



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ  
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

## Ηλιακή Γεωμετρία: Σύστημα παρακολούθησης της αξιμούθιας ηλιακής τροχιάς

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γεώργιος Α. Ρεϊτζόπουλος

**Επιβλέπων :** Κωνσταντίνος Θ. Δέρβος  
Καθ. ΕΜΠ, Σχολή Ηλεκτρολόγων Μηχ/κων & Μηχ/κων Υπολογιστών

Αθήνα, Ιανουάριος 2014





ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ  
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

## Ηλιακή Γεωμετρία: Σύστημα παρακολούθησης της αζιμούθιας ηλιακής τροχιάς

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Γεώργιος Α. Ρεϊτζόπουλος

**Επιβλέπων :** Κωνσταντίνος Θ. Δέρβος

Καθ. ΕΜΠ, Σχολή Ηλεκτρολόγων Μηχ/κων & Μηχ/κων Υπολογιστών

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 31<sup>η</sup> Ιανουαρίου 2014.

.....  
Κ. Θ. Δέρβος  
Καθηγητής ΕΜΠ

.....  
Π. Βασιλείου  
Καθηγητής ΕΜΠ

.....  
Κ. Καραγιαννόπουλος  
Καθηγητής ΕΜΠ

Αθήνα, Ιανουάριος 2014

.....  
Γεώργιος Α. Ρεϊτζόπουλος

Διπλωματούχος Ηλεκτρολόγος Μηχανικός & Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Γεώργιος Α. Ρεϊτζόπουλος, 2014

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 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 τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

## ΠΕΡΙΛΗΨΗ

Σκοπός της παρούσας διπλωματικής είναι η κατασκευή προγράμματος για τον υπολογισμό της αζιμούθιας γωνίας με στόχο να χρησιμοποιηθεί για τη δημιουργία ιχνηλάτη αζιμούθιας γωνίας. Η υλοποίηση του ιχνηλάτη αζιμούθιας γωνίας, για την παρακολούθηση της αζιμουθιακής τροχιάς του Ήλιου, έγινε στα πλαίσια της διπλωματικής εργασίας με τη συσκευή Zaber T-RS 60. Για τον προγραμματισμό του ιχνηλάτη είναι αναγκαία η άριστη γνώση των αστρονομικών αλγορίθμων αλλά και η θεωρία της ουράνιας σφαίρας, δηλαδή του πλανητικού συστήματος το οποίο παραδέχεται ως κέντρο του την Γη. Ο προγραμματισμός αυτός έγινε με τη βοήθεια της γλώσσας C#.

## ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ

Ιχνηλάτης, Ηλιακός ιχνηλάτης, Αζιμουθιακός Ιχνηλάτης, Αζιμούθια γωνία, Ήλιος, Φωτοβολταϊκή εγκατάσταση, Συγκεντρωτικό σύστημα



## ABSTRACT

The purpose of this diploma thesis is the construction of a program for the calculation of the azimuthal angle in order to be used to generate an azimuthal angle tracker. The implementation of the azimuthal angle tracker for monitoring the azimuthal orbit of the Sun , was realized at this diploma thesis, with the use of the device Zaber T-RS 60. To program the tracker, is required the accurate knowledge of the astronomical algorithms and also the theory of the celestial sphere, i.e. the planetary system whose center is considered to be planet Earth. The programming was done with the use of language C #.

## KEYWORDS

Tracker, Solar tracker, azimuth trackers, azimuth angle, Sun, Photovoltaic installation, Cumulative system





Στο σημείο αυτό νιώθω την υποχρέωση να ευχαριστήσω όλους αυτούς που συνέβαλαν στην εκπόνηση της παρούσας διπλωματικής εργασίας και πιο συγκεκριμένα:

Τον καθηγητή κ. Δέρβο Κωνσταντίνο, καθηγητή του τομέα Συστημάτων Μετάδοσης Πληροφορίας & Τεχνολογίας Υλικών, για την εμπιστοσύνη του στην ανάθεση της παρούσας διπλωματικής εργασίας και το ενδιαφέρον του καθ' όλη τη διάρκεια υλοποίησης της.

Τον κ. Αργυρόπουλο Θεόδωρο, Δρ. Ναυπηγό Μηχανικό, ΙΔΑΧ στο Εργαστήριο Ηλεκτροτεχνικών Υλικών, για τη βοήθεια και τις τεχνικές γνώσεις του.

Τέλος τους φίλους Μουλό Χρήστο, Αγγέλου Βαγγέλη, Σάρρα Στέφανο για την παντός είδους βοήθεια που μου παρείχαν & την Σπυροπούλου Ιωάννα για την στήριξη της.

Αθήνα, Ιανουάριος 2014

Γεώργιος Α. Ρεϊτζόπουλος



## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	11
ΚΕΦΑΛΑΙΟ Ι ΕΙΣΑΓΩΓΗ	13
ΚΕΦΑΛΑΙΟ ΙΙ Η ΗΛΙΑΚΗ ΓΕΩΜΕΤΡΙΑ	17
2.1 ΗΛΙΑΚΗ ΕΝΕΡΓΕΙΑ – ΦΩΣ	19
2.2 ΓΕΩΜΕΤΡΙΑ ΗΛΙΟΥ – ΓΗΣ	19
2.2.1 ΓΕΩΜΕΤΡΙΑ ΗΛΙΟΥ – ΓΗΣ	19
2.2.2 Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ ΩΣ ΠΡΟΣ ΣΗΜΕΙΟ ΠΑΝΩ ΣΤΗ ΓΗ	21
2.2.2.1 ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΟΠΟΘΕΣΙΑΣ ΠΑΝΩ ΣΤΗ ΓΗ	21
2.2.2.2 ΤΟ ΟΡΙΖΟΝΤΙΟ ΣΥΣΤΗΜΑ	22
2.2.3 ΗΛΙΑΚΗ ΩΡΑ & Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ	24
2.2.3.1 Η ΗΛΙΑΚΗ ΩΡΑ	24
2.2.3.2 Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ	27
ΚΕΦΑΛΑΙΟ ΙΙΙ ΑΝΑΝΕΩΣΙΜΕΣ ΠΗΓΕΣ ΕΝΕΡΓΕΙΑΣ	31
3.0 ΕΙΣΑΓΩΓΗ	33
3.1 ΗΛΙΑΚΗ ΕΝΕΡΓΕΙΑ	35
3.1.1 ΦΩΤΟΒΟΛΤΑΪΚΑ ΣΤΟΙΧΕΙΑ	36
3.1.2 ΣΥΓΚΕΝΤΡΩΤΙΚΑ ΣΥΣΤΗΜΑΤΑ	41
ΚΕΦΑΛΑΙΟ ΙV ΦΩΤΟΒΟΛΤΑΪΚΟΙ ΙΧΝΗΛΑΤΕΣ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΤΟΥ ΗΛΙΟΥ	45
4.1 ΙΧΝΗΛΑΤΕΣ (trackers)	47
4.2 ΦΩΤΟΒΟΛΤΑΪΚΟΙ ΙΧΝΗΛΑΤΕΣ (solar trackers)	47
4.3 ΦΩΤΟΒΟΛΤΑΪΚΟΙ ΙΧΝΗΛΑΤΕΣ ΑΖΥΜΟΥΘΙΟΥ	55
ΚΕΦΑΛΑΙΟ V ΠΡΟΣΟΜΟΙΩΣΗ ΤΟΥ ΗΛΙΑΚΟΥ ΙΧΝΗΛΑΤΗ	57

<b>5.1 Η ΣΥΣΚΕΥΗ ΠΡΟΣΟΜΟΙΩΣΗΣ ZABER T-RS 60</b>	<b>59</b>
<b>5.2 ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΟΥ ΗΛΙΑΚΟΥ ΙΧΝΗΛΑΤΗ</b>	<b>60</b>
<b>5.2.1 Η ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ</b>	<b>60</b>
<b>5.2.2 Η ΥΛΟΠΟΙΗΣΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΕΛΕΓΧΟΥ ΤΟΥ ΗΛΙΑΚΟΥ ΙΧΝΗΛΑΤΗ</b>	<b>61</b>
<b>5.2.2.1 ΥΠΟΛΟΓΙΣΜΟΙ ΤΩΝ ΔΕΔΟΜΕΝΩΝ</b>	<b>61</b>
<b>5.2.2.2 ΑΝΑΓΝΩΡΙΣΗ &amp; ΕΠΙΚΟΙΝΩΝΙΑ ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ ΜΕ ΤΗΝ ΣΥΣΚΕΥΗ ΠΡΟΣΟΜΟΙΩΣΗΣ ΤΟΥ ΗΛΙΑΚΟΥ ΙΧΝΗΛΑΤΗ ΜΕΣΩ ΘΥΡΑΣ USB</b>	<b>71</b>
<b>ΠΑΡΑΡΤΗΜΑ 1</b>	<b>73</b>
<b>ΠΙΝΑΚΑΣ ΧΑΡΑΚΤΗΡΙΣΤΩΝ ZABER T_RS 60 A</b>	
<b>ΠΑΡΑΡΤΗΜΑ 2</b>	<b>75</b>
<b>ΚΩΔΙΚΑΣ ΠΡΟΓΡΑΜΜΑΤΟΣ</b>	
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b>	<b>95</b>

Κεφάλαιο **I**

**Εισαγωγή**



Η ηλιακή ενέργεια, με τη βοήθεια της σύγχρονης τεχνολογίας, μπορεί να αξιοποιηθεί για την παραγωγή θερμότητας, την παραγωγή ηλεκτρικής ενέργειας αλλά και τον συνδυασμό τους.

Η χρήση του ηλιακού θερμοσίφωνα για την παραγωγή θερμού νερού για την κάλυψη των ενεργειακών αναγκών ενός κτηρίου ή ενός συγκροτήματος κτηρίων, τόσο για οικιακή όσο και για επαγγελματική χρήση, κερδίζει συνεχώς έδαφος τα τελευταία χρόνια καθώς αποτελεί μια οικονομική λύση, φιλική προς το περιβάλλον. Η χρήση ιχνηλάτη για την παρακολούθηση του ήλιου βοηθάει ώστε να αυξηθεί σημαντικά η απόδοση του, ιδιαίτερα στις σημαντικές απαιτήσεις μιας επιχείρησης του κλάδου (πχ τουρισμός, κατασκηνώσεις, περιοχές με μειωμένη ηλιοφάνεια κτλ.).

Η αξιοποίηση του ήλιου για την παραγωγή ηλεκτρικής ενέργειας, εντάσσει τα φωτοβολταϊκά συστήματα στις ανανεώσιμες πηγές ενέργειας, δημιουργώντας φθηνή και μη ρυπογόνο ενέργεια. Ακόμη η παραγωγή ενέργειας μέσω φωτοβολταϊκών συστημάτων ιχνηλάτησης αυξάνει σημαντικά την παραγωγή ηλεκτρικής ενέργειας όπως επίσης ακολουθεί, σε φυσιολογικές συνθήκες, τόσο την ημερήσια όσο και την ετήσια ζήτηση ενέργειας με αποτέλεσμα να αποτελεί ένα χρήσιμο εφεδρικό σύστημα για την κάλυψη των ενεργειακών αναγκών.

Εξίσου χρήσιμη είναι η ηλιακή ενέργεια και στα συγκεντρωτικά συστήματα για την παραγωγή ηλεκτρικής ενέργειας, όπου δεδομένη είναι η χρήση ιχνηλατών.





## Κεφάλαιο II

# Ηλιακή γεωμετρία

Το εν λόγω κεφάλαιο αναφέρεται στη φύση της ακτινοβολίας που εκπέμπεται από τον ήλιο και που προσπίπτει στην επιφάνεια της γης. Συγκεκριμένα, περιλαμβάνονται γεωμετρικές αναλύσεις, όπως η κατεύθυνση λήψης της άμεσης ηλιακής ακτινοβολίας και η γωνιά πρόσπτωσης της σε διαφορετικές επιφάνειες.



## 2.1 ΗΛΙΑΚΗ ΕΝΕΡΓΕΙΑ – ΦΩΣ

Στο παρελθόν, το φως θεωρείτο ότι ήταν ένα ηλεκτρομαγνητικό κύμα. Αυτή η έξυπνη θεωρία άφηνε αρκετά φαινόμενα ανεξήγητα. Την λύση έδωσε ο Einstein καταφέροντας να εξηγήσει την επίδραση της συχνότητας και της ενέργειας του φωτός στο φωτοηλεκτρικό φαινόμενο εισάγοντας τα κβάντα φωτός. Η προσέγγιση αυτή οδήγησε στην ανακάλυψη του στοιχειώδους σωματιδίου φωτός, του φωτονίου. Αυτό χαρακτηρίζει τη δυαδική φύση του φωτός, που συνιστά ένα κύμα ύλης.

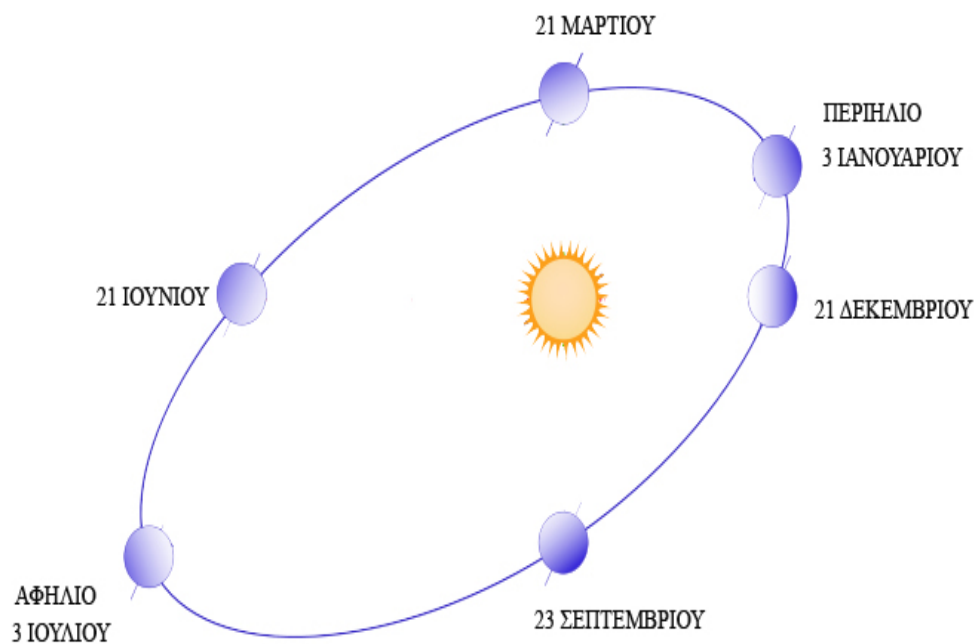
Κατά την μελέτη της ηλιακής ενέργειας ενδιαφερόμαστε κυρίως για τον ανά μονάδα επιφάνειας ρυθμό ροής ηλιακής ενέργειας του φωτός που προσπίπτει κάθετα σε έναν ηλιακό συλλέκτη. Αυτό ονομάζεται πυκνότητα ισχύος, ή ένταση φωτισμού, ή απλά ηλιακή ακτινοβολία και μετράται σε  $W/m^2$ .

## 2.2 ΓΕΩΜΕΤΡΙΑ ΗΛΙΟΥ – ΓΗΣ

### 2.2.1 ΓΕΩΜΕΤΡΙΑ ΗΛΙΟΥ – ΓΗΣ

Τα μαθηματικά που καθορίζουν την κίνηση της Γής γύρω από τον Ήλιο είναι ιδιαίτερα πολύπλοκα, αν λάβουμε υπόψη μας ότι ο Ήλιος αποτελεί σταθερό σημείο ως προς την περιφερόμενη Γη. Αναλύοντας την κίνηση σε συνιστώσες μας ενδιαφέρουν δύο κύριες που σχετίζονται με την παρακολούθηση του Ήλιου από οποιοδήποτε σημείο στην επιφάνεια της Γης. Οι κινήσεις αυτές είναι η ελλειπτική τροχιά και η περιστροφή της γης, γύρω από τον εαυτό της.

Η Γη κινείται σε ελλειπτική τροχιά γύρω από τον Ήλιο, όπως φαίνεται και στο σχήμα 2.1. Η τροχιά αυτή αναπτύσσεται στο εκλειπτικό επίπεδο (ecliptic plane). Η ολοκλήρωση της κίνησης της γης, στο εκλειπτικό επίπεδο, απαιτεί έναν ημερολογιακό έτος (365 ημέρες). Η απόσταση που μεσολαβεί μεταξύ του κέντρου του Ήλιου και του κέντρου της Γης μεταβάλλεται κατά τη μεταφορά της Γης γύρω από τον Ήλιο, καθ' όλη τη διάρκεια του έτους. Η κοντινότερη απόσταση της τροχιάς της Γης γύρω από τον Ήλιο σηματοδοτεί το περιήλιο, ενώ η μακρινότερη απόσταση σηματοδοτεί το αφήλιο της τροχιάς.



**Σχήμα 2.1** Η εκλειπτική τροχιά της Γης γύρω από τον Ήλιο

Ακόμη η γη περιστρέφεται γύρω από τον άξονα της, ο οποίος ονομάζεται πολικός άξονας. Η περιστροφή γύρω από τον πολικό άξονα απαιτεί 24 ώρες και η γωνία που σχηματίζεται ανάμεσα στο εκλειπτικό πεδίο και στον πολικό άξονα διατηρείται σταθερή και είναι ίση με  $23.45^\circ$ . Η απόσταση που μεσολαβεί μεταξύ του κέντρου του Ήλιου και του κέντρου της Γης μεταβάλλεται κατά την περιφορά της Γης γύρω από τον Ήλιο. Η κοντινότερη απόσταση της τροχιάς της Γης γύρω από τον Ήλιο σηματοδοτεί το περιήλιο, ενώ η μακρινότερη απόσταση σηματοδοτεί το αφήλιο της τροχιάς. Κατά την περιφορά της Γης γύρω από τον Ήλιο διατηρείται σταθερός ο προσανατολισμός του πολικού άξονα ως προς το εκλειπτικό επίπεδο.

