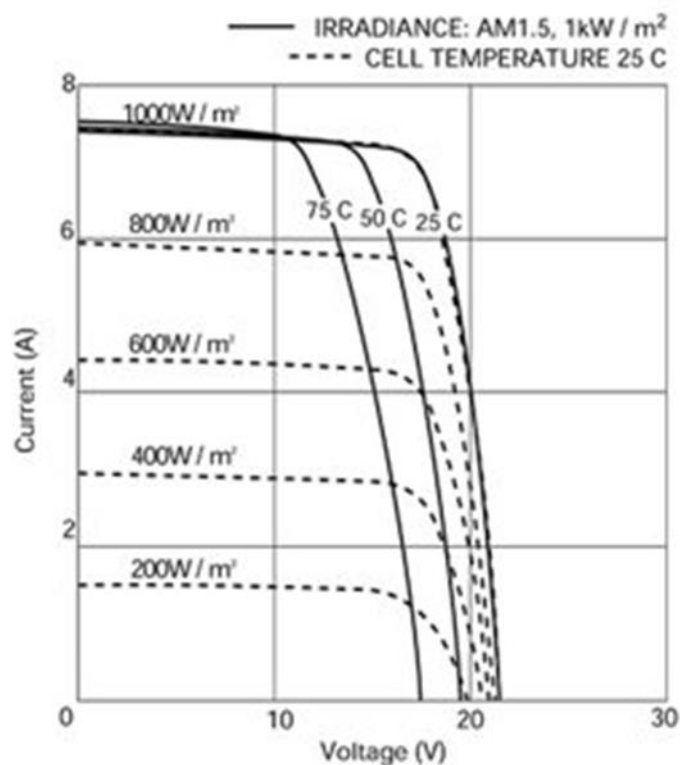
Στο σχεδιάγραμμα αυτό φαίνεται και το σημείο μέγιστης ισχύος υπό τις δεδομένες συνθήκες ακτινοβολίας. Για να εκμεταλλευτούμε την ισχύ που μπορεί να παρέχει το φ/β στοιχείο μας πρέπει να έχουμε φορτίο με αντίσταση  $R_{load} = V_m / I_m$ . Εδώ γίνεται η εξής ερώτηση: Πως αλλάζει το διάγραμμα I-V υπό διαφορετικές συνθήκες ακτινοβολίας; Αυτό φαίνεται στο παρακάτω σχήμα :



Σχήμα 20. Διάγραμμα I-V υπό διαφορετικές συνθήκες ακτινοβολίας

Βλέποντας αυτό το σχήμα καταλαβαίνουμε αμέσως το εξής σημαντικό πρόβλημα: Τα σημεία μέγιστης ισχύος δεν μπορούν να ικανοποιηθούν από ένα δεδομένο φορτίο για κυμαινόμενη ακτινοβολία. Αυτό το πρόβλημα μπορεί να αντιμετωπιστεί με τη χρήση ρυθμιστικών διατάξεων που φέρνουν συνεχώς την τάση και το ρεύμα σε ευνοϊκές τιμές για τις συνθήκες κάθε χρονικής στιγμής. Οι διατάξεις αυτές καλούνται MPPT (Maximum Power Point Trackers δηλαδή παρακολουθητές σημείου μέγιστης ισχύος). Επίσης, από το σχήμα παραπάνω μπορούμε να συμπεράνουμε ότι με την αύξηση της ακτινοβολίας αυξάνεται η  $V_{oc}$  και το  $I_{sc}$ , αν και το  $I_{sc}$  είναι πιο ευαίσθητο στις μεταβολές της ακτινοβολίας από την  $V_{oc}$ . Ακόμα παρατηρούμε ότι η ένταση του ρεύματος παραμένει περίπου σταθερή για το μεγαλύτερο τμήμα του διαγράμματος (υπό συνθήκες σταθερής ακτινοβολίας) λειτουργώντας περίπου ως πηγή ρεύματος. Το τελικό συμπέρασμα από την ηλεκτρική ανάλυση της φ/β μονάδας είναι η σημαντική

εξάρτηση της λειτουργίας από τους εξής τρεις παράγοντες : -Την ένταση της ηλιακής ακτινοβολίας που προσπίπτει στο φωτοβολταϊκό. -Την θερμοκρασία του στοιχείου. - Την αντίσταση του κυκλώματος που λειτουργεί ως φορτίο του φ/β. Όσον αφορά την θερμοκρασία του στοιχείου , θα περίμενε κανείς ότι με την αύξηση της περιβάλλουσας θερμοκρασίας λειτουργούν καλύτερα τα φωτοβολταϊκά (από διαίσθηση και μόνο). Αυτό όμως σε καμία περίπτωση δεν ισχύει όπως φαίνεται και στο παρακάτω σχήμα :



Σχήμα 21. Διάγραμμα I-V υπό διαφορετικές συνθήκες θερμοκρασίας

Ο βασικός λόγος που συμβαίνει αυτό είναι ότι με την αύξηση της θερμοκρασίας μειώνεται η κινητικότητα των φορέων του ηλεκτρικού ρεύματος.

## **1.8 Από τι αποτελείται ένα φωτοβολταϊκό σύστημα**

Ένα πλήρες φωτοβολταϊκό σύστημα αποτελείται από τα εξής υποσυστήματα

α) Την φωτοβολταϊκή γεννήτρια με την μηχανική στήριξη αυτής και ίσως ένα σύστημα παρακολούθησης της ηλιακής τροχιάς μαζί .

β) Το υποσύστημα αποθήκευσης (διάταξη μπαταριών) .

γ) Τον εξοπλισμό ελέγχου και προσαρμογής ισχύος (και συστήματα μετρήσεων και παρακολούθησης αν είναι αναγκαία) .

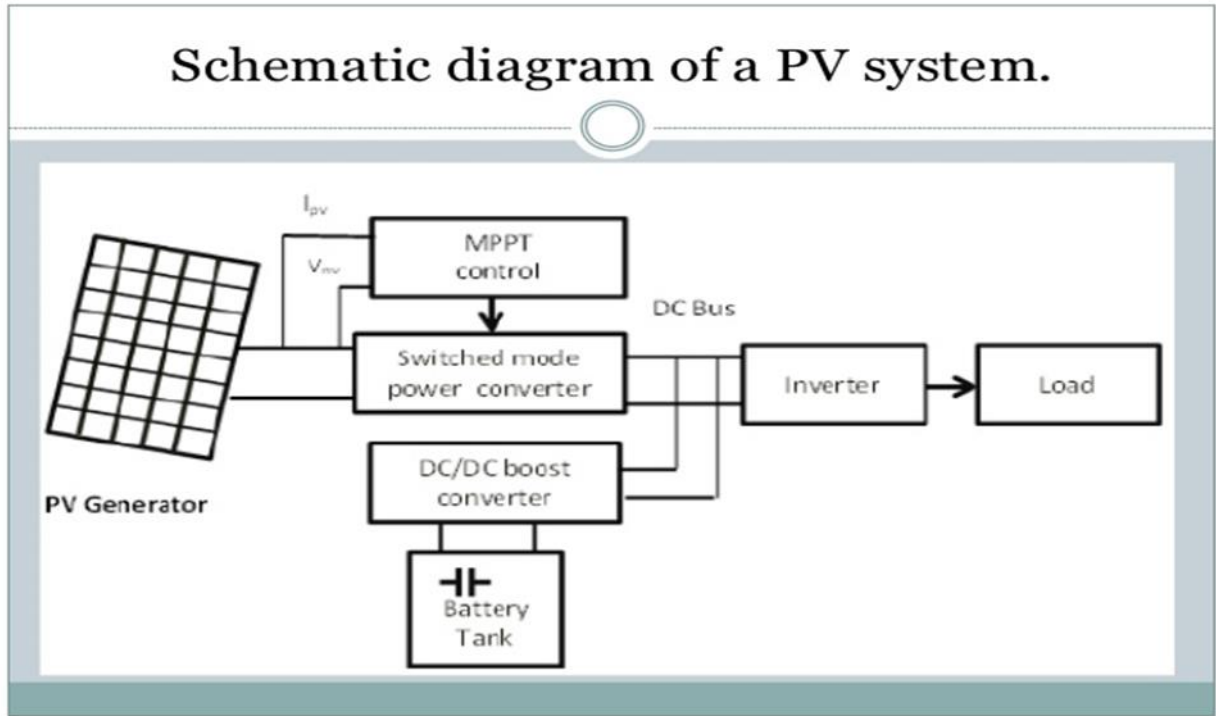
δ) Βοηθητικό σύστημα παροχής ισχύος για περίπτωση ανάγκης (όπως γεννήτρια πετρελαίου) , δημιουργώντας έτσι ένα υβριδικό αυτόνομο σύστημα παροχής ηλεκτρικής ισχύος . Η φωτοβολταϊκή γεννήτρια : Τα φωτοβολταϊκά στοιχεία (προαναφέρθηκαν ως κυψέλες – solar cells) παράγουν πολύ μικρή ισχύ, γι' αυτό οι κατασκευαστές τα συνενώνουν φτιάχνοντας τις φ/β μονάδες (PV modules). Συνδεδεμένες μονάδες δημιουργούν αυτό που κυκλοφορεί και στην αγορά με το όνομα φωτοβολταϊκό πάνελ (αλλιώς γνωστοί και ως συλλέκτες-PV panel). Τέλος στις μεγάλες εγκαταστάσεις παροχής ρεύματος από φ/β ενέργεια εμφανίζονται πάνελ συνδεδεμένα μεταξύ τους και πάνω σε μεγάλες μεταλλικές κατασκευές που καλούνται φ/β συστοιχίες (PV arrays) . Η αποθήκευση της ενέργειας : Η αποθήκευση της ενέργειας είναι απαραίτητη για τα φ/β συστήματα λόγω της έντονης διακύμανσης της ηλιακής ακτινοβολίας στη διάρκεια του χρόνου αλλά και της ημέρας. Υπάρχουν πολλές μορφές αποθήκευσης της ενέργειας. Ωστόσο τα περισσότερα συστήματα σήμερα χρησιμοποιούν μπαταρίες (συσσωρευτές) και μάλιστα τύπου lead-acid. Η ισχύς που αποθηκεύεται σε ένα συσσωρευτή, δηλαδή η χωρητικότητα του μετράται σε αμπερώρια (Ah). Με γνώμονα την ονομαστική του τάση υπολογίζουμε την μέγιστη δυνατή αποθήκευση ενέργειας με τον τύπο :  $V \text{ Ah} \cdot (\ )$  ον, π.χ. μια μπαταρία των 12Volts, 100Ah, έχει ονομαστική ικανότητα αποθήκευσης ηλεκτρικής ενέργειας 1,2kWh. Επίσης πρέπει να σημειωθεί ότι οι φορτισμένοι συσσωρευτές αυτοεκφορτίζονται με ρυθμό 2-5% τον μήνα. Τέλος πρέπει να αναφερθούμε στο βάθος της εκφόρτισης του συσσωρευτή, από το οποίο εξαρτάται η διάρκεια της χρήσιμης ζωής του. Συσσωρευτής με βάθος εκφόρτισης 10% πρέπει να χρησιμοποιεί μόνο το 10% της ονομαστικής του χωρητικότητας, αν δεν θέλουμε την πρόωρη καταστροφή του.