Για τους περισσότερους πρακτικούς υπολογισμούς που συνδέονται με την ηλιακή ενέργεια την οποία δέχεται ένας ηλιακός συλλέκτης, είναι χρήσιμο να υποθέτουμε ότι η Γη είναι σταθερή και ότι ο Ήλιος κινείται γύρω της, διαγράφοντας τροχιά πάνω στο εκλειπτικό πεδίο. Στην περίπτωση αυτή, ο πολικός άξονας της Γης είναι κάθετος στο επίπεδο του ουράνιου Ισημερινού.

## 2.2.2 Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ ΩΣ ΠΡΟΣ ΕΝΑ ΣΗΜΕΙΟ ΠΑΝΩ ΣΤΗ ΓΗ

Μέχρι εδώ εξετάσαμε την κίνηση της Γης γύρω από τον Ήλιο. Ακολούθως θα παρουσιαστούν τα μαθηματικά που χρειάζονται για τον υπολογισμό της θέσης του Ήλιου από οποιαδήποτε τοποθεσία πάνω στην επιφάνεια της Γης.

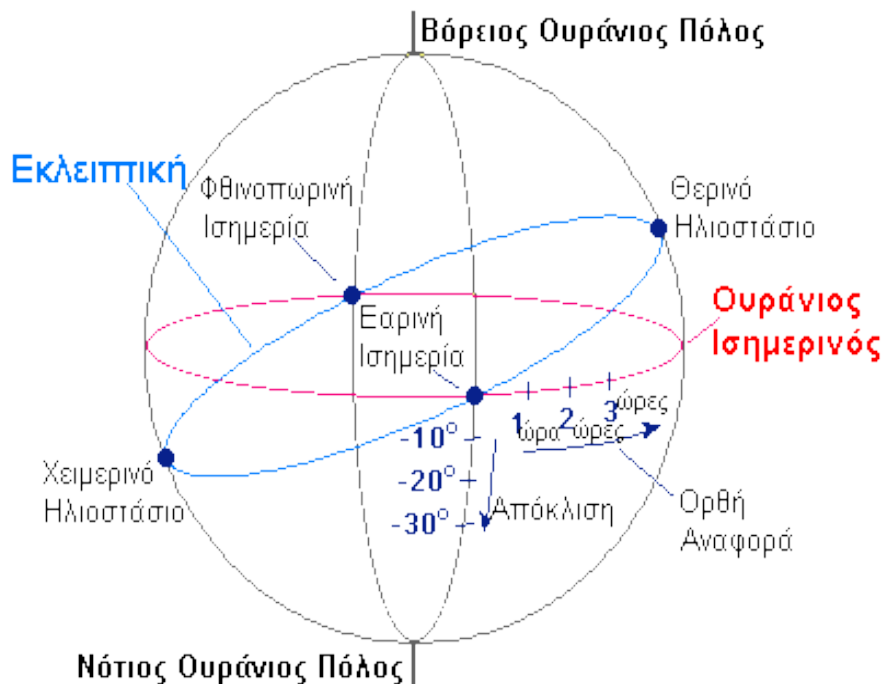
### 2.2.2.1 ΠΡΟΣΔΙΟΡΙΣΜΟΣ ΤΟΠΟΘΕΣΙΑΣ ΠΑΝΩ ΣΤΗ ΓΗ

Κάθε τοποθεσία πάνω στη Γη προσδιορίζεται μονοσήμαντα από δύο σφαιρικές συντεταγμένες: το γεωγραφικό μήκος και το γεωγραφικό πλάτος. Οι γωνίες μετρούνται από το κέντρο της Γης. Το επίπεδο αναφοράς είναι το επίπεδο του Ισημερινού, το οποίο τέμνει κάθετα τον πολικό άξονα της Γης, και συναντά την επιφάνεια της Γης κατά μήκος του Ισημερινού.

Το γεωγραφικό πλάτος ορίζεται από τη γωνία που σχηματίζεται ανάμεσα στο επίπεδο του Ισημερινού και τη νοητή γραμμή που ενώνει το κέντρο της Γης με την τοποθεσία που πρόκειται να προσδιοριστεί πάνω στην επιφάνεια της Γης. Οι κύκλοι που τέμνουν την επιφάνεια της Γης παράλληλα με τον Ισημερινό καθορίζουν τις θέσεις πάνω στην επιφάνεια της Γης με κοινό γεωγραφικό πλάτος. Εξ ορισμού το γεωγραφικό πλάτος λαμβάνεται θετικό στο Βόρειο ημισφαίριο και αρνητικό στο Νότιο.

Για τον προσδιορισμό του γεωγραφικού μήκους απαιτείται ένα επίπεδο αναφοράς το οποίο θα τέμνει κάθετα το επίπεδο του Ισημερινού και θα περιλαμβάνει τον πολικό άξονα της Γης. Αυτό το επίπεδο δημιουργεί δύο ημικύκλια από τον έναν πόλο έως τον άλλο, που ονομάζονται μεσημβρινοί. Στο γεωγραφικό μήκος μηδέν βρίσκεται εξ ορισμού ο μεσημβρινός που διέρχεται από το Greenwich στο Ηνωμένο Βασίλειο. Το γεωγραφικό μήκος οποιασδήποτε τοποθεσίας καθορίζεται από τη γωνία που σχηματίζεται ανάμεσα στο μεσημβρινό αναφοράς και το μεσημβρινό ο οποίος περνάει από την τοποθεσία της θέσης που πρόκειται να προσδιοριστεί. Εξ ορισμού το γεωγραφικό μήκος λαμβάνεται θετικό για τα γεωγραφικά μήκη που βρίσκονται δυτικά του Greenwich.

Αξίζει να σημειωθεί πως το σημείο αναφοράς στην επιφάνεια της Γης είναι η τομή μεταξύ του Ισημερινού και του μεσημβρινού αναφοράς, που περνάει από το Greenwich. Επίσης το γεωγραφικό πλάτος συμβολίζεται με  $\varphi$  και το γεωγραφικό μήκος με  $\alpha$ .



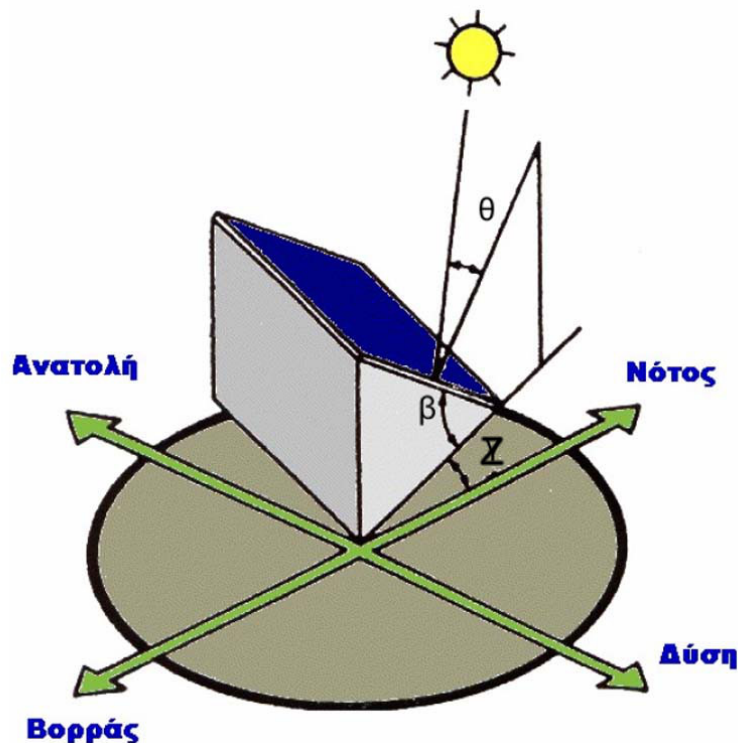
Σχήμα 2.2

### 2.2.2.2 ΤΟ ΟΡΙΖΟΝΤΙΟ ΣΥΣΤΗΜΑ

Προκειμένου να μπορεί να καθορίζεται η θέση ενός αστρικού σώματος από οποιαδήποτε τοποθεσία πάνω στην επιφάνεια της Γης, απαιτείται ένα νέο σύστημα συντεταγμένων, το οριζόντιο σύστημα. Ουσιαστικά, το οριζόντιο σύστημα αποτελεί ένα φυσικό σύστημα συντεταγμένων για κάθε παρατηρητή που βρίσκεται πάνω στην επιφάνεια της Γης. Το επίπεδο αναφοράς γίνεται το οριζόντιο επίπεδο που εφάπτεται με την επιφάνεια της Γης στη θέση του παρατηρητή. Το οριζόντιο επίπεδο τέμνει την ουράνια σφαίρα στον ορίζοντα. Η τομή μεταξύ της καθέτου στο επίπεδο αναφοράς στη θέση του παρατηρητή και της ουράνιας σφαίρας, προσδιορίζει το ζενίθ της συγκεκριμένης θέσης. Σύμφωνα με το οριζόντιο σύστημα, η Γη είναι σταθερή και βρίσκεται στο κέντρο του σύμπαντος. Τα αστέρια, συμπεριλαμβανομένου του Ήλιου, είναι στερεωμένα στην επιφάνεια της περιστρεφόμενης ουράνιας σφαίρας, ενώ οι αποστάσεις τους αγνοούνται.

Στο οριζόντιο σύστημα συντεταγμένων η θέση του Ήλιου ως προς το σημείο όπου βρίσκεται ο παρατηρητής μπορεί να καθοριστεί από δύο γωνίες, το ύψος  $\beta$  υπεράνω του ορίζοντα, και το αζιμούθιο  $Z$  ως προς το Νότο (Σχήμα 2.3). Οι γωνίες

μετρούνται από τη θέση στην οποία βρίσκεται ο παρατηρητής με επίπεδο αναφοράς το οριζόντιο έδαφος.



Σχήμα 2.3

Το ύψος υπεράνω του ορίζοντα είναι η γωνία υπό την οποία φαίνεται ο Ήλιος ως προς το οριζόντιο τοπικό επίπεδο του παρατηρητή στη συγκεκριμένη θέση στην επιφάνεια της Γης. Το εύρος τιμών για το ύψος του Ήλιου είναι από  $-90^\circ$  έως  $+90^\circ$ . Το ύψος γίνεται θετικό όταν ο Ήλιος βρίσκεται πάνω από τον ορίζοντα και αρνητικό όταν βρίσκεται από κάτω του. Η συμπληρωματική γωνία που σχηματίζεται ως προς το ζενίθ είναι  $90^\circ - \beta$ .

Το αζιμούθιο  $Z$  είναι η γωνία που δημιουργείται ανάμεσα σε μια ορισμένη κατεύθυνση και την προβολή του Ήλιου πάνω στο οριζόντιο επίπεδο αναφοράς. Η κατεύθυνση αναφοράς επιλέγεται να είναι ο Νότος. Οι αζιμουθιακές γωνίες που κατευθύνονται προς τη Δύση θεωρούνται θετικές.

Διευκρινίζεται ότι η ουράνια σφαίρα πραγματοποιεί την 24ωρη περιστροφή ως προς τον πολικό άξονα της Γης, όχι ως προς το νοητό άξονα που καθορίζεται μεταξύ του ζενίθ και του ναδίρ της τοποθεσίας.

## 2.2.3 ΗΛΙΑΚΗ ΩΡΑ & Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ

### 2.2.3.1 Η ΗΛΙΑΚΗ ΩΡΑ

Η ηλιακή ώρα έχει ως αναφορά τον Ήλιο και δεν ταυτίζεται με την τοπική ώρα. Προκειμένου να καθορισθεί σε μια τοποθεσία η ηλιακή ώρα  $T_{SOLAR}$ , θα πρέπει να ληφθούν υπόψη διορθώσεις που οφείλονται είτε στις διαφορετικές θέσεις πάνω στην επιφάνεια της Γης, είτε στις παρεκκλίσεις που προέρχονται από την ετήσια τροχιά της Γης.

Για τον προσδιορισμό της ηλιακής ώρας εφαρμόζονται οι ακόλουθες δύο διορθώσεις:

- ✓ Η πρώτη διόρθωση αναφέρεται στη θέση του παρατηρητή πάνω στην επιφάνεια της Γης. Προσδιορίζεται από τη διαφορά που υπάρχει ανάμεσα στο γεωγραφικό μήκος του μεσημβρινού του παρατηρητή  $L_{LOC}$  και του μεσημβρινού αναφοράς  $L_{ST}$  για τον καθορισμό της τοπικής ώρας (για την Ελλάδα αναλογεί ο μεσημβρινός  $L_{ST} = -30^\circ$ , που διέρχεται ανατολικά της Κωνσταντινουπόλεως). Για κάθε τοποθεσία πάνω στη Γη, ο μεσημβρινός αναφοράς που χρησιμοποιείται για τον καθορισμό της τοπικής ώρας,  $L_{ST}$ , μπορεί να προσδιορίζεται από την ακόλουθη εξίσωση:

$$L_{ST} = (15^\circ/h)\Delta T_{GMT}$$

$\Delta T_{GMT}$ : η διαφορά μεταξύ της ώρας GMT (Greenwich Mean Time) και της τοπικής ώρας, που συνιστά ακέραιο αριθμό ωρών (πχ για την Ελλάδα = -2)

$L_{ST}$ : ο μεσημβρινός αναφοράς για τον καθορισμό της τοπικής ώρας

$15^\circ/h$ : =  $360^\circ/24$  ώρες

- ✓ Η δεύτερη διόρθωση εισάγεται ανάλογα με την ημέρα του έτους, μέσω της λεγόμενης «εξίσωσης χρόνου»,  $E$ , και λαμβάνει υπόψη της τη μεταβολή της χρονικής διάρκειας (σε min) που απαιτεί ο Ήλιος για να μεταφερθεί κατά μήκος της ηλιακής τροχιάς του στην ουράνια σφαίρα και να διανύσει τον ουρανό του παρατηρητή.

Η διόρθωση που εισάγεται με την εξίσωση του χρόνου  $E$  αποδίδεται συνήθως από εμπειρικές μαθηματικές σχέσεις της μορφής:



$$E = 229,2 \cdot (0,000075 + 0,001868 \cdot \cos B - 0,032077 \cdot \sin B - 0,014615 \cdot \cos 2B - 0,04089 \cdot \sin 2B)$$

Όπου  $B$  η γωνία ημέρας, η οποία προσδιορίζεται από την ημέρα του έτους,  $DoY$  (*Day of Year*), κατά την οποία επιχειρείται ο προσδιορισμός της ηλιακής ώρας. Η γωνία ημέρας μπορεί να συσχετιστεί με την πρώτη ημέρα του ημερολογιακού έτους, οπότε για την παραπάνω εξίσωση χρόνου ισχύει:

$$B = (DoY - 1) \cdot \frac{360^\circ}{365}$$

Η παράμετρος λαμβάνει ακέραιες τιμές που μεταβάλλονται από 1 έως 365.

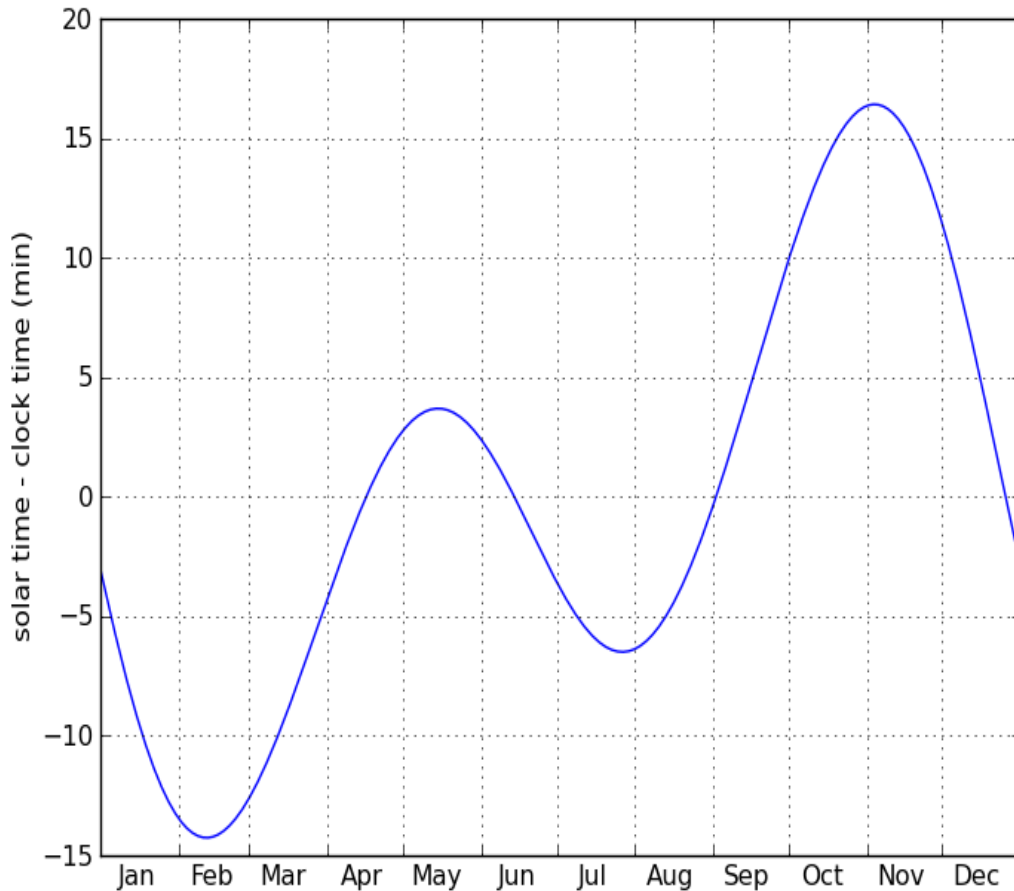
Μια άλλη περισσότερο εύχρηστη εμπειρική σχέση για την περιγραφή της εξίσωσης χρόνου,  $E$ , που συνήθως χρησιμοποιείται στον προσδιορισμό της θέσης του Ήλιου, είναι η ακόλουθη:

$$E = 9,87 \cdot \sin 2B - 7,53 \cdot \cos B - 1,5 \cdot \sin B$$

Όπου η γωνία ημέρας,  $B$ , προσδιορίζεται με αναφορά την ημέρα κατά την οποία ο Ήλιος θα εισέλθει στην εαρινή ισημερία, και δίνεται από τη σχέση:

$$B = (DoY - 81) \cdot \frac{360^\circ}{364}$$

Η διόρθωση (σε λεπτά) που εισάγεται με την εξίσωση χρόνου,  $E$ , αντισταθμίζει την εκκεντρότητα μεταξύ της τροχιάς της Γης και της γωνίας του άξονα της Γης. Η εξίσωση χρόνου,  $E$ , φαίνεται στο σχήμα 2.2.



**Σχήμα 2.2** Η γραφική παράσταση του όρου διόρθωσης που εισάγει η εξίσωση χρόνου  $E$

Όταν όλα τα ανωτέρω μεγέθη για τους παράγοντες διόρθωσης είναι γνωστά, μπορεί να υπολογιστεί η ηλιακή ώρα,  $T_{SOLAR}$ , σύμφωνα με την εξίσωση:

$$T_{SOLAR} = T_{LOCAL} + E + 4 \frac{\min}{o} (L_{ST} - L_{LOCAL})$$

$T_{SOLAR}$ : η ηλιακή ώρα, για δεδομένο τόπο, δεδομένη ημέρα, σε συγκεκριμένη ηλιακή θέση

$T_{LOCAL}$ : η αντίστοιχη τοπική ώρα (χρησιμοποιείται η χειμερινή ώρα καθ' όλη τη διάρκεια του έτους)

$E$ : η διόρθωση που εισάγεται από την εξίσωση χρόνου για την ημέρα υπολογισμού

$L_{ST}$ : ο κεντρικός μεσημβρινός για την τοπική ώρα ζώνης,  $-30^\circ$  για την Ελλάδα

$L_{LOCAL}$ : το γεωγραφικό μήκος της τοποθεσίας όπου προσδιορίζεται η ώρα

επίσης ο όρος “4 λεπτά της ώρας / γωνία μοίρας” προέρχεται από το γεγονός ότι η Γη περιστρέφεται γύρω από τον άξονα της κατά  $1^\circ$ , εντός 4 λεπτών της ώρας.

### 2.2.3.2 Η ΘΕΣΗ ΤΟΥ ΗΛΙΟΥ

Στόχος είναι να καθοριστεί η θέση του Ήλιου με δύο γωνίες στο οριζόντιο σύστημα, την ηλιακή απόκλιση,  $\delta$ , και την ωριαία γωνία,  $\omega$ .

#### *Ηλιακή απόκλιση, $\delta$*

Όπως έχει ήδη αναφερθεί, η γωνία  $\delta$  που σχηματίζεται ανάμεσα στο επίπεδο του ουράνιου Ισημερινού και της νοητής ευθείας η οποία ενώνει τα κέντρα της Γης και του Ηλίου, ονομάζεται ηλιακή απόκλιση. Η γωνία  $\delta$  μπορεί να υπολογιστεί με τη βοήθεια της παρακάτω εξίσωσης:

$$\delta(^{\circ}) = 23,45^{\circ} \sin\left[\left(DoY + 284\right) \frac{360^{\circ}}{365}\right]$$

Ο όρος 284 αποδίδει τον αριθμό των ημερών που απαιτούνται να παρέλθουν μετά την εαρινή ισημερία ώστε η θέση του Ήλιου να αντιστοιχεί στην έναρξη του ημερολογιακού έτους.

Μια εναλλακτική σχέση μπορεί να προκύψει από το γεγονός ότι η πρώτη ημέρα του ημερολογιακού έτους υπολείπεται κατά 81 ημέρες από την επερχόμενη εαρινή ισημερία, όποτε η σχέση υπολογισμού της ηλιακής απόκλισης γίνεται:

$$\delta(^{\circ}) = 23,45^{\circ} \sin\left[\left(DoY - 81\right) \frac{360^{\circ}}{365}\right]$$

Στη διάρκεια οποιασδήποτε ημερολογιακής ημέρας,  $DoY$ , οι δύο σχέσεις δίνουν ταυτόσημα αποτελέσματα ηλιακής απόκλισης ανεξαρτήτως ημισφαιρίου. Η ηλιακή απόκλιση,  $\delta$ , είναι μηδέν και στις δύο ισημερίες, ενώ έχει τιμές  $+23^{\circ}27'$  ( $= 23,45^{\circ}$ ) κατά το θερινό ηλιοστάσιο και  $-23^{\circ}27'$  κατά το χειμερινό ηλιοστάσιο.

## **Ωριαία γωνία $\omega$**

Στην αστρονομία η ωριαία γωνία συνιστά μία από τις συντεταγμένες του Ισημερινού συστήματος συντεταγμένων, το οποίο χρησιμοποιείται προκειμένου να αποτυπώνονται οι θέσεις των αστερών πάνω στην ουράνια σφαίρα.

Η ωριαία γωνία ενός σημείου, είναι η γωνία  $\omega$  που σχηματίζεται μεταξύ των δύο ημιεπιπέδων που ορίζονται από:

- i.* τον πολικό άξονα της Γης και το ζενίθ της τοποθεσίας
- ii.* τον πολικό άξονα της Γης και το σημείο που πρόκειται να περιγραφεί πάνω στην ουράνια σφαίρα.

Η ωριαία γωνία,  $\omega$ , λαμβάνει αρνητικές τιμές ανατολικά του μεσημβρινού που εμπεριέχει το ζενίθ (πρωί) και θετικές τιμές δυτικά του (απόγευμα). Η γωνιακή μετατόπιση της Γης ως προς τον άξονα περιστροφής της αναπαριστά χρόνο, με τη γωνία των  $15^\circ$  να αντιστοιχεί σε μία ώρα. Γενικά, για τη γωνία ώρας ισχύει ότι:

$$\omega = 15^\circ/h \cdot (T_{SOLAR} - 12h)$$

όπου 12 η ώρα για την ηλιακή μεσημβρία. Κατά τη στιγμή της ηλιακής μεσημβρίας, ο Ήλιος βρίσκεται στο ζενίθ της περιοχής, οπότε  $\omega = 0$ .

## **Προσδιορισμός της θέσης του Ήλιου**

Για τον προσδιορισμό της θέσης του Ήλιου, δεδομένης της γωνίας ώρας, χρειάζεται να υπολογίσουμε το ύψος,  $\beta$ , του Ήλιου καθώς και την ηλιακή αζιμούθια γωνία,  $Z$ .

Το ύψος του Ήλιου υπολογίζεται από τη σχέση:

$$\sin\beta = \sin\delta \cdot \sin\varphi + \cos\delta \cdot \cos\varphi \cdot \cos\omega$$

Ενώ η ηλιακή αζιμούθια γωνία υπολογίζεται από τη σχέση:

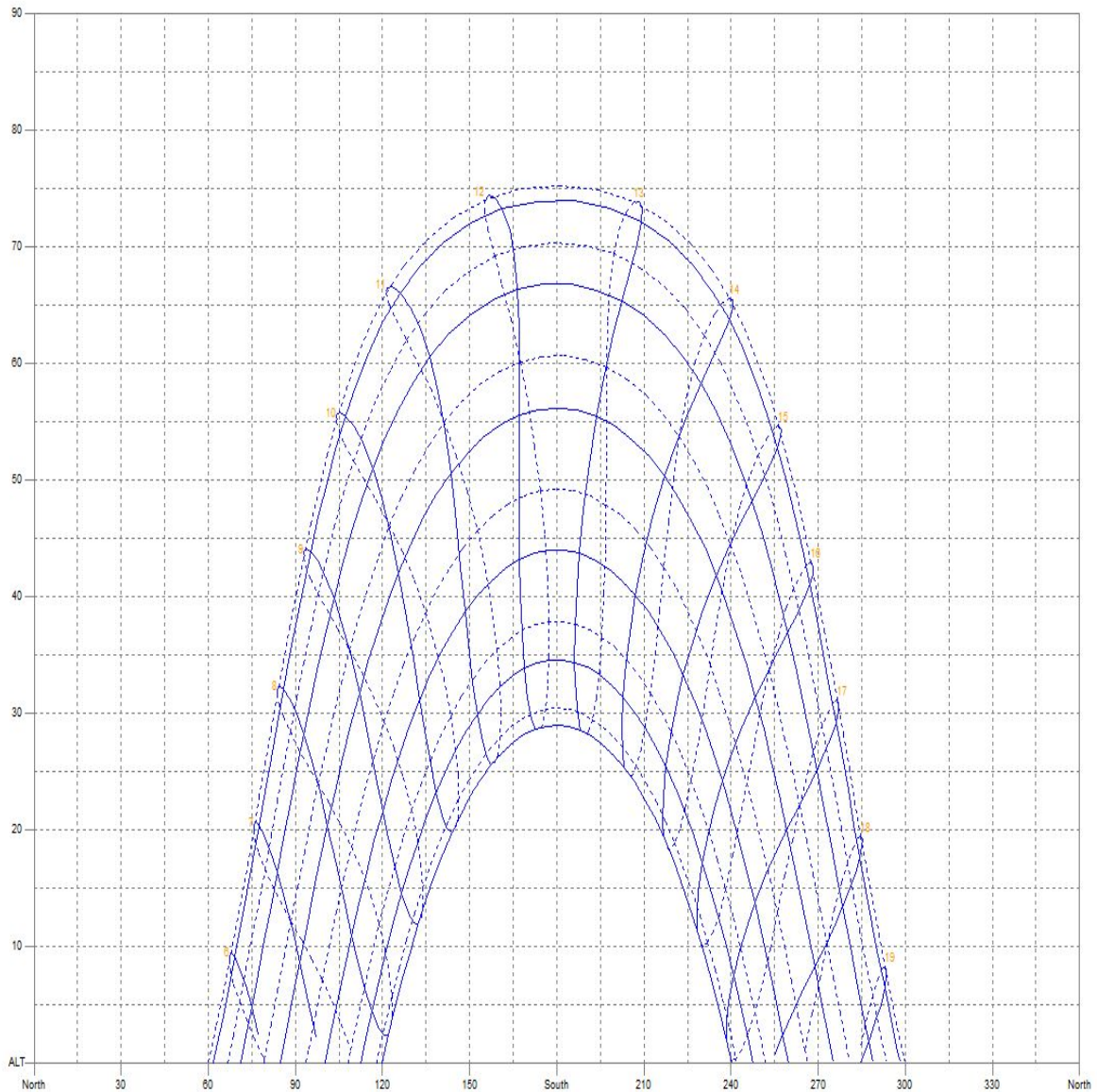
$$\cos Z = \frac{\sin\beta \cdot \sin\varphi - \sin\delta}{\cos\beta \cdot \cos\varphi}$$

όπου

$\beta$ : το ύψος του Ήλιου από τη συγκεκριμένη περιοχή

$Z$ : η γωνία για το ηλιακό αζιμούθιο ως προς νότο, στη συγκεκριμένη περιοχή και ώρα

Ο τύπος προσδιορισμού του αζιμούθιου ισχύει ως έχει μόνο όταν  $\omega \leq 0$ . Στην αντίθετη περίπτωση ( $\omega > 0$ ) ο τύπος παρέχει το  $-\cos Z$ . Διευκρινίζεται επίσης πως όλες οι γωνίες είναι σε μοίρες.



**Σχήμα 2.3** Η μεταβολή του ηλιακού ύψους ( $\beta$ ) και της αζιμούθιας γωνίας ως προς νότο κατά τη διάρκεια των ηλιοστασίων & των ισημεριών στην περιοχή της Αθήνας

## Κεφάλαιο III

# Ανανεώσιμες πηγές ενέργειας

Στο κεφάλαιο αυτό παρουσιάζονται εν συντομία οι ανανεώσιμες πηγές ενέργειας και αναφέρονται πιο αναλυτικά αυτές που χρησιμοποιούν τον ήλιο για παραγωγή ηλεκτρικής ενέργειας.



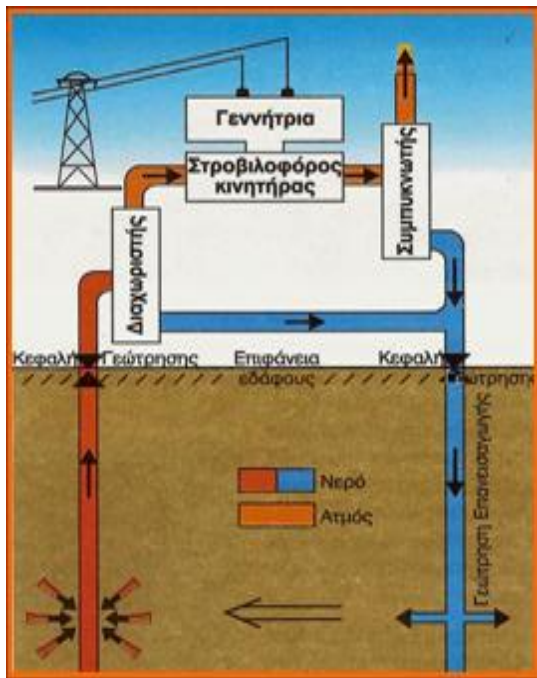


### 3.0 ΕΙΣΑΓΩΓΗ

Ο ήλιος είναι η μοναδική πηγή θερμότητας και φωτός που δέχεται η επιφάνεια της Γης προκειμένου να διατηρείται η ζωή του πλανήτη. Η ηλιακή ακτινοβολία παρέχει τεράστια ποσά ενέργειας στη Γη. Δοθείσης της πυκνότητας ισχύος της ηλιακής σταθεράς, η οποία θεωρείται σήμερα ίση με  $1367 \text{ W/m}^2$ , και της περιμέτρου των  $40000 \text{ km}$  που αντιστοιχούν στην Γη, αλλά και η εξασθένηση της ηλιακής ακτινοβολίας που προκαλείται από την ατμόσφαιρα, προκύπτει πως η ετήσια μέση ισχύς που προσπίπτει συνολικά στη Γη στο επίπεδο της θάλασσας είναι ίση με  $102000 \text{ TW}$ . Το συνολικό ποσό ενέργειας που ακτινοβολείται από τον ήλιο στην επιφάνεια της γης είναι ίσο με  $10.000$  φορές περίπου την ετήσια παγκόσμια ενεργειακή κατανάλωση. Ενώ η ηλιακή ενέργεια που φθάνει στη γη κατά τη διάρκεια τριών ημερών είναι περισσότερη απ' ό,τι το κατ' εκτίμηση σύνολο των ορυκτών καυσίμων στη γη. Κατά μέσο όρο, προσπίπτουν  $1750 \text{ kWh}$  σε κάθε τετραγωνικό μέτρο κάθε χρόνο.

Η Ελλάδα αποτελεί προνομιακή χώρα στην ηλιοφάνεια, όπως φαίνεται και στο χάρτη για την κατανομή του ηλιακού δυναμικού του σχήματος 3.1. Το πλεονέκτημα αυτό δεν είναι φυσικά το μοναδικό. Άλλα πλεονεκτήματα είναι η ενέργεια που περικλείουν οι άνεμοι αλλά και τα ποτάμια, παρότι δεν είναι τεραστίων διαστάσεων ως προς την παροχή νερού. Τα πλεονεκτήματα αυτά μπορούν να αξιοποιηθούν πέραν άλλων λόγων για την παραγωγή ηλεκτρικής ενέργειας, μέσω των ανανεώσιμων πηγών ενέργειας (ΑΠΕ).

Οι ΑΠΕ σκοπό έχουν την αξιοποίηση φυσικών φαινομένων για την παραγωγή ηλεκτρικής ενέργειας. Μορφές ΑΠΕ είναι η αιολική ενέργεια η οποία αξιοποιεί τους ανέμους, η ηλιακή ενέργεια η οποία αξιοποιεί την ηλιακή ακτινοβολία που προσπίπτει στη Γη, η γεωθερμία ή γεωθερμική ενέργεια η οποία αξιοποιεί τη φυσική θερμική ενέργεια της Γης που διαρρέει από το θερμό εσωτερικό του πλανήτη προς την επιφάνεια και τέλος η ενέργεια που παράγεται από τα υδροηλεκτρικά εργοστάσια που αξιοποιούν τη δυναμική ενέργεια του νερού.



**Σχήμα 3.1** Γεωθερμική, Ηλιακή, Αιολική και Υδροηλεκτρική ενέργεια

Οι ΑΠΕ μπορούν να βοηθήσουν στην κάλυψη των διαρκώς αυξανόμενων ενεργειακών αναγκών της χώρας και να μειώσουν τυχόν εισαγωγές ή κοστοβόρες διαδικασίες με σημαντικές συνέπειες στο περιβάλλον.