Εξοπλισμός ελέγχου:

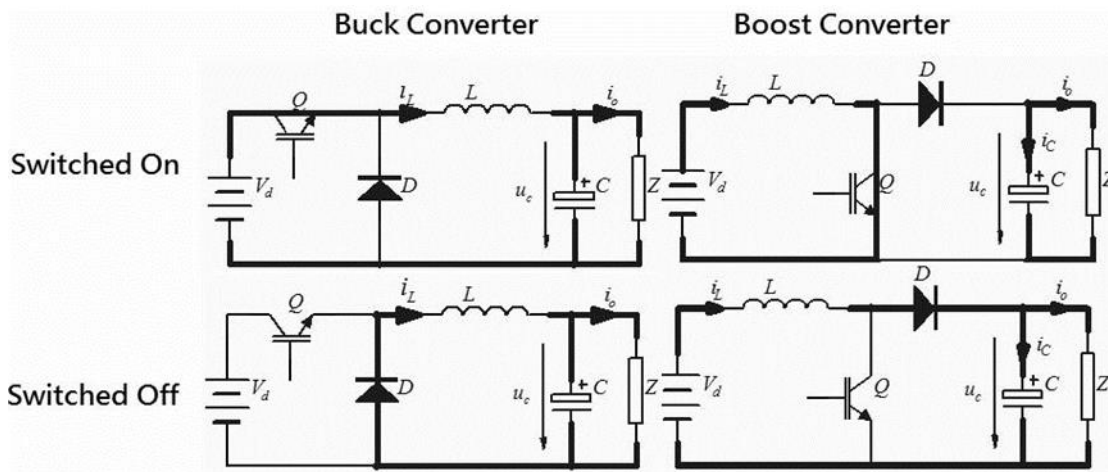
- Blocking Diode (δίοδος φραγής) : Όπως έχει αναλυθεί στο φωτοβολταϊκό φαινόμενο, τα φ/β στοιχεία συμπεριφέρονται ως δίοδοι στο σκοτάδι (την νύκτα). Για το λόγο αυτό τοποθετούμε μια δίοδο φραγής ανάμεσα στο στοιχείο και τον συσσωρευτή για να αποτρέψουμε την αποφόρτιση του τελευταίου.

-Charge regulator (ρυθμιστής φόρτισης) : Ο ρυθμιστής φόρτισης περιορίζει την μέγιστη τάση στην μπαταρία για να αποτρέψει την υπερφόρτωση. Δυο είναι οι βασικοί τύποι ρυθμιστών, ο εν σειρά και ο εν παραλλήλω. Είναι προφανής η κατανάλωση ισχύος από τις μεταβλητές αντιστάσεις των δυο παραπάνω κυκλωμάτων. Η κατανάλωση αυτή μπορεί να μειωθεί σημαντικά αν αντικατασταθεί η εν σειρά μεταβλητή αντίσταση του δεύτερου κυκλώματος από ένα διακόπτη (είτε μηχανικό αλλά συνήθως ένα MOSFET ή ένα BJT ).

- Οι διατάξεις MPPT (Maximum Power Point Trackers ) : Όπως έχει εξηγηθεί και στο κεφάλαιο για τις χαρακτηριστικές I-V των φωτοβολταϊκών στοιχείων, για μεταβαλλόμενες συνθήκες ακτινοβολίας δεν μπορούμε με δεδομένο φορτίο να είμαστε συνεχώς στην θέση μέγιστης ισχύος στο διάγραμμα. Ο MPPT μας δίνει την δυνατότητα αυτή : Έχοντας συνεχώς σταθερό φορτίο, ξεγελά το φωτοβολταϊκό στοιχείο ότι μεταβάλλεται το φορτίο συνεχώς στην τιμή που αντιστοιχεί στο σημείο μέγιστης ισχύος για την λαμβανόμενη ακτινοβολία κάθε χρονικής στιγμής. Για το σκοπό αυτό υπολογίζεται κάθε χρονική στιγμή η τάση και το ρεύμα του κυκλώματος, ενώ ένας υπολογιστής υπολογίζει τις αλλαγές των παραμέτρων ενός boost converter (dc/dc converter βλέπε από κάτω) για να επιτύχει την παρακολούθηση του μεγίστου. Ένα μπλοκ διάγραμμα της λειτουργίας αυτής φαίνεται παρακάτω :



Σχήμα 22. Μπλοκ διάγραμμα ενός Maximum Power Point Tracker



Σχήμα 23. Buck Converter Boost Converter

-DC/DC Converters (μετατροπείς συνεχούς/συνεχούς) : Μετατρέπουν την ισχύ του φ/β πάνελ σε συνεχή άλλης στάθμης. Βασικότερες τοπολογίες είναι οι buck και boost converters.

-DC/AC converters-inverters (μετατροπείς συνεχούς/εναλλασσομένου) : Χρησιμοποιούνται όταν απαιτείται σύνδεση εναλλασσομένου φορτίου στο φ/β σύστημα. Η απόδοση του μετατροπέα μπορεί να φτάσει και το 97% αρκεί να λειτουργεί στην ονομαστική του ισχύ. Αν λειτουργεί υπό μερικό φορτίο η απόδοση του μπορεί να πέσει σημαντικά. Το φ/β σύστημα μπορεί με χρήση inverter να συνδεθεί (υπό προϋποθέσεις) στο δημόσιο δίκτυο παροχής ρεύματος, λειτουργώντας ως πηγή ηλεκτρικής ισχύος για αυτό.