Η ένταξη των ανανεώσιμων πηγών ενέργειας (ΑΠΕ) στο εθνικό σύστημα παραγωγής ενέργειας πρέπει να γίνεται με κριτήρια τη δημιουργία ενός σταθερού συστήματος παραγωγής και διανομής ηλεκτρικής ενέργειας, που θα καλύπτει κάθε γωνία της χώρας με φθινό ρεύμα σε κάθε οικογένεια, εξασφαλίζοντας την προστασία του περιβάλλοντος, το σεβασμό περιοχών φυσικού κάλους και φυσικά την προστασία της ζωής και της υγείας του τοπικού πληθυσμού. Ένα τέτοιο σύστημα αξιοποιεί τις ΑΠΕ για τη μείωση της εξάρτησης της χώρας στο τομέα της ενέργειας, την εξασφάλιση φθινορού ρεύματος για όλους μέσω της ανάπτυξης της τεχνολογίας και των φυσικών πηγών ενέργειας που υπάρχουν στον τόπο μας.

Συνεπώς η ένταξη τους δεν πρέπει να υπηρετεί το σκοπό του κέρδους όπως γίνεται σήμερα με αποτελέσματα ιδιαίτερα καταστρεπτικά για τον τόπο, όπως είναι η μη αξιοποίηση παραγωγικών εδαφών για την καλλιέργεια αγροτικών προϊόντων, η καταστροφή περιοχών φυσικού κάλους, αλλά και περιοχών τουριστικού ενδιαφέροντος, η κακή ποιότητα συντήρησης (συνολικά όχι μόνο στις ΑΠΕ), η μη πρόβλεψη των συνεπειών συνολικά μετά τη φυσική γήρανση των εγκαταστάσεων και του εξοπλισμού τέτοιων έργων και φυσικά η παραγωγή ακριβού ηλεκτρικού ρεύματος.

### **3.1 ΗΛΙΑΚΗ ΕΝΕΡΓΕΙΑ**

Στην παρούσα διπλωματική μας ενδιαφέρει η ηλιακή ενέργεια και ακτινοβολία διότι έχουν το ιδιαίτερο χαρακτηριστικό του Ήλιου, ο οποίος συνεχώς κινείται ως προς ακίνητο παρατηρητή πάνω στη Γη. Η συνεχής αυτή κίνηση επί της ουσίας είναι η κίνηση της βασικής πηγής ενέργειας. Αυτό δημιουργεί δυσκολίες όταν απαιτείται συνεχής και μέγιστη απόδοση. Για παράδειγμα κατά τη διάρκεια της νύχτας είναι αδύνατο να αξιοποιήσουμε την ηλιακή ακτινοβολία για παραγωγή ηλεκτρικής ενέργειας.

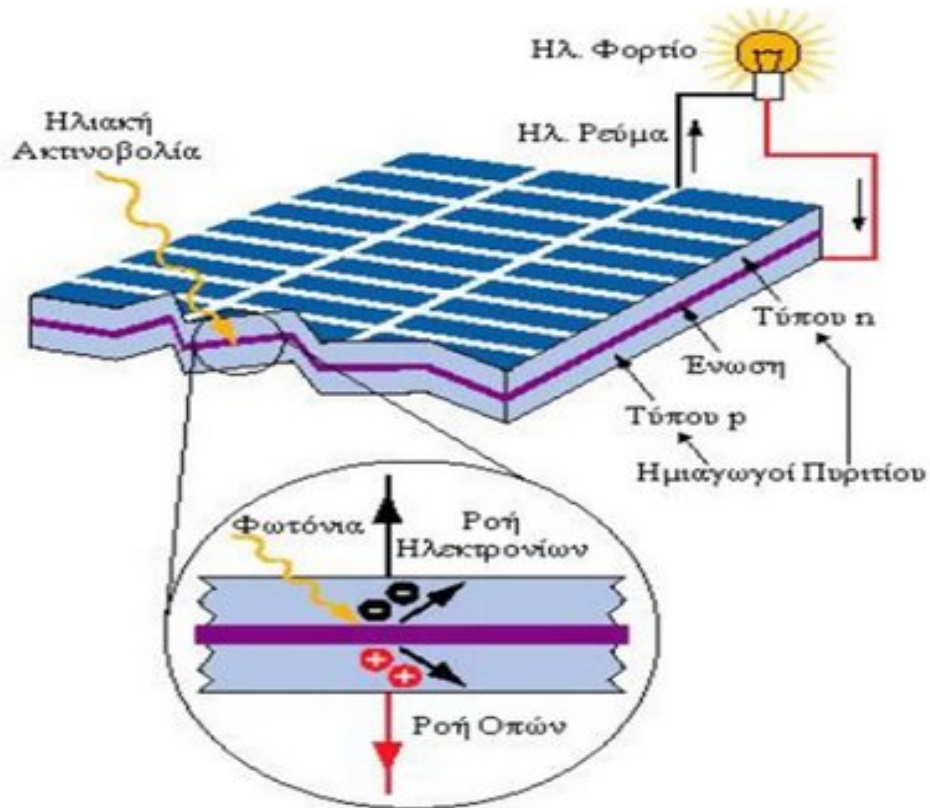
Όμως και κατά τη διάρκεια της μέρας υπάρχουν αντικειμενικές δυσκολίες στην αποδοτικότερη αξιοποίησή της ηλιακής ακτινοβολίας για την παραγωγή ηλεκτρικής ενέργειας. Τέτοιες δυσκολίες είναι η συνεχής αλλαγή θέσης του Ήλιου, οι καιρικές συνθήκες, η κλίση του εδάφους και ο προσανατολισμός του.

Την ηλιακή ακτινοβολία μπορούμε την χρησιμοποιήσουμε για την παραγωγή ηλεκτρικής ενέργειας με δυο τρόπους: μέσω των φωτοβολταϊκών στοιχείων και μέσω των συγκεντρωτικών ηλιακών συστημάτων.

### 3.1.1 ΦΩΤΟΒΟΛΤΑΪΚΑ ΣΤΟΙΧΕΙΑ

Η παραγωγή ηλεκτρικής ενέργειας από φωτοβολταϊκά στοιχεία αποτελεί κομμάτι μιας ολόκληρης επιστήμης που σχετίζεται με τα υλικά, τις ιδιότητές τους και την συμπεριφορά τους ως ηλεκτρικά στοιχεία. Η εξέλιξη αυτής της επιστήμης είναι βασική για την αύξηση της απόδοσης των φωτοβολταϊκών στοιχείων στην παραγωγή ηλεκτρικής ενέργειας, αλλά και στη μείωση τους κόστους τους ώστε να παράγεται φθηνό ηλεκτρικό ρεύμα χωρίς να επιβαρύνεται το περιβάλλον.

Το φωτοβολταϊκό στοιχείο (λέγεται και φωτοβολταϊκή κυψέλη) αποτελεί την ελάχιστη δομική μονάδα για την παραγωγή ηλεκτρικής ενέργειας από την ηλιακή ακτινοβολία. Τα συνήθη εμπορικά φωτοβολταϊκά στοιχεία αποτελούνται από διόδους επαφής p-n και έχουν εμβαδό φωτιζόμενης επιφάνειας που ποικίλλει από λίγα τετραγωνικά εκατοστά μέχρι περίπου 180 τετραγωνικά εκατοστά. Τα φωτοβολταϊκά στοιχεία κατατάσσονται, ανάλογα με το υλικό της κατασκευής τους, σε ΦΒ στοιχεία μονοκρυσταλλικού πυριτίου, πολυκρυσταλλικού πυριτίου, λεπτών υμενίων και άμορφου πυριτίου. Κάθε φωτοβολταϊκό στοιχείο όταν δέχεται την ηλιακή ακτινοβολία, παρέχει στην έξοδο του συνεχή τάση περίπου ίση προς 0,5 V. Η μέγιστη ένταση του ρεύματος που παράγεται στην έξοδο ενός ΒΦ στοιχείου είναι ανάλογη της επιφάνειας του και προσδιορίζεται κυρίως από την τιμή της πυκνότητας ισχύος της προσπίπτουσας φωτεινής ακτινοβολίας.



Σχήμα 3.2 Σχεδιαστική απεικόνιση λειτουργίας ΦΒ στοιχείου




Τα ΦΒ στοιχεία συνδέονται ηλεκτρικά μεταξύ τους για το σχηματισμό πλαισίων (modules). Το φωτοβολταϊκό πλαίσιο αποτελεί τη βασική κατασκευαστική μονάδα που χρησιμοποιείται για τη δημιουργία κάθε ΦΒ εγκατάστασης, ανεξαρτήτως μεγέθους εγκατεστημένης ΦΒ ισχύος.



**Σχήμα 3.3** ΦΒ πάνελο

Οι ΦΒ εγκαταστάσεις χρησιμοποιούνται είτε για την ηλεκτρική τροφοδότηση απομονωμένων οικισμών, οικιών ή εγκαταστάσεων, είτε κυρίως για παραγωγή ηλεκτρικής ενέργειας η οποία διοχετεύεται μέσω των τοπικών σταθμών χαμηλής ή μέσης τάσης στα κατά τόπους δίκτυα ηλεκτρικής ενέργειας.

Είναι κατανοητό από τον τρόπο λειτουργίας τους, πως όσο περισσότερη ηλιακή ακτινοβολία δέχονται τα ΒΦ στοιχεία, τόσο μεγαλύτερη είναι και η ένταση του ρεύματος, συνεπώς και η παραγωγή ηλεκτρικής ενέργειας.

Συγκριτικός πίνακας φωτοβολταϊκών τεχνολογιών (εξοπλισμός που κυκλοφορεί στην αγορά στις αρχές 2008)			
ΤΥΠΟΣ	'Λεπτού υμενίου' ή 'Thin Film'	Πολυκρυσταλλικά	Μονοκρυσταλλικά
Εμφάνιση			
Απόδοση	α-Si: 4,2-6,6% μ-Si: 8,1-8,5% CIS-CIGS: 6-11% CdTe: 6-11,1%	11-14,8%	11-19,3%
Απαιτούμενη επιφάνεια ανά kWp	9-25 m <sup>2</sup>	7-9 m <sup>2</sup>	5,5-9 m <sup>2</sup>
Μέση ετήσια παραγωγή ενέργειας (kWh ανά kWp) <small>(μέση τιμή για Ελλάδα και για ένα τυπικό σύστημα με νότιο προσανατολισμό και κατάλληλη κλίση)</small>	1.300-1.450	1.300	1.300
Μέση ετήσια παραγωγή ενέργειας (kWh ανά m <sup>2</sup> ) <small>(μέση τιμή για Ελλάδα και για ένα τυπικό σύστημα με νότιο προσανατολισμό και κατάλληλη κλίση)</small>	50-160	145-185	145-235
Ετήσια μείωση εκπομπών διοξειδίου του άνθρακα (kg CO <sub>2</sub> ανά kWp)	1.300-1.450	1.300	1.300

Σχήμα 3.4





### 3.1.2 ΣΥΓΚΕΝΤΡΩΤΙΚΑ ΗΛΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

Οι τεχνολογίες ηλιακής συγκέντρωσης είναι διατάξεις που συγκεντρώνουν την ηλιακή ενέργεια εστιάζοντας την ηλιακή ακτινοβολία σ' ένα σημείο ή μια γραμμή εστίασης.

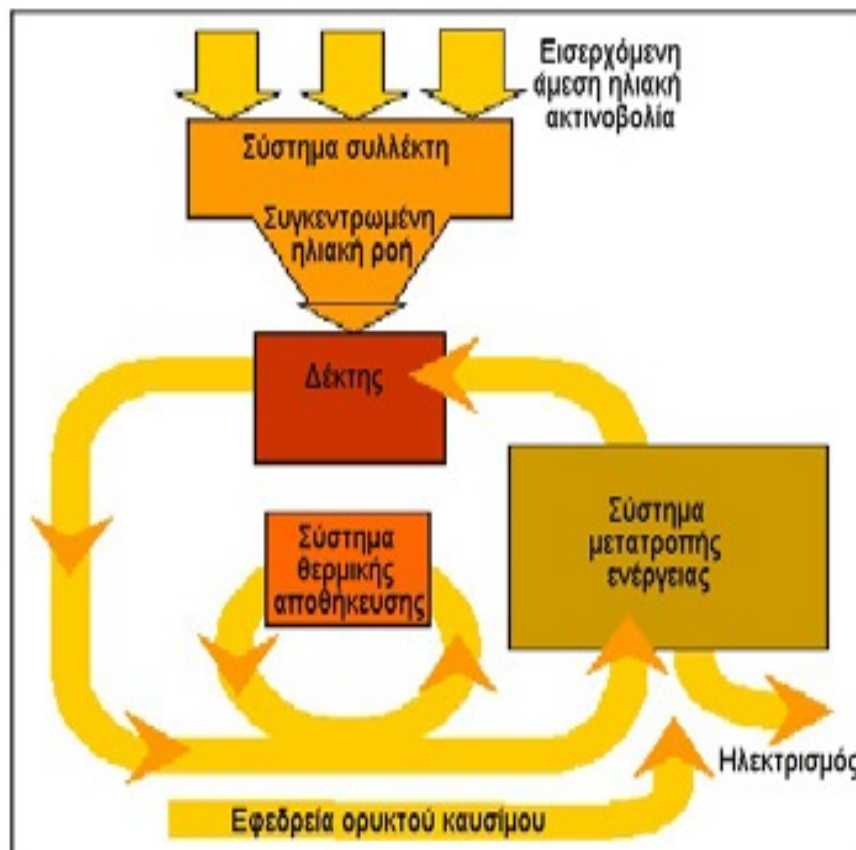


Σχήμα 3.5 Συγκεντρωτικό ηλιακό σύστημα

Τα συστήματα Ηλεκτροπαραγωγής με Ηλιακή Συγκέντρωση (ΗΗΣ) βρίσκονται ήδη σε λειτουργία με μονάδες ισχύος πολλών MW να παράγουν ηλιακή ηλεκτρική ενέργεια παγκοσμίως τα τελευταία 15 έτη. Τα τελικά στάδια ηλεκτροπαραγωγής με χρήση συστημάτων ΗΗΣ είναι όμοια με αυτά της συμβατικής ηλεκτροπαραγωγής (σχήμα 3.5), καθώς η τελική διεργασία ενεργειακής μετατροπής βασίζεται στη χρήση ατμού ή αερίου για την περιστροφή στροβίλων η την κίνηση ενός εμβόλου σε μια μηχανή Stirling. Σε ένα σύστημα ΗΗΣ ο ατμός ή το θερμό αέριο παράγεται μέσω της συγκέντρωσης της άμεσης ηλιακής ακτινοβολίας.

Όλες οι τεχνολογίες ηλιακής θερμικής ηλεκτροπαραγωγής περιλαμβάνουν έναν αριθμό βασικών σταδίων:

- Συλλογή της άμεσης ηλιακής ακτινοβολίας μέσω ενός συστήματος συλλεκτών.
- Συγκέντρωση (ή εστίαση) της ακτινοβολίας σε έναν δέκτη.
- Μετατροπή της από το δέκτη σε θερμική ενέργεια.
- Μεταφορά της θερμικής ενέργειας στο σύστημα ενεργειακής μετατροπής.
- Μετατροπή της θερμικής ενέργειας σε ηλεκτρισμό.



**Σχήμα 3.6** Βασικές αρχές της τεχνολογίας Ηλεκτροπαραγωγής με Ηλιακή Συγκέντρωση.

Πολλά τέτοια συστήματα είναι δυνατά, τα οποία μάλιστα μπορούν να συνδυάζονται με άλλες ανανεώσιμες και μη τεχνολογίες σε υβριδικά συστήματα. Ωστόσο θεωρείται, ειδικότερα από το πρόγραμμα SolarPaces της Διεθνούς Επιτροπής Ενέργειας (IEA), ότι οι τέσσερις πολλά υποσχόμενες αρχιτεκτονικές θερμικής ηλιακής ηλεκτροπαραγωγής, οι οποίες χαρακτηρίζονται από την τεχνική που χρησιμοποιείται για τη συγκέντρωση της ηλιακής ενέργειας, είναι:

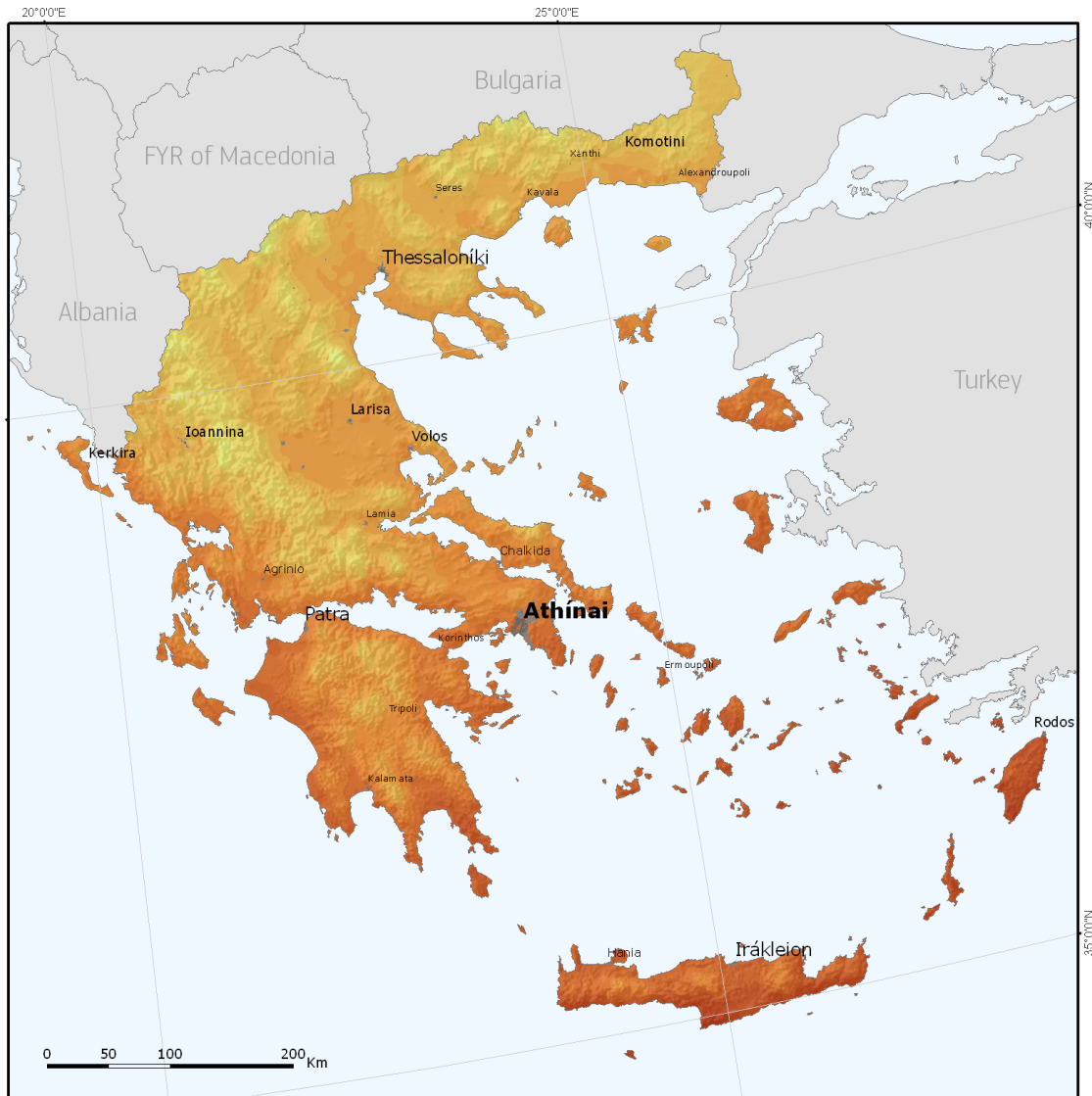
- τα παραβολικά κοίλα,
- οι ηλιακοί πύργοι ισχύος,
- τα συστήματα δίσκου, και
- ο συλλέκτης Fresnel.