## **ΚΕΦΑΛΑΙΟ 2**

### **ΘΕΩΡΙΑ Φ/Β ΗΛΙΟΣΤΑΤΗ**

#### **2.1 Εισαγωγή**

Αφού έχουν αναφερθεί όλα όσα χρειάζονται από θεωρητικής άποψης για την φωτοβολταϊκή τεχνολογία, ήρθε η ώρα να δούμε με τι ασχολείται η παρούσα εργασία. Σκοπός της είναι η κατασκευή ενός photovoltaic solar tracking system (φ/β παρακολουθητής της τροχιάς του ήλιου). Υπάρχουν διάφοροι τρόποι να υλοποιηθεί η εφαρμογή αυτή με ή χωρίς αισθητήρα ανίχνευσης της θέσης του ήλιου, ανοικτού ή κλειστού βρόχου ελέγχου. Η κατασκευή του εν λόγω solar tracker βασίζεται σε έλεγχο κλειστού βρόχου (με ανάδραση) χωρίς χρήση επιπλέον αισθητήρα, καθώς το ίδιο το φωτοβολταϊκό πάνελ θα χρησιμοποιηθεί για να ανιχνευθεί η θέση του ήλιου. Όπως έχει προαναφερθεί στην θεωρία, στόχος των solar trackers είναι να βρίσκουν την θέση του ήλιου στον ουρανό, μεγιστοποιώντας έτσι την ηλιακή ακτινοβολία που προσπίπτει πάνω στην επιφάνεια του φωτοβολταϊκού πάνελ. Η συγκεκριμένη κατασκευή χρησιμοποιεί τον σκοπό του tracking (που είναι η κίνηση του πάνελ) ως μέσο για την εύρεση του προσανατολισμού που πρέπει να έχει η κατασκευή για να επιτύχει μεγιστοποίηση της προσπίπτουσας ακτινοβολίας. Με τον τρόπο αυτό περιορίζουμε το αρχικό κόστος κατασκευής, αποφεύγοντας την αγορά ή κατασκευή ενός αρκετά ακριβού αισθητήρα. Πώς όμως γίνεται η ανίχνευση της θέσης του ήλιου στον ορίζοντα; Υπάρχει η μέθοδος να χρησιμοποιηθεί ένα μοντέλο της κίνησης του ήλιου, αποθηκευμένο στην μνήμη ενός υπολογιστή και ένα πρόγραμμα να εκτελεί έναν



αλγόριθμο διαδοχικών προβλεπόμενων θέσεων του ήλιου, ενεργοποιώντας τα μοτέρ κίνησης του πλαισίου της κατασκευής. Και πάλι όμως είναι αναγκαία η χρήση οπτικών κωδικοποιητών ελέγχου θέσης των μοτέρ (σήματα ανάδρασης ελέγχου), για τυχούσες αποκλίσεις της κίνησης των μοτέρ λόγω εξωτερικών παραγόντων (σφάλματα, ολίσθηση, αέρας, γήρανση των ηλεκτρικών και μηχανικών μερών). Στην παρούσα εργασία δεν χρησιμοποιείται κανείς αισθητήρας (ούτε ηλιακός, ούτε θέσης των μοτέρ), παρά μόνο το ίδιο το φωτοβολταϊκό πάνελ. Δημιουργείται συνεπώς το ερώτημα πώς γίνεται η ανίχνευση του μεγίστου της έντασης της ηλιακής ακτινοβολίας στον ουρανό; Η εξήγηση ακολουθεί στην επόμενη παράγραφο.

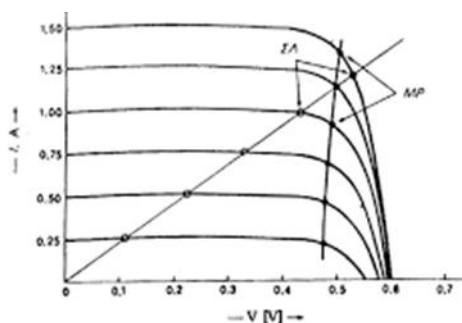
## **2.2 Η εύρεση του μεγίστου της έντασης της ηλιακής ακτινοβολίας**

Η λειτουργία του αλγορίθμου εύρεσης του μεγίστου της έντασης της ηλιακής ακτινοβολίας στον ουρανό βασίζεται σε μια τεχνική μερικής σάρωσης του ουρανού σε δύο άξονες, τους άξονες κίνησης των μοτέρ του solar tracker. Πρόκειται για την οριζόντια αζιμουθιακή κίνηση (θα αναφέρεται στο εξής ως κίνηση West/East δηλαδή Δύση/Ανατολή) και την ζενιθιακή κίνηση, γνωστή και ως ανύψωση (θα αναφερόμαι στο εξής σε αυτήν ως κίνηση Up/Down, δηλαδή Πάνω/Κάτω). Αυτό το πρόβλημα εύρεσης του μεγίστου είναι κυρτό, δυο μεταβλητών. Η κυρτότητα είναι προφανής λόγω της φύσης του γεωμετρικού προβλήματος και συνεπώς η εύρεση του μεγίστου ταυτίζεται με το πρόβλημα εύρεσης τοπικού μεγίστου (στα κυρτά προβλήματα το τοπικό ακρότατο είναι και ολικό). Επίσης λόγω της κυρτότητας μπορούμε να αναλύσουμε την αναζήτηση μεγίστου σε δυο βήματα :

- 1) Αναζήτηση και εύρεση του μεγίστου στον άξονα West/East (ορίζεται ως max\_we)
  - 2) Αναζήτηση και εύρεση του μεγίστου του max\_we στον άξονα Up/Down (ορίζεται ως max\_ud)
- Η θεωρία των κυρτών προβλημάτων αναζήτησης ακροτάτων μας εξασφαλίζει ότι:  $max_{total} = max_{ud} = max(max_{we})$  Δηλαδή ότι ο προσανατολισμός της μέγιστης ακτινοβολίας μπορεί να βρεθεί αν κινούμενοι αζιμουθιακά (με σταθερή ανύψωση) για κάποια γωνία ανύψωσης εντοπίσουμε το αρχικό μέγιστο, σταματήσουμε σε αυτό και κατόπιν κινηθούμε ζενιθιακά (πάνω-κάτω) μέχρι να βρούμε το μέγιστο του μεγίστου. Έτσι λοιπόν έχει λυθεί από μαθηματικής σκοπιάς η εύρεση του βέλτιστου προσανατολισμού του πάνελ και αρκεί να υλοποιηθεί ο βασικός αλγόριθμος που να συγκλίνει σε αυτόν.

### 2.3 Η επιλογή του φορτίου και η χρήση της ηλεκτρικής ισχύος για την εύρεση του μεγίστου

Όπως αναφέρθηκε και στην προηγούμενη παράγραφο, στην υλοποίηση του αλγορίθμου για την εύρεση της θέσης του ήλιου χρησιμοποιήθηκε το κριτήριο της μεγιστοποίησης της ηλεκτρικής ισχύος που αυτό παράγει. Αυτός ασφαλώς είναι και ο κυρίαρχος στόχος του solar tracker από κατασκευής του: Να μεγιστοποιήσει την ισχύ που παράγει το φωτοβολταϊκό πάνελ, ακόμα και αν η θέση αυτού δεν ευθυγραμμίζεται με τον ήλιο. Όμως η καθετοποίηση του πάνελ ως προς την κατεύθυνση της ηλιακής ακτινοβολίας, αυξάνει την ένταση της ακτινοβολίας που προσπίπτει σε αυτό και το αποτέλεσμα αυτού στο διάγραμμα I-V φαίνεται παρακάτω :



Σχ.11. Η μετατόπιση των σημείων λειτουργίας (βλ) ενός Φ/β στοιχείου περίπου και η απομάκρυνση του από τα αντίστοιχα σημεία της μέγιστης ισχύος (MP) της καμπύλης έντασης (I) -τάσης (V) για διαφορετικές συνθήκες ακτινοβολίας αλλά για σταθερή τιμή της αντίστασης του κυκλώματος.

Σχήμα 24. Διάγραμμα I-V για σταθερό φορτίο και μεταβαλλόμενες συνθήκες ακτινοβολίας

Από αυτό μπορούμε να διαπιστώσουμε ότι για δεδομένο φορτίο (στην εργασία αυτή μια ωμική αντίσταση ισχύος) οποιασδήποτε τιμής, αύξηση της προσπίπτουσας ακτινοβολίας συνεπάγεται και αύξηση της ισχύος που διατίθεται στο φορτίο. Άρα η μέθοδος του αλγορίθμου έχει λογική συνοχή με την χρήση της ηλεκτρικής ισχύος στην θέση της έντασης της ηλιακής ακτινοβολίας. Όμως από το σχήμα 2.1 παρατηρεί κανείς ότι για μια συγκεκριμένη μεταβολή της ακτινοβολίας δεν έχουμε την ίδια ευαισθησία για διαφορετικές τιμές του φορτίου. Δηλαδή :

-Στην οριακά μεγάλη τιμή φορτίου ( $R_l \text{ o } a \text{ d} \rightarrow \infty$ ) βλέπουμε ότι μεγάλες μεταβολές στην ένταση της προσπίπτουσας ακτινοβολίας επιφέρουν εξαιρετικά μικρές μεταβολές

στην τάση Voc, έχουμε δηλαδή πολύ μικρή ευαισθησία για να ανιχνεύσουμε την μέγιστη ένταση ακτινοβολίας.

-Στην οριακά μικρή τιμή φορτίου ( $I = 0$ ) βλέπουμε ότι μικρές μεταβολές στην ένταση της προσπίπτουσας ακτινοβολίας επιφέρουν εξαιρετικά μεγάλες μεταβολές στο ρεύμα Isc, έχουμε δηλαδή μεγάλη ευαισθησία για να ανιχνεύσουμε την μέγιστη ένταση ακτινοβολίας.

Έχει αναφερθεί παραπάνω ότι στην εργασία αυτή χρησιμοποιείται ένα απλό ωμικό φορτίο ισχύος γιατί σκοπός είναι η μελέτη της κατασκευής του solar tracker και όχι η ολοκληρωμένη εφαρμογή με battery charge control, converters κ.τ.λ. Έχοντας αυτό ως βάση, είναι δυνατό να χρησιμοποιήσουμε μόνο την τάση υπό φορτίο για τον υπολογισμό της ισχύος που θα δέχεται ο μικροελεγκτής για την εκτέλεση του αλγορίθμου :  $P_{\text{electrical}} = V_{\text{load}} \cdot I_{\text{load}} = (V_{\text{load}}^2) / R_{\text{load}}$  κάνοντας έτσι πιο εύκολη την υλοποίηση της μετατροπής των αναλογικών σημάτων σε ψηφιακά από τον μετατροπέα του μικροελεγκτή. Έτσι λοιπόν, ενώ θεωρητικά η βελτιστοποίηση του αλγορίθμου αναφέρεται στην ένταση της προσπίπτουσας ηλιακής ακτινοβολίας, κατέληξε (με όλες τις υποθέσεις που προηγήθηκαν) στην μεγιστοποίηση μονάχα μιας τάσης, της  $V_{\text{load}}$ . Έχουμε δηλαδή την εξής λογική ακολουθία υλοποίησης για τον solar tracker : (Θέση ήλιου) → (Ανίχνευση προσανατολισμού μέγιστης έντασης προσπίπτουσας ακτινοβολίας) → (Ανίχνευση προσανατολισμού μέγιστης ηλεκτρικής ισχύος) → (Ανίχνευση προσανατολισμού μέγιστης  $V_{\text{load}}$  υπό ωμικό φορτίο δεδομένης τιμής αντίστασης). Από όλα τα παραπάνω προκύπτει το πρόβλημα της τιμής της ωμικής αντίστασης που πρέπει να χρησιμοποιηθεί για φορτίο, δεδομένου ότι η ευαισθησία της τιμής αυτής θα επηρεάσει την χρήση του συστήματος ως αισθητήρα ανίχνευσης μέγιστης ακτινοβολίας. Η τιμή θα εξαρτηθεί από τις εξής παραμέτρους :

1. Τα ηλεκτρικά χαρακτηριστικά του πάνελ που θα χρησιμοποιηθεί για τις μετρήσεις.
  2. Την ευαισθησία της  $V_{\text{load}}$  ως προς την επιλεγείσα τιμή της  $R_{\text{load}}$ .
  3. Την ανάγκη μεγιστοποίησης της ισχύος που θα παρέχει το φ/β πάνελ στο φορτίο του.
- Κυριότερο των τριών είναι αναμφισβήτητα το τρίτο. Για να το επιτύχουμε όμως πρέπει να γνωρίζουμε το επίπεδο της ακτινοβολίας, πράγμα αδύνατο αφού μεταβάλλεται τόσο εποχιακά, όσο και κατά την εξέλιξη της ημέρας. Επιλέγουμε ωστόσο προσεγγιστικά την τιμή εκείνη που μεγιστοποιεί την παραγόμενη ηλεκτρική ισχύ υπό 'καλές' συνθήκες, ήτοι ακτινοβολία σχεδόν πλήρους ήλιου (έχει αναλυθεί στην θεωρία)  $2100 \text{ mW cm}^{-2}$ . Η προσέγγιση αυτή δεν είναι κακή, αν λάβει κανείς υπόψη του ότι είναι

επιτεύξιμο με τον solar tracker να έχουμε λαμβανόμενη ένταση ακτινοβολίας περίπου ενός πλήρους ήλιου .

Έλεγχος της δεύτερης παραμέτρου : από το σχήμα 2.6 βλέπουμε ότι με αντίσταση φορτίου αυτήν που μας δίνει μέγιστη ισχύ υπό συνθήκες ενός ήλιου μας παρέχει ευαισθησία αρκετά κοντά στην μέγιστη ευαισθησία της Vload συναρτήσει της έντασης της προσπίπτουσας ακτινοβολίας, ενώ εξακολουθεί να έχει πολύ καλή ευαισθησία για τις περισσότερες σχετικά μεγάλες εντάσεις που φαίνονται στο διάγραμμα. Χάνει βέβαια αρκετά σε πολύ μικρές εντάσεις, που είναι ελάχιστος σημασίας για την παραγωγή ηλεκτρικής ενέργειας από φ/β . Εν κατακλείδι η τιμή της αντίστασης φορτίου που μεγιστοποιεί την ικανότητα του φ/β πάνελ να παράγει ηλεκτρική ενέργεια υπό συνθήκες ‘καλής’ ακτινοβολίας είναι μια εξαιρετική τιμή και από άποψη λειτουργίας του συστήματος ως ανιχνευτή της θέσης του ήλιου . Επομένως , η τιμή της αντίστασης του φορτίου θα υπολογιστεί από το maximum power point του διαγράμματος του φ/β πάνελ για ακτινοβολία ενός πλήρους ήλιου.

## **ΚΕΦΑΛΑΙΟ 3**

### **ARDUINO**

#### **3.1 Εισαγωγή**

Το Arduino είναι ένας single-board μικροελεκτης, δηλαδή μια απλή μητρική πλακέτα ανοικτού κώδικα, με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στην C++ ). Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider. Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν προ-συναρμολογημένες: το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino μόνοι τους.

#### **3.2 Ιστορικό**

Το 2005, ένα σχέδιο κίνησε προκειμένου να φτιαχτεί μία συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν πιο φθηνή από άλλα πρωτότυπα συστήματα διαθέσιμα εκείνη την περίοδο. Οι ιδρυτές Massimo Banzi

και David Cueartielles ονόμασαν το σχέδιο από τον Arduino της Inrea[1] και ξεκίνησαν να παράγουν πλακέτες σε ένα μικρό εργοστάσιο στην Ιβρέα, κωμόπολη της επαρχίας Τορίνο στην περιοχή Πεδεμόντιο της βορειοδυτικής Ιταλίας- την ίδια περιοχή στην οποία στεγαζόταν η εταιρία υπολογιστών Olivetti[2].

Το σχέδιο Arduino είναι μία διακλάδωση της πλατφόρμας Wiring για λογισμικό ανοικτού κώδικα και προγραμματίζεται χρησιμοποιώντας μια γλώσσα βασισμένη στο Wiring (σύνταξη και βιβλιοθήκες), παρόμοια με την C++ με απλοποιήσεις και αλλαγές, καθώς και ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE).

### **3.3 Εκδόσεις**

- Τον Σεπτέμβριο του 2006 ανακοινώθηκε το Arduino Mini
- Τον Οκτώβρη του 2008 ανακοινώθηκε το Arduino Duemilanove. Αρχικά βασίστηκε στο Atmel Atmega168, αλλά μετά στάλθηκε με το ATmega328.
- Τον Μάρτιο του 2009 ανακοινώθηκε το Arduino Mega. Είναι βασισμένο στο Atmel ATmega1280
- Από τον Μάιο του 2011 πάνω από 300,000 Arduino ήταν σε χρήση σε όλο τον κόσμο
- Τον Ιούλιο του 2012 ανακοινώθηκε το Arduino Leonardo. Είναι βασισμένο στο Atmel ATmega32u4
- Τον Οκτώβριο του 2012 ανακοινώθηκε το Arduino Due. Είναι βασισμένο στο Atmel SAM3X8E, που είχε πυρήνα ARM Cortex-M3
- Τον Νοέμβριο του 2012 ανακοινώθηκε το Arduino Micro. Είναι βασισμένο στο Atmel ATmega32u4
- Τον Μάιο του 2013 ανακοινώθηκε το Arduino Robot. Είναι βασισμένο στο Atmel ATmega32u4 και ήταν το πρώτο επίσημο Arduino με ρόδες
- Τον Μάιο του 2013 ανακοινώθηκε το Arduino Yun. Είναι Βασισμένο στο ATmega32u4 και στο Atheros AR9331 και ήταν το πρώτο προϊόν wifi που συνδύαζε το Arduino με το Linux.