Το βασικό συστατικό του συστήματος Ηλεκτροπαραγωγής με Ηλιακή Συγκέντρωση (ΗΗΣ) είναι ο ηλιακός συλλέκτης (ή αλλιώς ανακλαστήρας-συγκεντρωτής-ηλιοστάτης). Αποτελείται από την ανακλαστική επιφάνεια (κάτοπτρα-καθρέπτες) που κατευθύνει τις ηλιακές ακτίνες στο δέκτη, μια μεταλλική δομή στήριξης, το σύστημα παρακολούθησης του ήλιου και το σύστημα ελέγχου.

Είναι φανερό πως τα ΦΒ συστήματα όσο και τα συγκεντρωτικά ηλιακά συστήματα εξαρτώνται, για την λειτουργία τους, όχι μόνο από την ηλιακή ακτινοβολία που προσπίπτει στη Γη, αλλά και την ένταση της ακτινοβολίας αυτής. Οι τεχνολογίες που περιγράφηκαν παραπάνω είναι πιο αποδοτικές σε χώρες με μεγαλύτερη ηλιοφάνεια καθ' όλη τη διάρκεια του έτους, όπως η Ελλάδα. Στο σχήμα 4. Φαίνεται η κατανομή του ηλιακού δυναμικού για τον ελλαδικό χώρο.

# Global irradiation and solar electricity potential Horizontally mounted photovoltaic modules

## GREECE / ΕΛΛΑΔΑ



Yearly sum of global irradiation  
[kWh/m<sup>2</sup>]

1300 1400 1500 1600 1700 1800 1900 2000



975 1050 1125 1200 1275 1350 1425 1500

- Urban area
- Water body

Yearly sum of solar electricity generated by 1kW<sub>p</sub>  
system with performance ratio 0.75  
[kWh/kW<sub>peak</sub>]

Projection: Lambert Azimuthal Equal Area, WGS84, lat 52° lon 10°  
Source of ancillary data: CORINE Land Cover  
DTM SRTM-30  
GISCO database  
Geonames  
Natural Earth



Authors: Thomas Huld, Irene Pinedo-Pascua  
European Commission - Joint Research Centre  
Institute for Energy and Transport, Renewable Energy Unit  
PVGIS <http://re.jrc.ec.europa.eu/pvgis/>

Σχήμα 3.7 Κατανομή του ηλιακού δυναμικού για τον ελλαδικό χώρο

## Κεφάλαιο IV

# Φωτοβολταϊκοί Ιχνηλάτες παρακολούθησης του Ηλίου (Solar Tracker)

Στο κεφάλαιο αυτό αναλύονται οι ιχνηλάτες παρακολούθησης του ήλιου, οι κατηγορίες αυτών, αλλά και η χρησιμότητα τους στις ανανεώσιμες πηγές ενέργειας (στα φωτοβολταϊκά)



## 4.1 ΙΧΝΗΛΑΤΕΣ (trackers)

Όπως έχει προαναφερθεί υπάρχουν πολλοί λόγοι οι οποίοι μας δημιουργούν την ανάγκη για συνεχή παρακολούθηση της κίνησης του ήλιου. Η ανάγκη αυτή καλύπτεται από τους ιχνηλάτες (trackers). Ο ιχνηλάτης είναι διάταξη η οποία παρακολουθεί συνεχώς την κίνηση του ήλιου. Η διάταξη περιλαμβάνει το μηχανολογικό μέρος (ρότορας, βάση, διάταξης για την κίνηση κτλ.) και το ηλεκτρονικό κομμάτι, όπου επί της ουσίας είναι οι διατάξεις που χρειάζονται ώστε να ελέγχεται η κίνηση του ιχνηλάτη, βάση προκαθορισμένων εντολών και αλγορίθμων.

## 4.2 ΦΩΤΟΒΟΛΤΑΪΚΟΙ ΙΧΝΗΛΑΤΕΣ (solar trackers)

Ο φωτοβολταϊκός ιχνηλάτης είναι μια κινητή βάση πάνω στην οποία τοποθετείται το φωτοβολταϊκό στοιχείο. Αυτή η μηχανολογική διάταξη δίνει την δυνατότητα στο φωτοβολταϊκό να αλλάζει συνεχώς θέσεις προσπαθώντας να εγκλωβίσει στο μέγιστο δυνατό τις ακτίνες του ήλιου. Μέσω αυτής της διαδικασίας το φωτοβολταϊκό είναι καθ' όλη τη διάρκεια της ημέρας κάθετο στις ακτίνες του ήλιου εγκλωβίζοντας κάθε στιγμή τη μέγιστη δυνατή ηλιακή ενέργεια για μεγαλύτερη απόδοση της φωτοβολταϊκής εγκατάστασης.

Για την υλοποίηση του ηλιακού ιχνηλάτη χρησιμοποιούνται μεταλλικά μέρη για τη στήριξη των φωτοβολταϊκών πανέλων, τα οποία πληρούν συγκεκριμένες προϋποθέσεις ώστε να καλύπτονται αντίστοιχες προδιαγραφές για την αντοχή της διάταξης. Βασικότερες προϋποθέσεις οι οποίες πρέπει να καλύπτονται είναι το βάρος, πρέπει να αντέχει το φορτίο των φωτοβολταϊκών πανέλων που θα σηκώσει, η αντοχή στους ανέμους ώστε να μην καταστραφεί από έναν ισχυρό άνεμο, η αξιοπιστία των υλικών σε συνθήκες έντονου βάρους και εξωτερικών καιρικών συνθηκών (βροχή, υγρασία, χαλάζι κτλ.) Επίσης χρησιμοποιούνται ηλεκτρομηχανολογικές ή υδραυλικές διατάξεις για την κίνηση στους αντίστοιχους άξονες (μονού ή διπλού άξονα). Οι διατάξεις αυτές εξασφαλίζουν την κίνηση χωρίς προβλήματα και με αξιοπιστία. Η επιλογή της διάταξης εξαρτάται από τις προϋποθέσεις που έχουν τεθεί για το φωτοβολταϊκό σύστημα. Βασικές προϋποθέσεις για ιχνηλάτες σε συνήθη φωτοβολταϊκά συστήματα που χρησιμοποιούνται για παραγωγή ηλεκτρικής ενέργειας είναι η οικονομία στην κατανάλωση ενέργειας και η μειωμένες ανάγκες για συντήρηση. Σε ηλιακούς ιχνηλάτες που χρησιμοποιούνται για ερευνητικούς σκοπούς, σημαντικός παράγοντας είναι η ακρίβεια.

Ακόμη για την λειτουργία των ιχνηλατών χωρίς συνεχή εποπτεία χρησιμοποιούνται ηλεκτρονικές πλακέτες ή συστήματα αυτοματισμού (plc) στα οποία είναι προεγκατεστημένοι οι αστρονομικοί αλγόριθμοι καθώς και άλλες κύριες και βοηθητικές λειτουργίες.

Οι αστρονομικοί αλγόριθμοι, οι οποίοι αποτελούν το βασικό κορμό της διπλωματικής εργασίας και αναλύονται στα κεφάλαια ... , είναι μαθηματικά μοντέλα με την βοήθεια των οποίων υπολογίζουμε με ακρίβεια τη θέση του ήλιου.

Οι ηλεκτρονικές πλακέτες ή τα συστήματα αυτοματισμού (plc) χρησιμοποιούνται ώστε να "τρέχουν" συνεχώς τους αστρονομικούς αλγορίθμους και να υπολογίζουν την θέση στην οποία πρέπει να βρίσκεται ο ιχνηλάτης. Κατόπιν δίνουν εντολή στις μηχανολογικές ή υδραυλικές διατάξεις για να προσανατολιστεί ο ιχνηλάτης στην αντίστοιχη θέση. Οι κύριες λειτουργίες αφορούν τον προσανατολισμό μέσω των διατάξεων κίνησης του ιχνηλάτη.

Οι βοηθητικές λειτουργίες είναι ο έλεγχος της εύρυθμης λειτουργίας του ιχνηλάτη αλλά και των συνθηκών στις οποίες λειτουργεί ώστε να προστατευθεί από φθορά ή καταστροφή. Λόγω της έκθεσης τους στις εξωτερικές συνθήκες, τα φωτοβολταϊκά συστήματα με ιχνηλάτες, είναι εξοπλισμένα με μικρούς μετεωρολογικούς σταθμούς ώστε να παρακολουθούν τους ανέμους, έντονες βροχοπτώσεις ή χαλαζόπτωση. Έτσι λοιπόν στις βοηθητικές λειτουργίες περιλαμβάνεται η αξιοποίηση των δεδομένων καιρού για να προσανατολίζεται ο ιχνηλάτης ανάλογα με την ένταση του ανέμου ή της χαλαζόπτωσης ώστε να προστατεύεται από φθορά ή καταστροφή. Ακόμη μέσω του σταθμού ελέγχεται η ηλιοφάνεια για να διαπιστωθεί αν το φωτοβολταϊκό σύστημα αποδίδει βάση των προδιαγραφών του στις συγκεκριμένες συνθήκες και να διαπιστωθούν πιθανές δυσλειτουργίες ώστε να διορθωθούν. Μια ακόμα βοηθητική λειτουργία είναι η μετακίνηση σε θέση καθαρισμού (κάθετη) για την απομάκρυνση της ρύπανσης που κάθετα στα πάνελ και μειώνει την απόδοσή τους.

Με τη χρήση των tracker βελτιώνεται η απόδοση της φωτοβολταϊκής εγκατάστασης παραγωγής ηλεκτρικής ενέργειας. Η βελτίωση εξαρτάται από την ποιότητα των υλικών του φωτοβολταϊκού συστήματος αλλά και την σωστή υλοποίησή του (πχ αποφυγή σκιάσεων τόσο μόνιμων όσο και πρόσκαιρων κατά τη διάρκεια της ημέρας). Η βελτίωση της απόδοσης κυμαίνεται από 20-25% στους ιχνηλάτες (tracker) ενός άξονα έως και 35-40% σε αυτούς δύο αξόνων, στην παραγωγή ηλεκτρικής ενέργειας, ανάλογα με την εποχή, τον τύπο, την αξιοπιστία και την ποιότητα του συστήματος, όπως προαναφέρθηκε. Τέλος καθοριστικός παράγοντας είναι η εξέλιξη της τεχνολογίας που χρησιμοποιείται.

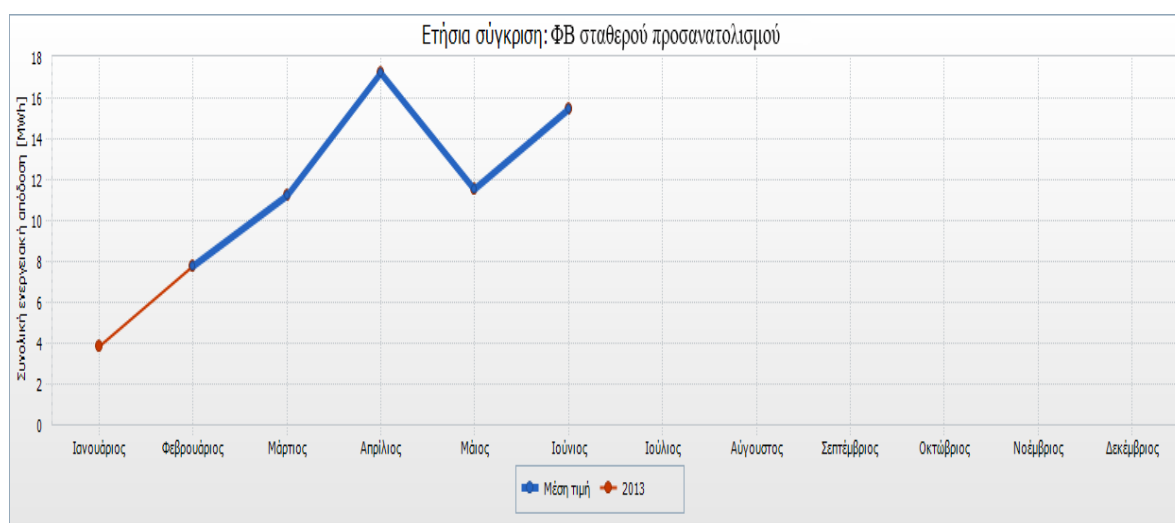


Παραδείγματος χάριν συγκρίνουμε δυο φωτοβολταϊκά συστήματα στην ίδια περιοχή. Το πρώτο ΦΒ σύστημα αποτελείται από ΦΒ πλαίσια σταθερού προσανατολισμού, ενώ το δεύτερο αποτελείται από ΦΒ πλαίσια τοποθετημένα σε ιχνηλάτη διπλού άξονα (βλέπε παρακάτω).

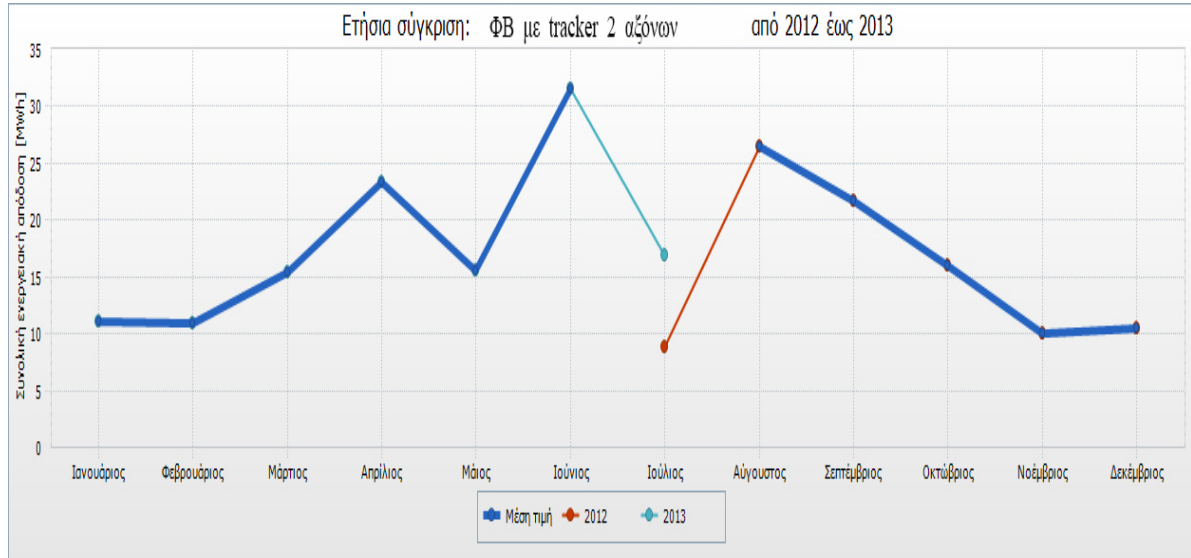
Τα ΦΒ συστήματα βρίσκονται στην περιοχή Λούτσι της Βοιωτίας. Παρατίθενται συγκριτικά στοιχεία για τον μήνα Απρίλιο, για μια τυχαία μέρα με αρκετή ηλιοφάνεια και για τους μήνες Φεβρουάριο μέχρι Μάιο. Εξετάστηκαν αυτοί οι μήνες, διότι το ΦΒ πάρκο με σταθερά πλαίσια ξεκίνησε να λειτουργεί από τις 16/1/2013. Επίσης δεν συμπεριλάβαμε τον μήνα Ιούνιο, παρότι φαίνεται στο διάγραμμα διότι δεν ήταν επαρκεί τα στοιχεία που είχαμε στη διάθεση μας.