### **3.4 Πλατφόρμα**

#### **3.4.1 Υλικό (Hardware)**

Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR (ATmega328 και ATmega168 στις νεότερες εκδόσεις, ATmega8 στις παλαιότερες) και συμπληρωματικά εξαρτήματα για την διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση του σε άλλα κυκλώματα. Όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και έναν κρυσταλλικό ταλαντωτή 16MHz (ή κεραμικό αντηχητή σε κάποιες παραλλαγές). Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα bootloader, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.

Σε εννοιολογικό επίπεδο, στην χρήση του Arduino software stack, όλα τα boards προγραμματίζονται με μία RS-232 σειριακή σύνδεση, αλλά ο τρόπος που επιτυγχάνεται αυτό διαφέρει σε κάθε hardware εκδοχή. Οι σειριακές πλάκες Arduino περιέχουν ένα απλό level shifter κύκλωμα για να μετατρέπει μεταξύ σήματος επιπέδου RS-232 και TTL. Τα τωρινά Arduino προγραμματίζονται μέσω USB, αυτό καθίσταται δυνατό μέσω της εφαρμογής προσαρμοστικών chip USB-to-Serial όπως το FTDI FT232. Κάποιες παραλλαγές, όπως το Arduino mini και το ανεπίσημο Boarduino, χρησιμοποιούν ένα

αφαιρούμενο USB-to-Serial καλώδιο ή board, Bluetooth ή άλλες μεθόδους. (Όταν χρησιμοποιείτε με παραδοσιακά εργαλεία microcontroller αντί για το Arduino IDE, πρότυπος προγραμματισμός AVR ISP χρησιμοποιείται)

Ο πίνακας Arduino εκθέτει τα περισσότερα microcontroller I/O pins για χρήση από άλλα κυκλώματα. Τα Diecimila, Duemilanove και το τρέχον Uno παρέχουν 14 ψηφιακά I/O pins, έξι από τα οποία μπορούν να παράγουν pulse-width διαμορφωμένα σήματα, και έξι αναλογικά δεδομένα. Αυτά τα pins βρίσκονται στην κορυφή του πίνακα μέσω female headers 0.1 ιντσών (2,2mm). Διάφορες εφαρμογές ασπίδων plug-in είναι εμπορικώς διαθέσιμα.

Το Arduino nano, και το Arduino-Compatible Bare Bones Board και Boarduino Board ενδέχεται να παρέχει male header pins στο κάτω μέρος του board προκειμένου να συνδέονται σε Breadboards.

Υπάρχουν πολλά boards συμβατά με και προερχόμενα από Arduino boards. Κάποια είναι λειτουργικά ισάξια με ένα Arduino και μπορεί να χρησιμοποιηθούν εναλλακτικά. Πολλοί είναι το βασικό Arduino με την προσθήκη καινοτόμων output drivers, συχνά για την χρήση σχολικής μόρφωσης για να απλοποιήσουν την κατασκευή buggies και μικρών robot. Άλλες είναι ηλεκτρικά ισάξια αλλά αλλάζουν τον παράγοντα μορφής, επιτρέποντας κάποιες φορές την συνεχόμενη χρήση των Shields ενώ κάποιες όχι. Κάποιες παραλλαγές είναι τελείως διαφορετικοί επεξεργαστές, με ποικίλα επίπεδα συμβατότητας.

### **3.4.2 Επίσημες πλακέτες**

Το πρωτότυπο υλισμικό του Arduino κατασκευάζεται από την Ιταλική εταιρία Smart Projects.[9] Κάποιες πλακέτες με την μάρκα του Arduino έχουν σχεδιαστεί από την Αμερικάνικη εταιρία SparkFun Electronics.[10] Δεκαέξι εκδοχές του Arduino Hardware έχουν χρησιμοποιηθεί εμπορικά μέχρι τώρα:

1. Το Serial Arduino, προγραμματισμένο με μία σειριακή DE-9 σύνδεση χρησιμοποιώντας τεχνολογία ATmega8

2. Το Arduino Extreme, με ένα USB interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega8
3. Το Arduino Mini, μία έκδοση μινιατούρας του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168
4. Το Arduino Nano, ένα ακόμα πιο μικρό, USB τροφοδοτούμενη έκδοχή του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168 (ATmega328 για την νεότερη έκδοση)
5. Το LilyPad Arduino, ένα μινιμαλιστικό σχέδιο για εφαρμογές ένδυσης και E-textiles χρησιμοποιώντας τεχνολογία surface-mounted AT-mega328
6. Το Arduino NG, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία ATmega8
7. Το Arduino NG plus, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία atmega168
8. Το Arduino Bluetooth, με Bluetooth interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega168
9. Το Arduino Diecimila, με ένα USB interface και χρησιμοποιεί τεχνολογία ATmega168 σε ένα DIP28 πακέτο
10. Το Arduino Duemilanove (“2009”), χρησιμοποιεί τεχνολογία ATmega168 (ATmega328 για την καινούργια έκδοση) και τροφοδοτείται μέσω ενέργειας USB/DC, αυτόματα εναλλασσόμενης
11. Το Arduino Mega, χρησιμοποιώντας τεχνολογία surface-mounted ATmega1280 για περαιτέρω I/O και μνήμη[11]
12. Το Arduino Uno, χρησιμοποιώντας την ίδια τεχνολογία ATmega328 όπως το τελευταίο μοντέλο Duemilanove, αλλά ενώ το Duemilanove χρησιμοποιεί ένα FTDI chipset για το USB, το Uno χρησιμοποιεί τεχνολογία ATmega8U2 προγραμματισμένο ως σειριακός μετατροπέας
13. Το Arduino Mega2560, χρησιμοποιεί τεχνολογία surface-mounted ATmega2560 φέρνοντας την ολική μνήμη στα 256kB. Επίσης ενσωματώνει την νέα τεχνολογία ATmega8U2 (ATmega16U2 σε αναθεώρηση τύπου 3) USB chipset.
14. Το Arduino Leonardo, με ένα ATmega32U4 chip που εξαλείφει την ανάγκη για συνδεσιμότητα μέσω USB και μπορεί να χρησιμοποιηθεί ως ψηφιακό πληκτρολόγιο ή ποντίκι. Κυκλοφόρησε στο Maker Faire Bay Area το 2012



15. Το Arduino Esplora, με εμφάνιση που παραπέμπει σε χειριστήριο κονσόλας βιντεοπαιχνιδιών με joystick και ενσωματωμένους αισθητήρες για ήχο, φως, θερμοκρασία και επιτάχυνση

16. Το Arduino Due είναι ένα μικροχειριστήριο board βασισμένο στην τεχνολογία Atmel SAM3X8E ARM Cortex-M3 CPU. Είναι το πρώτο board της Arduino βασισμένη σε επεξεργαστή 32-bit ARM microcontroller[12][6]

### **3.4.3 Shields**

Τα Arduino και τα Arduino συμβατά boards χρησιμοποιούν την τεχνολογία των shields, τυπωμένων boards επεκτάσεων κυκλωμάτων που συνδέονται στα κανονικά παρεχόμενα Arduino pin-headers. Τα shields μπορούν να παρέχουν έλεγχο στα motors, GPS, Ethernet, LCD εικόνας ή breadboarding (προτυποποίησης). Ένας αριθμός από ασπίδες μπορεί επίσης να γίνει και DIY

### **3.5 Λογισμικό**

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισαγάγει τον προγραμματισμό στους καλλιτέχνες και τους νέους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Δεν υπάρχει συνήθως καμία ανάγκη να επεξεργαστείτε αρχεία make ή να τρέξετε προγράμματα σε ένα περιβάλλον γραμμής εντολών. Ένα πρόγραμμα ή κώδικας που γράφτηκε για Arduino ονομάζεται σκίτσο (sketch).

Τα Arduino προγράμματα είναι γραμμένα σε C ή C++. Το Arduino IDE έρχεται με μια βιβλιοθήκη λογισμικού που ονομάζεται "Wiring" από το πρωτότυπο σχέδιο Wiring γεγονός που καθιστά πολλές κοινές λειτουργίες εισόδου/εξόδου πολύ πιο εύκολες. Οι χρήστες πρέπει μόνο να ορίσουν δύο λειτουργίες για να κάνουν ένα πρόγραμμα κυκλικής εκτέλεσης:

-setup():μία συνάρτηση που τρέχει μία φορά στην αρχή του προγράμματος η οποία αρχικοποιεί τις ρυθμίσεις

-loop():μία συνάρτηση η οποία καλείται συνέχεια μέχρι η πλακέτα να απενεργοποιηθεί

Ένα τυπικό πρώτο πρόγραμμα για έναν μικροελεγκτή αναβοσβήνει απλά ένα LED. Στο περιβάλλον του Arduino, ο χρήστης μπορεί να γράψει ένα πρόγραμμα σαν αυτό:

```
#define LED_PIN 13
```

```
void setup () {  
  pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output  
}  
  
void loop () {  
  digitalWrite (LED_PIN, HIGH); // turn on the LED  
  delay (1000); // wait one second (1000 milliseconds)  
  digitalWrite (LED_PIN, LOW); // turn off the LED  
  delay (1000); // wait one second  
}
```

Είναι ένα χαρακτηριστικό των περισσότερων πλακετών Arduino ότι έχουν ένα LED και μία αντίσταση φορτίου που συνδέονται μεταξύ του pin 13 και του εδάφους, ένα βολικό χαρακτηριστικό για πολλά απλά τεστ. Ο προηγούμενος κώδικας δεν θα αναγνωριστεί από ένα κανονικό μεταγλωττιστή C ++ ως έγκυρο πρόγραμμα, έτσι ώστε όταν ο χρήστης κάνει κλικ στο κουμπί "Upload to I / O board" στο IDE, ένα αντίγραφο του κώδικα θα γραφτεί σε ένα προσωρινό αρχείο με ένα παραπάνω include στην κορυφή και μία πολύ απλή συνάρτηση main() στο τέλος, για να φτιάξει ένα έγκυρο C++ πρόγραμμα.

Το IDE του Arduino χρησιμοποιεί το GNU toolchain και το AVR Libc για να μεταγλωττίζει προγράμματα και το avrdude για να φορτώνει προγράμματα στην πλακέτα.

Δεδομένου ότι η πλατφόρμα Arduino χρησιμοποιεί Atmel μικροελεγκτές, το περιβάλλον ανάπτυξης της Atmel, το AVR Studio ή το νεότερη έκδοση του Atmel Studio, μπορεί επίσης να χρησιμοποιηθεί για την ανάπτυξη λογισμικού για το Arduino.

### **3.6 Ανάπτυξη**

Η κύρια ομάδα ανάπτυξης του Arduino αποτελείται από τους: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, David Mellis και Nicholas Zambetti. Ο Massimo Banzi έδωσε συνέντευξη στις 21 Μαρτίου του 2009 στο επεισόδιο 61 του FLOSS Weekly στο κανάλι TWiT.tv, στο οποίο συζήτησε την ιστορία και τους στόχους του προγράμματος Arduino. Επίσης, έδωσε μια ομιλία στο TEDGlobal 2012 Conference, όπου περιέγραψε διάφορες χρήσεις των πλακετών Arduino σε όλο τον κόσμο.

Το Arduino είναι υλισμικό ανοιχτού λογισμικού: τα σχέδια αναφοράς του υλισμικού του Arduino διανέμονται υπό την Creative Commons Attribution Share-Alike 2.5 άδεια και είναι διαθέσιμα στην ιστοσελίδα του Arduino. Ο σχεδιασμός και η παραγωγή αρχείων για κάποιες εκδόσεις του υλισμικού Arduino είναι επίσης διαθέσιμοι. Ο πηγαίος κώδικας για το IDE είναι διαθέσιμος και διανέμεται υπό την GNU General Public License, έκδοση 2. Παρά το γεγονός ότι το υλισμικό και τα σχέδια του λογισμικού είναι διαθέσιμα ελεύθερα υπό άδειες πνευματικών δικαιωμάτων, οι προγραμματιστές έχουν ζητήσει η ονομασία "Arduino" είναι αποκλειστική για το επίσημο προϊόν και δεν πρέπει να χρησιμοποιείται για τις εργασίες χωρίς άδεια. Το επίσημο έγγραφο πολιτικής σχετικά με τη χρήση του ονόματος Arduino τονίζει ότι το πρόγραμμα είναι ανοιχτό στη συνεργασία με άλλους στο επίσημο προϊόν. Αρκετά προϊόντα συμβατά με Arduino που κυκλοφορούν στο εμπόριο έχουν αποφύγει το όνομα "Arduino" χρησιμοποιώντας την κατάληξη "-duino" με παραλλαγές στο όνομα.

## ΚΕΦΑΛΑΙΟ 4

### Η ΚΑΤΑΣΚΕΥΗ

Η εργασία μας ονομάζεται σχεδίαση και κατασκευή διάταξης παρακολούθησης ηλίου και έχει σκοπό να ακολουθάει την πορεία του ήλιου όλες τις εποχές.

Για να υλοποιήσουμε την κατασκευή αυτή χρειαστήκαμε:

1) Δύο βηματικούς κινητήρες με τα αντίστοιχα drivers οι οποίοι ο ένας στην βάση θα περιστρέφει όλη την κατασκευή, δηλαδή αυτός θα ακολουθάει την πορεία του ήλιου και ο άλλος θα περιστρέφει το πάνελ. Οι κινητήρες αυτοί και τα αντίστοιχα drivers είναι τα παρακάτω:

A) Κινητήρας 1 : 42BYGHM809 12V με driver DRV8825

B) Κινητήρας 2 : 28BYJ-48 5V με driver ULN2003a

2) Για τον πρώτο κινητήρα χρειαστήκαμε επιπλέον ένα τροφοδοτικό για να πάρουμε τα 12V.

3) Τοποθετήσαμε μία ηλεκτρονική πυξίδα HMC5883L την οποία την προγραμματίσαμε με το που μπαίνει σε λειτουργία το σύστημα σε όπια θέση και να βρίσκεται το πάνελ αυτό θα βρίσκει την ανατολή.

4) Επίσης χρησιμοποιήσαμε ένα real time clock DS1307 το οποίο κρατάει στην μνήμη του την ημερομηνία και την ώρα και ξέρει όλες τις μέρες του χρόνου πότε είναι η ανατολή και πότε η δύση του ήλιου.

5) Ένα arduino στο οποίο όλα συνδέονται πάνω και αυτό υλοποιεί όλο το πρόγραμμα.

6) Τέλος χρησιμοποιήσαμε ένα breadboard για να βάλουμε πάνω του όλα τα τσιπάκια και να τα συνδέσουμε με το arduino.

## ΤΟ ΠΡΟΓΡΑΜΜΑ

### Program.cs

```
using System;
using System.Threading;
using System.Threading.Tasks;
namespace ArduinoController
{
class Program
{
static void Main(string[] args)
```

```

{
PortController.Instance.Initialize();
var task = Tasks.RunPeriodically(async () =>
{
var calculator = new SunHelper();
if (DateTime.Now > calculator.CalculateSunrise() && DateTime.Now <
calculator.CalculateSunset())
PortController.Instance.WriteLine("DO");
await Task.CompletedTask;
}, new TimeSpan(0, 0, 0, 30), CancellationToken.None);
task.Wait();
var test = "";
}
}
}

```

### PortController.cs

```

using System;
using System.Collections.Generic;
using System.Configuration;
using System.IO.Ports;
namespace ArduinoController
{
class PortController
{
private SerialPort _serial;
private bool _isInitialized = false;
private readonly object _lockObj = new object();
private static string _buffer = string.Empty;
private static object _bufferLockObj = new object();
private static Stack<string> _messages = new Stack<string>();
private static PortController _instance;
public bool _isConnected = false;
public static PortController Instance
{
get { return _instance ?? (_instance = new PortController()); }
}
private PortController()
{
this._serial = new
SerialPort(ConfigurationManager.AppSettings.Get("Port"),
int.Parse(ConfigurationManager.AppSettings.Get("BaudRate")));
this._serial.DataReceived += DataReceivedHandler;
}
protected virtual void DoInitialize()
{
while (true)
{
this._serial.WriteLine("{{{start}}}");
if(_isConnected)
break;
}
}
private static void DataReceivedHandler(object sender,
SerialDataReceivedEventArgs e)
{
var sp = (SerialPort) sender;
var size = sp.BytesToRead;
if (size > 0)
{
var charArray = new byte[size];