ΦΒ ΠΑΡΚΟ	ΕΓΚΑΤΕΣΤΗΜΕΝ Η ΙΣΧΥΣ (KWp)	ΦΕΒΡΟΥΑΡΙΟΣ MWh	ΜΑΡΤΙΟΣ MWh	ΑΠΡΙΛΙΟΣ MWh	ΜΑΙΟΣ MWh
ΣΤΑΘΕΡΟΥ ΠΡΟΣΑΝΑΤΟΛΙΣΜΟΥ	108	7,79	11,27	17,27	11,56
ΠΡΟΣΑΝΑΤΟΛΙΣΜΟΥ 2 ΑΞΟΝΩΝ	107,16	10,89	15,43	23,41	15,61

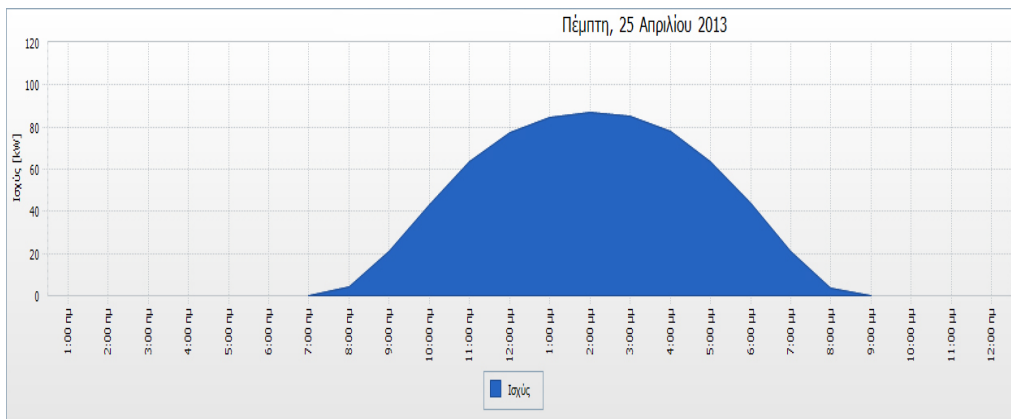
**ΠΙΝΑΚΑΣ 4.1 ΠΑΡΑΓΩΓΗ ΗΛΕΚΤΡΙΚΗΣ ΕΝΕΡΓΕΙΑΣ ΣΕ MWh**



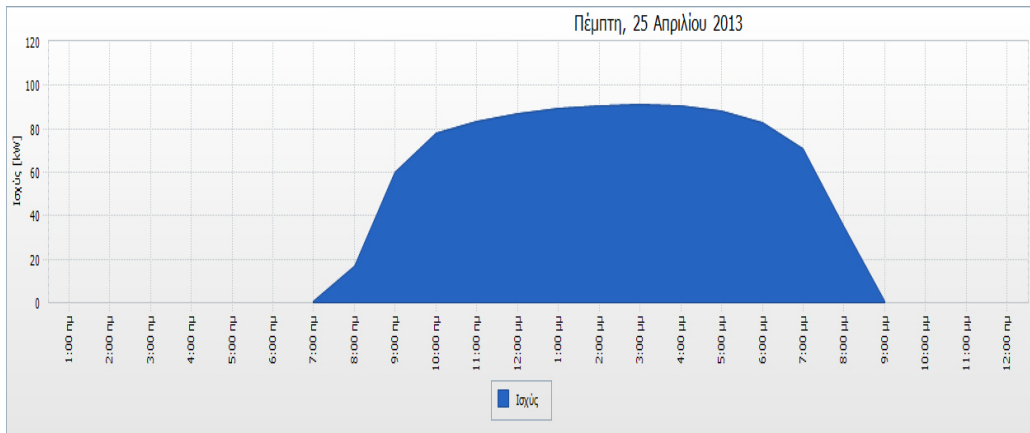
**Σχήμα 4.1** Παραγωγή για ΦΒ πάρκο σταθερού προσανατολισμού



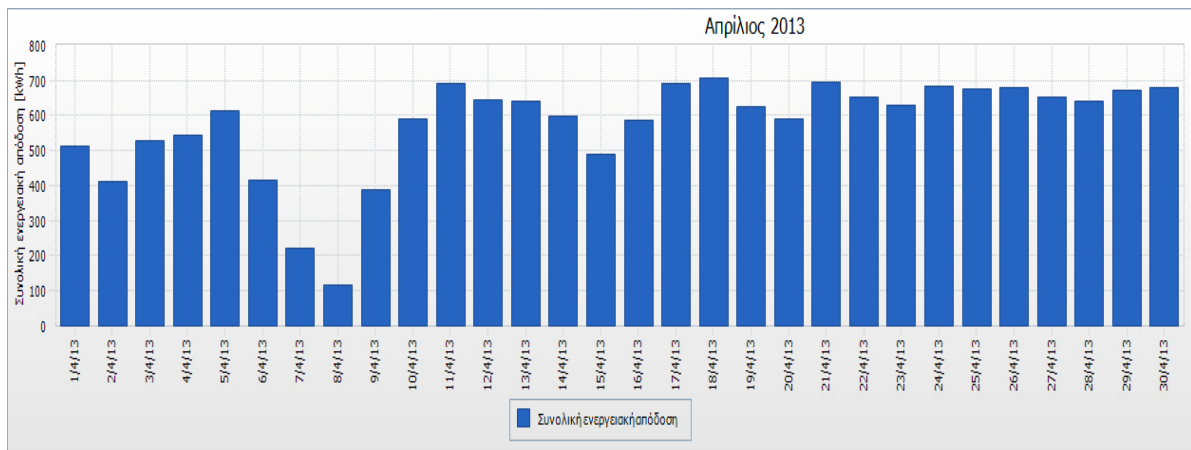
**Σχήμα 4.2** Παραγωγή για ΦΒ με tracker



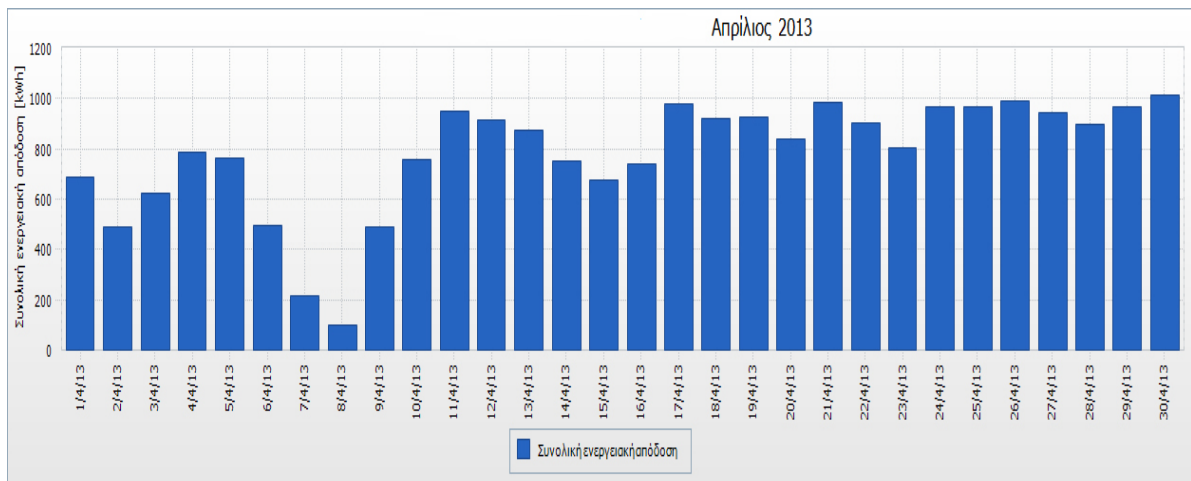
**Σχήμα 4.3** Ημερήσια παραγωγή ΦΒ σταθερού προσανατολισμού



Σχήμα 4.4 Ημερήσια παραγωγή ΦΒ με tracker



Σχήμα 4.5 Μηνιαία παραγωγή ΦΒ σταθερού προσανατολισμού



#### Σχήμα 4.6 Μηνιαία παραγωγή ΦΒ με tracker

Όπως παρατηρούμε και από τα διαγράμματα είναι ξεκάθαρη η διαφορά στην ποσότητα παραγόμενης ηλεκτρικής ενέργειας, τόσο σε καθημερινό επίπεδο, όσο και σε μηνιαίο.

Για παράδειγμα από τον πίνακα 4.1 συγκρίνοντας τις τιμές της παραγόμενης ηλεκτρικής ενέργειας μπορούμε να καταλήξουμε στον παρακάτω πίνακα 4.2 για την διαφορά απόδοσης του ΦΒ με tracker σε σχέση με το ΦΒ σταθερών πλαισίων.

	ΦΕΒΡΟΥΑΡΙΟΣ	ΜΑΡΤΙΟΣ	ΑΠΡΙΛΙΟΣ	ΜΑΙΟΣ
ΔΙΑΦΟΡΑ ΣΕ ΑΠΟΛΥΤΗ ΤΙΜΗ (MWh)	3,1	4,16	6,14	4,05
ΠΟΣΟΣΤΙΑΙΑ ΔΙΑΦΟΡΑ (%)	39,8	36,91	35,55	35,04

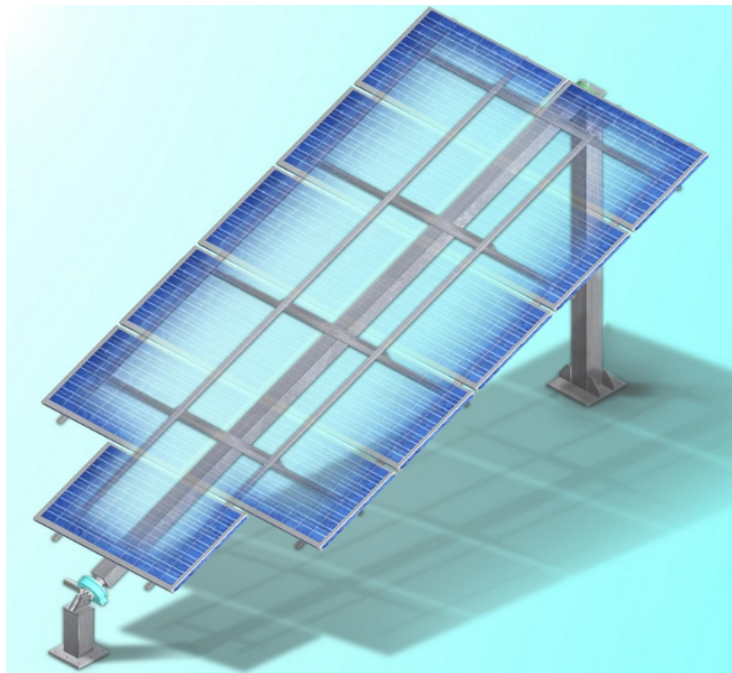
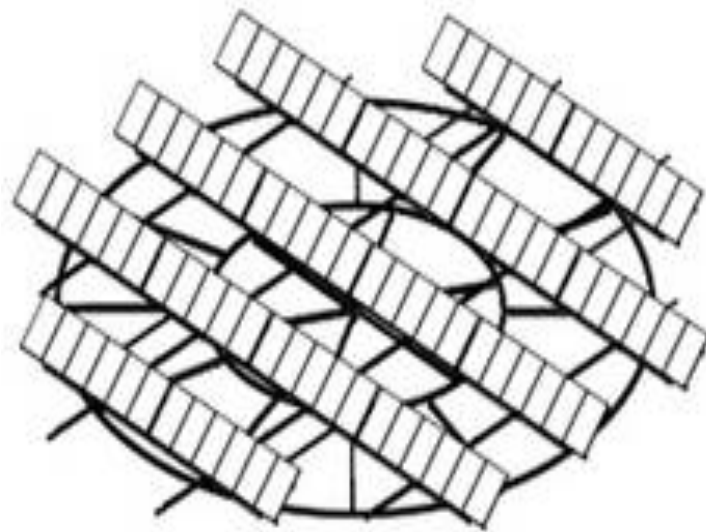
Πίνακας 3.2

Παρατηρούμε πως για τους τέσσερις τυχαίους μήνες που εξετάσαμε, η απόδοση του ΦΒ πάρκου με εγκατεστημένους tracker για την παρακολούθηση της κίνησης του Ήλιου είναι πιο αποδοτικά από 35% μέχρι 40% σε σχέση με το ΦΒ πάρκο με πανέλα σταθερού προσανατολισμού. Αξίζει να σημειώσουμε πως τον μήνα με τη μεγαλύτερη μέση ηλιοφάνεια όπου και τα δύο ΦΒ πάρκα είχαν τη μέγιστη απόδοση τους στο εξεταζόμενο διάστημα, η διαφορά της απόδοσης μειώνεται στο ελάχιστο αυτού του διαστήματος που εξετάζουμε. Αντίθετα το χειμώνα η διαφορά της απόδοσης αυξάνεται στο μέγιστο.

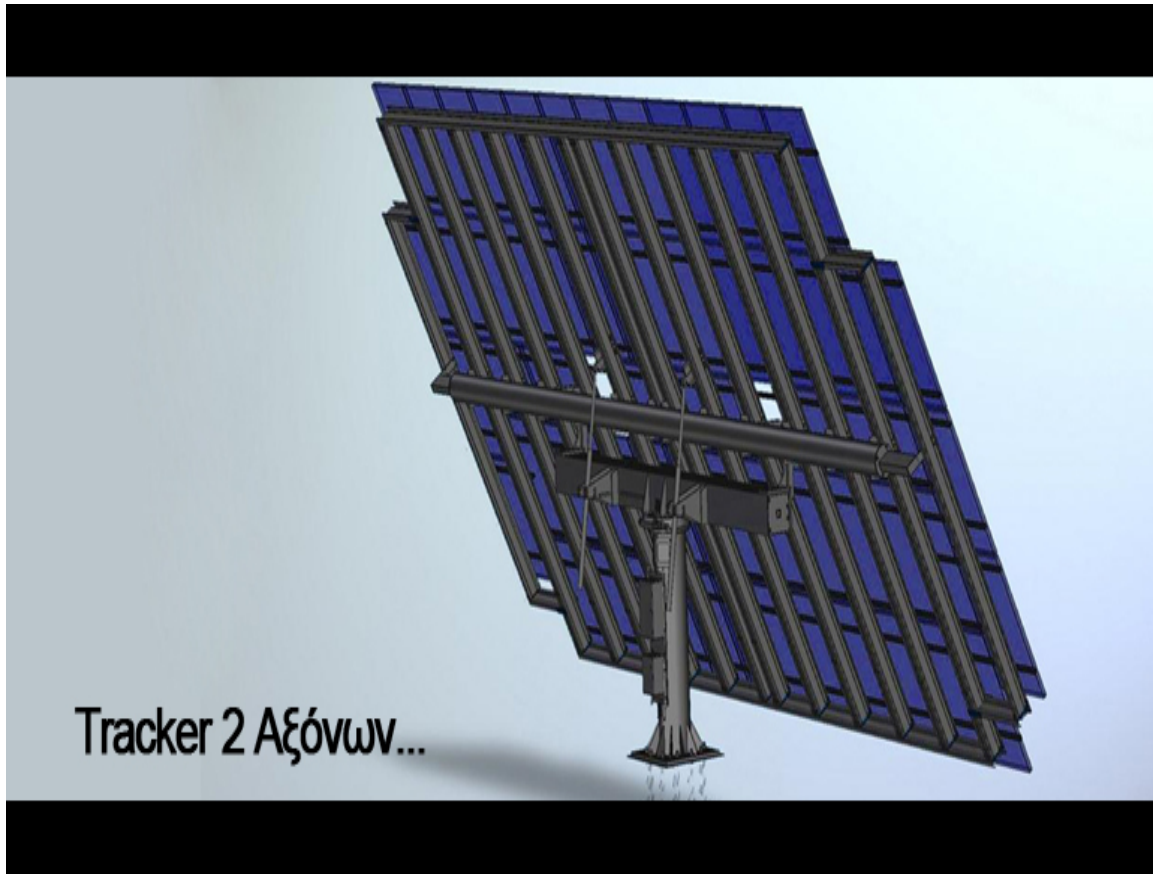
Τα παραπάνω βέβαια αναφέρονται ως συμπέρασμα το οποίο προκύπτει από εγκατεστημένα πάρκα στην ίδια ακριβώς περιοχή, με διαφορετικά ΦΒ πανέλα εγκατεστημένα, μια μικρή διαφορά στη μέγιστη ισχύς εγκατάστασης και δεν μπορούν να αποτελέσουν επιστημονική απόδειξη της διαφοράς απόδοσης αφού δεν πραγματοποιήθηκαν στις ίδιες ακριβώς συνθήκες, με ακριβής ελέγχους από εξειδικευμένο προσωπικό και εξοπλισμό.

Ο φωτοβολταϊκός ιχνηλάτης μπορεί να διακριθεί στις εξής κατηγορίες:

- **Ιχνηλάτης ενός άξονα** ο οποίος παρακολουθεί την ημερήσια (αζιμουθιακή) κίνηση του ήλιου σε σταθερή κλίση, έχοντας όμως την ανάλογη κλίση ανάλογα με την θέση του ήλιου σε κάθε εποχή. Ο προσδιορισμός της κλίσης αυτής ανάλογα με την εποχή γίνεται είτε χειροκίνητα είτε με μηχανισμό.



- **Ιχνηλάτης δυο αξόνων** οποίος μεταβάλλει όχι μόνο την αζιμουθιακή αλλά και τη γωνία ανύψωσης, παρακολουθώντας με ακρίβεια την κίνηση του ήλιου και πραγματοποιώντας το καλύτερο αποτέλεσμα δέσμευσης ηλιακών ακτινών.