```

```

sp.Read(charArray, 0, size);

lock (_bufferLockObj)
{
    _buffer += System.Text.Encoding.Default.GetString(charArray);
}
}
ExtractMessages();
DispatchMessaes();
}
private static void ExtractMessages()
{
    while (true)
    {
        var start = _buffer.IndexOf("{{{");
        if (start >= 0)
        {
            var end = _buffer.IndexOf("}}}", start);
            if (end > start)
            {
                _messages.Push(_buffer.Substring(start + 3, end - start - 3));
                lock (_bufferLockObj)
                {
                    _buffer = _buffer.Substring(end);
                }
            }
            else
            {
                break;
            }
        }
        else
        {
            break;
        }
    }
}
private static void DispatchMessaes()
{
    while (_messages.Count > 0)
    {
        var message = _messages.Pop();
        switch (message)
        {
            case "started":
                var startedEvent =
                    (BaseEvent)Activator.CreateInstance(typeof(StartedEvent));
                startedEvent.Run();
                break;
            case "OK":
                var moveEvent = (BaseEvent) Activator.CreateInstance(typeof
                    (MoveEvent));

                moveEvent.Run();
                break;
            default:
                Console.WriteLine(message);
                break;
        }
    }
}
public void Initialize()
{

```

```

if (!_isInitialized)
{
lock (_lockObj)
{
if (!_isInitialized)
{
this._serial.Open();
this.DoInitialize();
_isInitialized = true;
}
}
}
}
public void WriteLine(string message)
{
this._serial.WriteLine($"{{" + message + "}}");
}
}
}

```

#### Tasks.cs

```

using System;
using System.Threading;
using System.Threading.Tasks;
namespace ArduinoController
{
public static class Tasks
{
public static async Task RunPeriodically(Action action, TimeSpan interval,
Cancellation token)
{
while (true)
{
action();
await Task.Delay(interval, token);
}
}
}
}

```

#### MoveEvent.cs

```

namespace ArduinoController
{
sealed class StartedEvent : BaseEvent
{
protected override void DoRun()
{
base.DoRun();
this._controller._isConnected = true;
}
}
class MoveEvent : BaseEvent
{
protected override void DoRun()
{
base.DoRun();
var calculator = new SunHelper();
var sunPosition = calculator.CalculateSunPosition();
this._controller.WriteLine(sunPosition.Azimuth.ToString());
}
}
abstract class BaseEvent
{

```

```

protected readonly PortController _controller;
public BaseEvent()
{
    this._controller = PortController.Instance;
}
public void Run()
{
    this.DoRun();
}
protected virtual void DoRun()
{
}
}
}

```

```

SunCalculator.cs
using System;
using System.Configuration;
using System.Globalization;
enum CalcMode
{
    Sunset = 0, Sunrise = 1
}
public class SunHelper
{
    public double Latitude { get; }
    public double Longitude { get; }
    private readonly SunCalculator _calculator;
    public SunHelper()
    {
        this.Latitude = double.Parse(ConfigurationManager.AppSettings["Latitude"]);
        this.Longitude = double.Parse(ConfigurationManager.AppSettings["Longitude"]);
        this._calculator = new SunCalculator();
    }
    public SunPosition CalculateSunPosition()
    {
        return this._calculator.CalculateSunPosition(DateTime.Now, this.Latitude,
            this.Longitude, TimeZone.CurrentTimeZone);
    }
    public DateTime CalculateSunrise()
    {
        return this._calculator.CalculateSunrise(DateTime.Now, this.Latitude,
            this.Longitude,
            TimeZone.CurrentTimeZone);
    }
    public DateTime CalculateSunset()
    {
        return this._calculator.CalculateSunset(DateTime.Now, this.Latitude,
            this.Longitude,
            TimeZone.CurrentTimeZone);
    }
}
class SunCalculator
{
    private const double Deg2Rad = Math.PI / 180.0;
    private const double Rad2Deg = 180.0 / Math.PI;
    public SunPosition CalculateSunPosition(DateTime dateTime, double latitude,
        double
        longitude, TimeZone localZone = null)

```



```

{
// Convert to UTC
dateTime = dateTime.ToUniversalTime();
// Number of days from J2000.0.
double julianDate = 367 * dateTime.Year -
(int)((7.0 / 4.0) * (dateTime.Year + (int)((dateTime.Month + 9.0) / 12.0))) +
(int)((275.0 * dateTime.Month) / 9.0) + dateTime.Day - 730531.5;

double julianCenturies = julianDate / 36525.0;
// Sidereal Time
double siderealTimeHours = 6.6974 + 2400.0513 * julianCenturies;
double siderealTimeUT = siderealTimeHours + (366.2422 / 365.2422) *
(double)dateTime.TimeOfDay.TotalHours;
double siderealTime = siderealTimeUT * 15 + longitude;
// Refine to number of days (fractional) to specific time.
julianDate += (double)dateTime.TimeOfDay.TotalHours / 24.0;
julianCenturies = julianDate / 36525.0;
// Solar Coordinates
double meanLongitude = CorrectAngle(Deg2Rad * (280.466 + 36000.77 *
julianCenturies));
double meanAnomaly = CorrectAngle(Deg2Rad * (357.529 + 35999.05 *
julianCenturies));
double equationOfCenter = Deg2Rad * ((1.915 - 0.005 * julianCenturies) *
Math.Sin(meanAnomaly) + 0.02 * Math.Sin(2 * meanAnomaly));
double ellipticalLongitude = CorrectAngle(meanLongitude + equationOfCenter);
double obliquity = (23.439 - 0.013 * julianCenturies) * Deg2Rad;
// Right Ascension
double rightAscension = Math.Atan2(Math.Cos(obliquity) *
Math.Sin(ellipticalLongitude),
Math.Cos(ellipticalLongitude));
double declination = Math.Asin(Math.Sin(rightAscension) * Math.Sin(obliquity));
// Horizontal Coordinates
double hourAngle = CorrectAngle(siderealTime * Deg2Rad) - rightAscension;
if (hourAngle > Math.PI)
hourAngle -= 2 * Math.PI;
double altitude = Math.Asin(Math.Sin(latitude * Deg2Rad) * Math.Sin(declination)
+
Math.Cos(latitude * Deg2Rad) * Math.Cos(declination) * Math.Cos(hourAngle));
// Nominator and denominator for calculating Azimuth
// angle. Needed to test which quadrant the angle is in.
double aziNom = -Math.Sin(hourAngle);
double aziDenom = Math.Tan(declination) * Math.Cos(latitude * Deg2Rad) -
Math.Sin(latitude * Deg2Rad) * Math.Cos(hourAngle);
double azimuth = Math.Atan(aziNom / aziDenom);
if (aziDenom < 0) // In 2nd or 3rd quadrant
azimuth += Math.PI;
else if (aziNom < 0) // In 4th quadrant
azimuth += 2 * Math.PI;
return new SunPosition()
{
Altitude = altitude * Rad2Deg,
Azimuth = azimuth * Rad2Deg
};
}
public DateTime CalculateSunrise(DateTime datetime, double latitude, double
longitude,
TimeZone localZone = null)
{
return CalculateSunriseSunset(datetime, latitude, longitude, CalcMode.Sunrise,
localZone);
}
}

```

```

public DateTime CalculateSunset(DateTime datetime, double latitude, double
longitude,
TimeZone localZone = null)
{
return CalculateSunriseSunset(datetime, latitude, longitude, CalcMode.Sunset,
localZone);
}
// calculate the sunset or rise, enter date without time.
// mode determines whether to calc sunrise or set.
private DateTime CalculateSunriseSunset(DateTime datetime, double latitude,
double
longitude, CalcMode mode, TimeZone localZone = null)
{
if (localZone == null)
localZone = TimeZone.CurrentTimeZone;
var date = datetime.Date.AddHours(12); // add 12 hours to the date to determine
the
middle of the day
// equation of time (in minutes)
var x = date.DayOfYear * 2 * Math.PI / 365; // fractional year in radians
var eqtime = 229.18 * (0.000075 + 0.001868 * Math.Cos(x) - 0.032077 * Math.Sin(x)
-
0.014615 * Math.Cos(2 * x) - 0.040849 * Math.Sin(2 * x));
// declination (in degrees)
var declin = 0.006918 - 0.399912 * Math.Cos(x) + 0.070257 * Math.Sin(x) -
0.006758 *
Math.Cos(2 * x);
declin = declin + 0.000907 * Math.Sin(2 * x) - 0.002697 * Math.Cos(3 * x) +
0.00148 *
Math.Sin(3 * x);
declin = declin * 180 / Math.PI;
// solar azimuth angle for sunrise and sunset corrected for atmospheric
refraction (in
degrees),
x = Math.PI / 180;
var hars = Math.Cos(x * 90.833) / (Math.Cos(x * latitude) * Math.Cos(x *
declin));
hars = hars - Math.Tan(x * latitude) * Math.Tan(x * declin);
hars = Math.Acos(hars) / x;
// Create a DaylightTime object for the specified year.
DaylightTime daylight = localZone.GetDaylightChanges(date.Year);
double offset = localZone.IsDaylightSavingTime(date) ?
(double)daylight.Delta.Hours :
0;
if (mode == CalcMode.Sunset)
x = 720 + 4 * (longitude + hars) - eqtime;
else
x = 720 + 4 * (longitude - hars) - eqtime;
x = (x / 60) + offset;
//return x;
return datetime.Date.Add(TimeSpan.FromHours(x));
}
private static double CorrectAngle(double angleInRadians)
{
if (angleInRadians < 0)
return 2 * Math.PI - (Math.Abs(angleInRadians) % (2 * Math.PI));
else if (angleInRadians > 2 * Math.PI)
return angleInRadians % (2 * Math.PI);
else
return angleInRadians;
}
}

```

```
public class SunPosition
{
public double Altitude { get; set; }
public double Azimuth { get; set; }
}
```

**App.config**

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
<appSettings>
<add key="Timezone" value="GTB Standard Time"/>
<add key="Latitude" value="37.948193"/>
<add key="Longitude" value="23.697183"/>
<add key="Port" value="COM4"/>
<add key="BaudRate" value="9600"/>
</appSettings>
<startup>
<supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.1" />
</startup>
</configuration>
```

## Κώδικας αρντουινο

```
#include <AccelStepper.h>
#include <TimeLib.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>
//wire.h addresses
#define address 0x1E
//base motor
#define stepPin 9
#define dirPin 8
#define enablePin 10
// upper motor. Motor pin definitions
#define HALFSTEP 8
#define motorPin1 4 // IN1 on the ULN2003 driver 1
#define motorPin2 5 // IN1 on the ULN2003 driver 1
#define motorPin3 6 // IN3 on the ULN2003 driver 1
#define motorPin4 7 // IN3 on the ULN2003 driver 1
int lookup[8] = {B01000, B01100, B00100, B00110, B00010, B00011, B00001,
B01001};
//GENERIC
```

```
int motorSpeed = 1500; //variable to set stepper speed
//compass
Adafruit_HMC5883_Unified mag = Adafruit_HMC5883_Unified(12345);
void initCompass()
{
  Serial.println("Initializing Compass!");
  if (!mag.begin())
  {
    /* There was a problem detecting the HMC5883 ... check your connections */
    Serial.println("Ooops, no HMC5883 detected ... Check your wiring!");
    while (1);
  }
}
float GetCompassHeading()
{
  /* Get a new sensor event */
  sensors_event_t event;
  mag.getEvent(&event);
  /* Display the results (magnetic vector values are in micro-Tesla (uT)) */
  //          Serial.println("X:          "          +          event.magnetic.x);
```

```

//Serial.println("Y: " + event.magnetic.y);
//Serial.println("Z: " + event.magnetic.z);
// Hold the module so that Z is pointing 'up' and you can measure the heading with
x&y. Calculate
heading when the magnetometer is level, then correct for signs of axis.
float heading = atan2(event.magnetic.y, event.magnetic.x);
// Once you have your heading, you must then add your 'Declination Angle', which is
the 'Error' of the
magnetic field in your location. Find yours here: http://www.magnetic-
declination.com/. Mine is: -13*
2' W, which is ~13 Degrees, or (which we need) 0.22 radians
float declinationAngle = 0.27;
heading += declinationAngle;
// Correct for when signs are reversed.
if (heading < 0)
heading += 2 * PI;
// Check for wrap due to addition of declination.
if (heading > 2 * PI)
heading -= 2 * PI;
// Convert radians to degrees for readability.
float headingDegrees = heading * 180 / M_PI;
//Serial.print("Heading (degrees): "); Serial.println(headingDegrees);
return headingDegrees;
}
void initX() {
Serial.println("Initializing Xmotor!");
// We set the enable pin to be an output
pinMode(enablePin, OUTPUT);
// then we set it HIGH so that the board is disabled until we get into a known state.
digitalWrite(enablePin, HIGH);
pinMode(stepPin, OUTPUT);
pinMode(dirPin, OUTPUT);
}
void moveX(float deg, float speed = 1) {
//rotate a specific number of degrees (negative for reverse movement) speed is any
number from .01 ->
1 with 1 being fastest - Slower is stronger
Serial.println(deg > 0 ? "right" : "left");
int dir = (deg > 0) ? LOW : HIGH;
digitalWrite(dirPin, dir);
}

```

```

int steps = abs(deg) * (1 / 0.225);
float usDelay = (1 / speed) * 250;
digitalWrite(enablePin, LOW);
for (int i = 0; i < steps; i++) {
//Serial.print(GetCompassHeading()); Serial.print("\n");
digitalWrite(stepPin, HIGH);
delayMicroseconds(usDelay);
digitalWrite(stepPin, LOW);
delayMicroseconds(usDelay);
}
delayMicroseconds(usDelay);
digitalWrite(enablePin, HIGH);
}
void breakX() {
digitalWrite(enablePin, LOW);
delay(400);
digitalWrite(enablePin, HIGH);
}
void InitY() {
Serial.println("Initializing Ymotor!");
pinMode(motorPin1, OUTPUT);
pinMode(motorPin2, OUTPUT);
pinMode(motorPin3, OUTPUT);
pinMode(motorPin4, OUTPUT);
}
void moveY(int direction, int step = 400)
{
if (direction == 0)
{
for (int i = 0; i <= step; i++)
{
setOutput(i % 8);
delayMicroseconds(motorSpeed);
}
} else {
for (int i = step; i >= 0; i--)
{
setOutput(i % 8);
delayMicroseconds(motorSpeed);
}
}
}
}

```

```

void setOutput(int out)
{
digitalWrite(motorPin1, bitRead(lookup[out], 0));
digitalWrite(motorPin2, bitRead(lookup[out], 1));
digitalWrite(motorPin3, bitRead(lookup[out], 2));
digitalWrite(motorPin4, bitRead(lookup[out], 3));
}
void initStartPosition(float degrees = 90.0, float error = 3.0)
{
Serial.println("Setting Start Position");
float eastDegrees = 90.0;
float currentDirection;
while (true) {
currentDirection = GetCompassHeading();
if(180 + degrees > currentDirection && currentDirection > degrees)
moveX(-2);
else
moveX(2);
if (currentDirection > degrees - error && currentDirection < degrees + error) {
breakX();
break;
}
}
delay(3000);
Serial.println("Panel position sucessfully set.");
}
String ReadLine()
{
String inData;
int messageStart;
int messageEnd;
while(true)
{
delay(10); //delay to allow buffer to fill
if (Serial.available() > 0) {
char c = Serial.read(); //gets one byte from serial buffer
inData += c; //makes the string readString
messageStart = inData.indexOf("{}");
}
}
}

```

```

messageEnd = inData.indexOf("}}}");
if(messageStart >=0 && messageStart + 3 < messageEnd)
{
return inData.substring(messageStart + 3, messageEnd);
}
}
}
}
}
void WriteLine(String Message)
{
String start = "{{{";
String stop = "}}}";
Serial.println(start + Message + stop);
}

void setup()
{
Serial.begin(9600);
Serial.println("Starting stepper exerciser.");
while(true)
{
if(ReadLine() == "start")
{
break;
}
}
}
initX();

InitY();
initCompass();
//initStartPosition();
WriteLine("started");
Serial.println("Initialized Connection");
}

void loop()
{
delay(1000);
String message = ReadLine();
if(message == "DO")
{
WriteLine("OK");
Serial.println("{{{Arduino Woke Up}}}");
initStartPosition(ReadLine().toFloat());
}
}
}

```



## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

- [1] N.Mohan, T.M. Undeland, “Power Electronics, Converters, Applications and Design”, John Wiley & Sons, 1995.
- [2] I. Khan, J. Tapson, I. de Vriew, “Frequency control of a current-fed inverter for induction heating”, IEEE Proceedings on Industrial Electronics, Vol. 1, No.1, 2000, pp.343-346.
- [3] Fairchild semiconductors, “Induction Heating System Topology Review”, Applications notes, July 2000.
- [4]

**ΑΘΗΝΑ**  
**31 ΜΑΙΟΥ 2017**