### 4.3 ΦΩΤΟΒΟΛΤΑΪΚΟΙ ΙΧΝΗΛΑΤΕΣ ΑΖΙΜΟΥΘΙΟΥ

Αντικείμενο της διπλωματικής εργασίας είναι η κατασκευή φωτοβολταϊκού ιχνηλάτη αζιμουθιακής γωνίας.

Όπως αναφέρεται ήδη πρόκειται για ιχνηλάτη ενός άξονα ο οποίος παρακολουθεί την ημερήσια (αζιμουθιακή) κίνηση του ήλιου σε σταθερή κλίση, έχοντας όμως την ανάλογη κλίση ανάλογα με την θέση του ήλιου σε κάθε εποχή.

Έχουν πιο απλές διατάξεις από τους ιχνηλάτες δυο αξόνων και είναι πιο οικονομικοί στην κατασκευή και την λειτουργία.

Η απόδοση τους είναι μικρότερη από τους ιχνηλάτες δυο αξόνων, αλλά μεγαλύτερη από τα φωτοβολταϊκά συστήματα που χρησιμοποιούν σταθερά πλαίσια. Η οικονομία στην κατασκευή και την συντήρηση τους καθιστά ελκυστικούς σε σχέση με τους ιχνηλάτες 2 αξόνων.





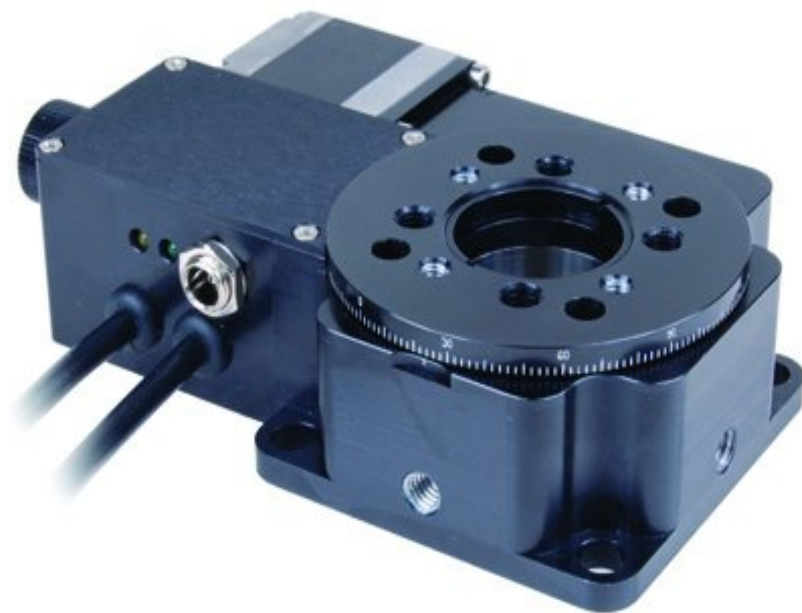
## Προσομοίωση του ηλιακού ιχνηλάτη

Στο κεφάλαιο αυτό παρουσιάζονται οι διατάξεις που χρησιμοποιήθηκαν για την προσομοίωση του ηλιακού ιχνηλάτη αλλά και η διαδικασία για την υλοποίηση του προγράμματος αξιοποιώντας την ηλιακή γεωμετρία και τους αστρονομικούς αλγορίθμους.



## 5.1 ZABER T-RS 60

Για την προσομοίωση του συστήματος ιχνηλάτησης της αζιμουθιακής γωνίας του ήλιου (azimuth tracker solar) χρησιμοποιήσαμε τον, μηχανοποιημένο περιστροφικό ρότορα με ενσωματωμένους ελεγκτές, T-RS 60 από τη σειρά T-RS της Zaber.



Ο συγκεκριμένος ρότορας έχει την δυνατότητα να περιστρέφεται με αρκετά μεγάλη ακρίβεια στη βάση των εντολών που δέχεται από την κονσόλα ελέγχου. Ως κονσόλα ελέγχου χρησιμοποιείται ένας ηλεκτρονικός υπολογιστής, απ' όπου μέσω διαφόρων κατάλληλων προγραμμάτων δίνονται οι εντολές για τον έλεγχο του ρότορα.

Ο ρότορας έχει δυνατότητα περιστροφής κατά  $360^{\circ}$  και μπορεί να κάνει μέχρι και 8 στροφές ανά λεπτό. Λειτουργεί με μικροβήματα με αντιστοίχιση μοίρας – μικροβήματος.

Για την υλοποίηση της διπλωματικής εργασίας χρησιμοποιήσαμε την αντιστοίχιση  $1^{\circ} = 4267$  micro steps.

Λόγο αδυναμίας του ρότορα να κινηθεί σε αρνητικές γωνίες ως προς την γωνία βάσης 0, μεταφέρουμε την γωνία βάσης στις 180 μοίρες. Η γωνία βάσης αποτελεί τον προσανατολισμό ως προς νότο απ' όπου υπολογίζεται η αζιμούθια γωνία.

Από την εταιρεία κατασκευής δίνονται διάφορες κονσόλες και έτοιμα προγράμματα για τον έλεγχο του ρότορα και τη δημιουργία του κατάλληλου προγράμματος για το αντικείμενο που μας ενδιαφέρει.

Στα πλαίσια της διπλωματικής επιλέχθηκε η υλοποίηση από μηδενική βάση με τη χρήση της γλώσσας προγραμματισμού C# (C sharp) για την ανάπτυξη ενός ολοκληρωμένου προγράμματος που θα λαμβάνει υπόψη του τους αστρονομικούς αλγορίθμους για την παρακολούθηση του Ήλιου από έναν ιχνηλάτη (solar tracker).

## **5.2 Προγραμματισμός του ηλιακού ιχνηλάτη**

Για την προσομοίωση του ηλιακού ιχνηλάτη υλοποιήσαμε πρόγραμμα μέσω γλώσσας προγραμματισμού, χρησιμοποιώντας τους αστρονομικούς αλγορίθμους για την ηλιακή γεωμετρία, όπως αναλύθηκαν στο κεφάλαιο II.

### **5.2.1 Η γλώσσα προγραμματισμού**

Ο έλεγχος της συσκευής μπορεί να πραγματοποιηθεί με πολλούς τρόπους. Η ίδια η εταιρεία έχει δημιουργήσει μια κονσόλα όπου υπάρχει η δυνατότητα ελέγχου μέσω έτοιμων προκαθορισμένων εντολών ή η δυνατότητα μέσω υποτυπώδους περιβάλλοντος προγραμματισμού να δημιουργηθούν σύντομα προγράμματα στις εξής γλώσσες προγραμματισμού: C#, javascript ή visual basic. Επίσης δίνεται η δυνατότητα ελέγχου μέσω labview.

Με κριτήριο την αξιοποίηση των αποτελεσμάτων της διπλωματικής εργασίας και σε άλλα συστήματα ιχνηλάτησης του ήλιου, πέραν της συγκεκριμένης συσκευής, επιλέχθηκε η κατασκευή προγράμματος μέσω γλώσσας προγραμματισμού.

Για τον παραπάνω αλλά και για τους επιπλέον λόγους που αναφέρονται και στο κεφάλαιο «Συμπεράσματα και προοπτική της διπλωματικής» επιλέχθηκε η κατασκευή του προγράμματος να γίνει μέσω γλώσσας προγραμματισμού εκ του μηδενός. Συγκεκριμένα επιλέχθηκε ως γλώσσα προγραμματισμού η C#.

## 5.2.2 Η υλοποίηση του προγράμματος ελέγχου του ηλιακού ιχνηλάτη

Η κατασκευή του προγράμματος για τον έλεγχο του αζιμουθιακού ιχνηλάτη αποτελείται από δυο βασικά στοιχεία:

1. Τον υπολογισμό των δεδομένων, ώστε να καταλήξουμε στην αζιμουθιακή γωνία που χρειαζόμαστε για τον προσανατολισμό του tracker
2. Στην αναγνώριση και επικοινωνία του υπολογιστή με την συσκευή προσομοίωσης του ιχνηλάτη μέσω θύρας usb.

### 5.2.2.1 Υπολογισμός των δεδομένων

Ο υπολογισμός των δεδομένων βασίζεται στους αστρονομικούς αλγορίθμους, που περιγράψαμε στο δεύτερο κεφάλαιο. Μέσω των αστρονομικών αλγορίθμων υπολογίζουμε τη θέση του ήλιου, ώστε να προσανατολίσουμε τον ηλιακό ιχνηλάτη (solar tracker) ως προς νότο (αζιμούθια γωνία).

Ο υπολογισμός της θέσης του ήλιου στον αζιμούθιο άξονα απαιτεί των υπολογισμό αρκετών παραμέτρων, όπως φαίνεται και στην παράγραφο 2.2.3 όπου περιγράφονται οι αστρονομικοί αλγόριθμοι.

Για να υπολογίσει το πρόγραμμα μας την αζιμούθια γωνία, εκτελεί πρώτα μια ακολουθία υπολογισμών.

Το πρόγραμμα είναι υλοποιημένο με βέλτιστο αλγόριθμο ώστε να μην επαναλαμβάνει υπολογισμούς οι οποίοι δεν χρειάζονται. Έτσι το πρόγραμμα «τρέχει» τέσσερις συναρτήσεις με τις οποίες υπολογίζει τα δεδομένα που χρειάζεται.

Πριν από οποιονδήποτε υπολογισμό το πρόγραμμα πρέπει να γνωρίζει σε ποιά γεωγραφική περιοχή βρίσκεται, ώστε να υπολογιστούν οι αντίστοιχοι παράμετροι για την συγκεκριμένη περιοχή. Δηλαδή πρέπει να είναι γνωστά το γεωγραφικό μήκος και το γεωγραφικό πλάτος.

Η πρώτη συνάρτηση που τρέχει το πρόγραμμα είναι η συνάρτηση readGPS(), με την οποία διαβάζονται τα γεωγραφικά δεδομένα της περιοχής λειτουργίας του tracker.

Επομένως ζητάμε από τον διαχειριστή του tracker να μας δώσει τα παρακάτω στοιχεία της γεωγραφικής περιοχής:

Πρώτα έχουμε ορίσει το σύστημα να ενημερώνει το διαχειριστή του tracker πως τα γεωγραφικά στοιχεία πρέπει να δίνονται με κόμμα και όχι με τελεία

```
Console.WriteLine("Οι δεκαδικοί αριθμοί δίνονται με κόμμα");
```

Οι εντολές `Console.WriteLine` είναι εντολές εκτύπωσης του περιεχομένου των παρενθέσεων, είτε αυτούσιο όταν περιέχεται σε εισαγωγικά, είτε το περιεχόμενο των μεταβλητών που αναφέρονται μέσα στις παρενθέσεις.

Έπειτα ορίζουμε το σύστημα να ζητάει το γεωγραφικό μήκος της περιοχής

```
Console.WriteLine("Δώσε γεωγραφικό μήκος");
```

και να το αποθηκεύει στην μεταβλητή τύπου string input1

```
string input1 = Console.ReadLine();
```

Ομοίως ζητάει το γεωγραφικό πλάτος της περιοχής

```
Console.WriteLine("Δώσε γεωγραφικό πλάτος");
```

και να το αποθηκεύει στην μεταβλητή τύπου string input2

```
string input2 = Console.ReadLine();
```

Ακολούθως ζητάει τον μεσημβρινό αναφοράς και τη διαφορά ζώνης ώρας. Τα δύο αυτά στοιχεία αποθηκεύονται στις μεταβλητές τύπου string, αντίστοιχα input3 & input4, και τις χρειαζόμαστε για τη μετατροπή της ώρας του συστήματος, αποκλειστικά σε χειμερινή ζώνη ώρας, όπου είναι υπολογισμένα όλα τα αποτελέσματα από τους αστρονομικούς αλγορίθμους.

```
Console.WriteLine("Δώσε μεσημβρινό αναφοράς");
```

```
string input3 = Console.ReadLine();
```

```
Console.WriteLine("Δώσε διαφορά ζώνης ώρας");
```

```
string input4 = Console.ReadLine();
```

Στη συνέχεια αποθηκεύουμε τις τέσσερις αυτές μεταβλητές, στις σταθερές με τα παρακάτω ονόματα, κατ' αντιστοιχία, διότι καθ' όλη τη διάρκεια της εκτέλεσης τα γεωγραφικά χαρακτηριστικά της περιοχής δεν αλλάζουν.

```
geomik = Double.Parse(input1);
```

```
geopla = Double.Parse(input2);
```

```
Lst = Double.Parse(input3);
```

```
ΔTgmt = Double.Parse(input4);
```

Αφού έχουμε εισάγει τα γεωγραφικά χαρακτηριστικά της περιοχής, ολοκληρώνεται η συνάρτηση readGPS().

Επειδή θέλουμε τα δεδομένα περιστροφής για την παρακολούθηση του ήλιου, να έχουμε την δυνατότητα να τα επεξεργαστούμε, ορίζουμε το αρχείο όπου θα αποθηκεύεται η ημέρα, η ώρα με ακρίβεια δευτερολέπτου, η αζιμούθια γωνία καθώς και η γωνία β. Για να είναι εύκολη η αξιοποίηση και επεξεργασία των αποτελεσμάτων αυτών, τα στοιχεία αποθηκεύονται σε μορφή .csv.

```
StreamWriter file = new StreamWriter(@"C:\Users\reitzo\Desktop\temp\" +  
DateTime.Now.DayOfYear.ToString() + ".csv", true);
```

Η αποθήκευση των στοιχείων αυτών γίνεται μετά την ολοκλήρωση τόσο των υπολογισμών τους, όσο και της κίνησης του ιχνηλάτη, με τις εντολές:

```
file.WriteLine(imerax_xronou + ";" + wra_twra + ";" + min_twra + ";"  
+ sec_twra + ";" + b_maires + ";" + (-Z_maires) + ";");
```

Επειδή θέλουμε το πρόγραμμα μας να εκτελείτε ατέρμονομα, εκτός της περίπτωσης που το απενεργοποιήσουμε εμείς, ξεκινάμε τη διαδικασία υπολογισμού των δεδομένων με:

```
while (true)
```

Στη συνέχεια υπολογίζουμε την ημέρα του χρόνου σε απόλυτο αριθμό (από το 1 έως το 365) και την αποθηκεύουμε στην μεταβλητή `imerax_xronou`, ως ακέραιο αριθμό.

```
{
```

```
    //Υπολογισμός της ημέρας του χρόνου
```

```

Console.WriteLine(ΔTgmt);
double[] stoixeia = readParams();
string yo;
DateTime dy = DateTime.Now;
Console.WriteLine("Date is " +
dy.DayOfYear.ToString());
yo = dy.ToString("yyyy");

//αποθήκευση ημέρας του χρόνου σε μεταβλητή int

Int32.Parse(dy.DayOfYear.ToString());
int imera_xronou;

imera_xronou = Int32.Parse(dy.DayOfYear.ToString());

```

Έπειτα καλείται η τρίτη συνάρτηση, για τον υπολογισμό της τρέχουσας ώρας. Η συνάρτηση αυτή αναγνωρίζει την τρέχουσα ώρα από τον υπολογιστή στον οποίο τρέχει το πρόγραμμα και ελέγχει αν είναι χειμερινή ή εαρινή. Στην περίπτωση που είναι εαρινή την μετατρέπει σε χειμερινή, αφού όπως αναφέρεται στην παράγραφο 2.2.4.1 χρησιμοποιείται για όλους τους υπολογισμούς η χειμερινή ώρα. Στη συνέχεια αποθηκεύει σε ακέραιες μεταβλητές, ξεχωριστά την ώρα, τα λεπτά και τα δευτερόλεπτα ώστε να είναι εύκολα επεξεργάσιμη, υποθηκεύσιμη και συγκρίσιμη η ώρα.

```
//Υπολογισμός της ώρας τώρα
```

```

int[] wras = ReadTime((int)stoixeia[7]);
int wras_twra = wras[0];
int mins_twra = wras[1];
int secs_twra = wras[2];

```

Στη συνέχεια μετατρέπουμε την ώρα ανατολής που έχει υπολογιστεί από τη δεύτερη συνάρτηση, σε μορφή η οποία μας βοηθάει να την συγκρίνουμε απευθείας με την τρέχουσα ώρα.



```
//Μετατρέπουμε την ώρα ανατολής και δύσης σε μορφή, η  
    οποία μας βοηθάει στους ελέγχους των βρόχων
```

```
DateTime dateAnatolis = new  
DateTime(DateTime.Today.Year, DateTime.Today.Month,  
DateTime.Today.Day, (int) stoixeia[0], (int)  
stoixeia[1], (int) stoixeia[2]);  
Console.WriteLine("Date Anatolis: " + dateAna-  
tolis.ToString());  
DateTime dateDysis = new DateTime(DateTime.Today.Year,  
DateTime.Today.Month, DateTime.Today.Day,  
(int)stoixeia[3], (int)stoixeia[4], (int)stoixeia[5]);  
Console.WriteLine("Datetime Dysis: " + dateDy-  
sis.ToString());  
Console.WriteLine("Datetime Now: " + DateTime.Now);  
Console.WriteLine("Ticks: " + DateTime.Now.Ticks);
```

Επειδή το πρόγραμμα θα τρέχει ατέρμονα, χρειάζεται να υπολογίζεται αν την τρέχουσα ώρα είναι ημέρα ή νύχτα. Αυτό επιτυγχάνεται με έναν βρόχο ελέγχου της ώρας. Ουσιαστικά ελέγχεται αν η τρέχουσα ώρα έχει ξεπεράσει την ώρα ανατολής. Στην περίπτωση που έχει ξεπεραστεί η ώρα ανατολής (δηλαδή με το που ανατέλλει ο Ήλιος) ο αλγόριθμος προχωράει στην υλοποίηση της επόμενης εντολής. Όμως στην περίπτωση που η τρέχουσα ώρα δεν έχει ξεπεράσει την ώρα ανατολής, υπολογίζεται το χρονικό διάστημα, σε millisecond, που απαιτείται μέχρι την ανατολή του Ηλίου. Για αυτό το διάστημα ορίζεται ο tracker σε sleep mode (κατάσταση "ύπνου").

```
// Συγκρίνουμε την ώρα τώρα με την ώρα ανατολής
```

```
while (dateAnatolis > DateTime.Now)  
{  
  
    //Υπολογισμός της χρονικής διάρκειας που  
    απομένει μέχρι την ανατολή του ηλίου  
  
    //για αυτή την χρονική διάρκεια τοποθετείται το  
    τρακερακι σε κατάσταση ύπνου  
  
    //αυτό το γίνεται για λόγους οικονομίας στη  
    λειτουργία του επεξεργαστή και κατανάλωσης  
    ενέργειας  
  
    TimeSpan ypnos = new  
    TimeSpan(dateAnatolis.Ticks -
```

```

DateTime.Now.Ticks);
Console.WriteLine("sleep for: " + yp-
nos.TotalMilliseconds);
Thread.Sleep(ypnos);
}

```

Όταν η τρέχουσα ώρα φτάσει στην ώρα ανατολής, το πρόγραμμα φεύγει από τον παραπάνω βρόχο και εκτελεί το ακόλουθο βρόχο. Ο βρόχος αυτός εκτελείται για το χρονικό διάστημα από την ανατολή του Ηλίου, μέχρι τη δύση του.

```

while (dateDysis > DateTime.Now)
{
...
}

```

Μέσα σε αυτόν το βρόχο εκτελείται το βασικό κομμάτι του προγράμματος, ο υπολογισμός της αζιμουθιακής γωνίας χρησιμοποιώντας επί μέρους υπολογισμούς μέσω των αστρονομικών αλγορίθμων.

Αρχικά υπολογίζονται η τρέχουσα ώρα, η τοπική ώρα, η γωνία ημέρας B, η εξίσωση χρόνου E και η ηλιακή ώρα TSOLAR ώστε να υπολογιστεί η γωνία ώρας ω. Η τρέχουσα ώρα υπολογίζεται καλώντας τη συνάρτηση, όπως είδαμε νωρίτερα, ReadTime, την οποία μετατρέποντας σε δεκαδική μορφή παίρνουμε την Tlocal. Γνωρίζοντας την ημέρα του χρόνου υπολογίζεται η γωνία ημέρας B και στη συνέχεια εξίσωση χρόνου E. Με γνωστή την εξίσωση χρόνου και τη Tlocal υπολογίζεται η ηλιακή ώρα TSOLAR και στη συνέχεια η γωνία ώρας ω.

```
//Υπολογισμός της ώρας τώρα
```

```

int[] wra = ReadTime((int)στοιχεία[7]);
int wra_twra = wra[0];
int min_twra = wra[1];
int sec_twra = wra[2];

```

```
//Μετατροπή τοπικής ώρας σε μορφή δεκαδικού
```

```

double Tlocal = sec_twra / 60;
Tlocal = (min_twra + Tlocal) / 60;

```

```

Tlocal = wra_twra + Tlocal;
Console.WriteLine(Tlocal + " Tlocal ");

//υπολογισμός της γωνιάς ημέρας B

double B_maires;
double B_rad;
double b_maires;
double Z_maires;
B_maires = 360 * (imeraxronou - 81) / 365;
B_rad = B_maires * 3.14159265 / 180;

//υπολογισμός της εξίσωσης χρόνου E

double e_xronou;
e_xronou = 9.87 * Math.Sin(2 * B_rad) - 7.53 *
Math.Cos(B_rad) - 1.5 * Math.Sin(B_rad);

//Υπολογισμός Tsolar

double Tsolar = Tlocal + (e_xronou / 60) + ((4
* (stoixeia[6] - stoixeia[9]))/60);

//Υπολογισμός γωνίας ω

double ω_maires;
ω_maires = 15 * (Tsolar - 12);
double ω_rad = (ω_maires * 3.14159265) / 180;

```

Όπως αναφέρεται στην παράγραφο 2.2.4.2: Η ωριαία γωνία,  $\omega$ , λαμβάνει αρνητικές τιμές ανατολικά του μεσημβρινού που εμπεριέχει το ζενίθ (πρωί) και θετικές τιμές δυτικά του (απόγευμα). Γι' αυτό το λόγο εξετάζουμε τις περιπτώσεις όταν η γωνία ώρας είναι μεγαλύτερη του μηδενός και μικρότερη, όπως φαίνεται και στον κώδικα, για τον υπολογισμό των υπολοίπων δεδομένων.

```

if (ω_maires > 0)
{
...

```

```

        Z_maires = Z_rad * 180 / 3.14159265;
    }

else

{
    ...

    Z_maires = - Z_rad * 180 / 3.14159265;
}

```

Αφού εξετάσουμε το πρόσημο της γωνίας ώρας  $\omega$ , διότι σχετίζεται άμεσα με τον πρόσημο της αζιμουθιακής γωνίας ως προς νότο, προχωράμε στον υπολογισμό της γωνίας  $\delta$ , του ύψους του ήλιου  $\beta$  και της αζιμουθιακής γωνίας  $Z$ . Τα αποτελέσματα της γωνίας  $\delta$  και του ύψους του ήλιου  $\beta$  μετατρέπονται σε ακτίνια (rad), ενώ της αζιμουθιακής γωνίας  $Z$  μετατρέπονται σε μικροβήματα, «αναγνώσιμα», όπως έχει αναφερθεί στην παράγραφο 5.1, από το Zaber T-RS 60.

```

double gonia_d;
double sin_imeras;
sin_imeras = Math.Sin(((360 * (imera_xronou
+ 284)) / 365) * 3.14159265 / 180);

//υπολογισμός της γωνιάς δ

gonia_d = 23.45 * sin_imeras;

//μετατροπή της γωνιάς δ σε rad

double rad_gonia_d;
rad_gonia_d = (3.14159265 * gonia_d) / 180;

//Υπολογισμός ύψους ήλιου β

double sin_b = (Math.Sin(rad_gonia_d) *
Math.Sin(stoixeia[10] * 3.14159265 / 180))

```

```

+ (Math.Cos(rad_gonia_d) *
Math.Cos(stoixeia[10] * 3.14159265 / 180) *
Math.Cos(w_rad));
double b_rad = Math.Asin(sin_b);
b_maires = b_rad * 180 / 3.14159265;
Console.WriteLine(b_maires + " ύψος σε
μοίρες");

//Υπολογισμός αζιμούθιας γωνίας Z

double Z_rad = Math.Acos((sin_b *
Math.Sin(stoixeia[10] * 3.14159265 / 180) -
Math.Sin(rad_gonia_d)) / (Math.Cos(b_rad) *
Math.Cos(stoixeia[10] * 3.14159265 /
180)));
Z_maires = Z_rad * 180 / 3.14159265;

//Μετατροπή αζιμουθιακής γωνίας σε
"μικροβήματα" για συμβατότητα

Console.WriteLine(Z_maires + " αζιμουθιακή
σε μοίρες");
double az_gonia = (768000 / 180) *
Z_maires;
int azimouthia_gonia;
azimouthia_gonia = (int)az_gonia;
Console.WriteLine(az_gonia + " αζιμούθια
γωνία ως προς νότο");

```

Στη συνέχεια βρίσκουμε τη θέση στην οποία πρέπει να βρίσκεται η συσκευή Zaber T-RS 60 ώστε να βρίσκεται σε γωνία ως προς νότο που να ταυτίζεται με την αζιμούθια γωνία που υπολογίστηκε.

```

//Μετακίνηση του tracker στην αζιμούθια
γωνία

int x = (768000 + azimouthia_gonia);

conversation.Request(Command.MoveAbsolute,

```

```
x);
```

```
Console.WriteLine(x + " απόλυτη θέση σε  
μικροβήματα");
```

Εδώ ολοκληρώνεται ο κύριος βρόχος υπολογισμού των δεδομένων που χρειάζονται για τον υπολογισμό της αζιμούθιας γωνίας. Ο βρόχος αυτός, όπως έχει ήδη αναφερθεί, εκτελείται συνεχώς από την ανατολή έως τη φύση του ηλίου. Για λόγους ενεργειακής οικονομίας, αφού δεν μας ενδιαφέρει ακρίβεια δευτερολέπτου, δίνουμε εντολή να ενεργοποιηθεί η λειτουργία ύπνου (sleep mode) για 2000 millisecond πριν προχωρήσει στην επανεκτέλεση του βρόχο.

```
Thread.Sleep(2000);
```

Δεδομένης της κίνησης του Ήλιου, τις ανάγκες για μειωμένη κατανάλωση ενέργειας αλλά και ελαχιστοποίησης της χρήσης ώστε να εξασφαλίζεται η φθορά των συστημάτων κίνησης του ιχνηλάτη, δεν επιθυμούμε την συνεχή εκτέλεση του βρόχου υπολογισμού της αζιμούθιας γωνίας αλλά και της αντίστοιχης κίνησης προσαρμογής του μηχανισμού. Για αυτό το λόγο δίνουμε την δυνατότητα στο πρόγραμμα να «παγώνει» τη λειτουργία για 2000 millisecond.

Όταν περάσει η ώρα δύσης, ως τελευταία λειτουργία αποθηκεύεται το αρχείο καταγραφής των δεδομένων που μας ενδιαφέρουν για επεξεργασία.

```
file.Flush();
```

Μετά την ώρα δύσης το πρόγραμμα δεν έχει κάποια λειτουργία να εκτελέσει μέχρι την ανατολή της επόμενης ημέρας. επομένως για να μην σπαταλάμε ενέργεια, αλλά και προστασίας των μηχανικών μερών του ιχνηλάτη από φθορά, τοποθετούμε τον ιχνηλάτη σε κατάσταση ύπνου. Μέσω αυτής της διαδικασίας ο ιχνηλάτης καθ' όλη τη διάρκεια της νύχτας παραμένει σταθερός και όταν υπολογιστεί η ώρα ανατολής, ξεκινάει ο λειτουργία του προγράμματος όπου υπολογίζεται η αζιμούθια γωνία και προσανατολίζεται ο ιχνηλάτης.

Έτσι λοιπόν υπολογίζουμε την ώρα ανατολής της επόμενης ημέρας και τοποθετούμε τον ιχνηλάτη σε κατάσταση ύπνου (sleep mode) μέχρι την ώρα ανατολής που υπολογίστηκε. Για λόγους οριακής ακρίβειας, δεδομένης της ακρίβειας που μας ενδιαφέρει, έχουμε προγραμματίσει τον ιχνηλάτη να ξεκινάει 1000 millisecond αργότερα από την υπολογισμένη ώρα ανατολής.

```

//Υπολογισμός της χρονικής διάρκειας μέχρι την αλλαγή
της ημέρας

//και τοποθέτηση του ιχνηλάτη σε κατάσταση ύπνου μέχρι
την αλλαγή της ημέρας

TimeSpan mesanyxta = new TimeSpan((new
DateTime(DateTime.Today.Year, DateTime.Today.Month,
DateTime.Today.Day + 1, 0, 0, 0)).Ticks -
DateTime.Now.Ticks);
Console.WriteLine("sleep for: " +
mesanyxta.TotalMilliseconds);
Thread.Sleep( (int) mesanyxta.TotalMilliseconds + 1000);

    }
}
}
}
}

```

### 5.2.2.2 Αναγνώριση και επικοινωνία του υπολογιστή με την συσκευή προσομοίωσης του ιχνηλάτη μέσω θύρας usb

Για την υλοποίηση του ιχνηλάτη με τη συσκευή Zaber T-RS 60 χρειάζεται η επικοινωνία με τη συσκευή εκτέλεσης του προγράμματος ( πχ laptop, pc, plc κτλ.). Στα πλαίσια της διπλωματικής, ο προγραμματισμός και η εκτέλεση του προγράμματος έγινε από φορητό υπολογιστή (laptop). Συνεπώς η επικοινωνία του φορητού υπολογιστή με την συσκευή ιχνηλάτησης υλοποιήθηκε μέσω θύρας usb.

Για την υλοποίηση της επικοινωνίας χρειάστηκαν:

- μια συνάρτηση ( η τέταρτη συνάρτηση) που εκτελείται κάθε φορά που το πρόγραμμα δίνει την εντολή εκτέλεσης αυτής. Η συνάρτηση αυτή "ανοιγοκλείνει" τη θύρα usb σε συνεχή και τακτά χρονικά διαστήματα. Αυτό είναι απαραίτητο για την επικοινωνία του υπολογιστή με τη συσκευή Zaber T-RS 60.

```

private static ZaberPortFacade CreatePortFacade()

{

    var packetConverter = new PacketConverter();
    packetConverter.MillisecondsTimeout = 50;
    var defaultDeviceType = new DeviceType();
    defaultDeviceType.Commands = new List<CommandInfo>();
    var portFacade = new ZaberPortFacade();
    portFacade.DefaultDeviceType = defaultDeviceType;
    portFacade.QueryTimeout = 1000;
    portFacade.Port = new TSeriesPort(
        new System.IO.Ports.SerialPort(),
        packetConverter);
    portFacade.DeviceTypes = new List<DeviceType>();
    return portFacade;

}

```

- τις εντολές που καλούν την συνάρτηση που αναφέρεται παραπάνω.

```


ZaberPortFacade facade = CreatePortFacade();
    facade.Open("COM3");
    Conversation conversation = facade.GetConversation(1);
    var topic = conversation.StartTopic();

```



## ΠΑΡΑΡΤΗΜΑ 1

### ΠΙΝΑΚΑΣ ΧΑΡΑΚΤΗΡΙΣΤΩΝ ZABER T\_RS 60 A

	Value	Alternate Unit
<b>Specification (click for definition)</b>		
Microstep Size (Default Resolution)	0.000234375 degrees	4.091 urad
Integrated Controller	Yes	
Range	360 degrees	
Accuracy	+/- 0.05 degrees	+/- 0.873 mrad
Repeatability	< .02 degrees	< 0.349 mrad
Backlash	< 0.06 degrees	< 1.047 mrad
Maximum Speed	13.2 deg/s	2.2 rpm
Minimum Speed	0.0011 deg/s	19.198 urad/s
Encoder Type	None	
Maximum Continuous Torque	80 N-cm	113.3 oz-in
Maximum Centered Load	200 N	44.9 lb
Maximum Cantilever Load	410 N-cm	580.6 oz-in
Guide Type	Pre-loaded ball bearing	
Stage Diameter	60 mm	2.362 "
Maximum Current Draw	450 mA	
Power Supply	12-16 VDC	
Power Plug	2.1mm Center Positive	
Motor Type	Stepper (2 phase)	
Inductance	4 mH	
Default Resolution	1/64 of a step	
Data Cable Connection	Minidin 6	
Mechanical Drive System	Precision Worm Gear	
Limit or Home Sensing	Magnetic home sensor	
Manual Control	Yes, Potentiometer	
Operating Temperature Range	0-50 degrees C	

RoHS Status	Compliant
CE Compliant	Compliant
Weight	0.48 kg

## ΠΑΡΑΡΤΗΜΑ 2 | ΚΩΔΙΚΑΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Στο παράρτημα αυτό παρατίθεται ο κώδικας που συντάχθηκε σε γλώσσα C# για την υλοποίηση του μοντέλου οδήγησης ιχνηλάτη αζιμούθιας γωνίας του ήλιου.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using log4net;
using Zaber;
using Microsoft.VisualBasic.FileIO;
using System.Threading;
using System.IO;
namespace ConsoleApplication1
{
    class Program
    {
        public static double geomik;
        public static double geopla;
        public static double Lst;
```

```

public static double ΔTgmt;

public static void readGPS()

{

    Console.WriteLine("Οι δεκαδικοί αριθμοί δίνονται με
κόμμα");

    Console.WriteLine("Δώσε γεωγραφικό μήκος");
    string input1 = Console.ReadLine();

    Console.WriteLine("Δώσε γεωγραφικό πλάτος");
    string input2 = Console.ReadLine();

    Console.WriteLine("Δώσε μεσημβρινό αναφοράς");
    string input3 = Console.ReadLine();

    Console.WriteLine("Δώσε διαφορά ζώνης ώρας");
    string input4 = Console.ReadLine();

    geomik = Double.Parse(input1);
    geopla = Double.Parse(input2);
    Lst = Double.Parse(input3);
    ΔTgmt = Double.Parse(input4);

    Console.WriteLine("Mhkos" + geomik.ToString() + "Pla-
tos" + geopla.ToString() + "Lst" + Lst.ToString());

}

```

```

public static double[] readParams()

{

    // βρίσκουμε την ημέρα του χρόνου

    DateTime dy = DateTime.Now;
    Console.WriteLine("Date is date 1h " +
    dy.DayOfYear.ToString());

    //αποθηκεύω την ημέρα του χρόνου σε μεταβλητή int

    Int32.Parse(dy.DayOfYear.ToString());
    int imera_xronou;
    imera_xronou = Int32.Parse(dy.DayOfYear.ToString());

    //υπολογισμός της γωνίας ημέρας B

    double B_maires;
    double B_rad;
    B_maires = 360 * (imera_xronou - 81) / 365;
    B_rad = B_maires * 3.14159265 / 180;

    //υπολογισμός της εξίσωσης χρόνου E

    double e_xronou;

```

```

e_xronou = 9.87 * Math.Sin((2 * B_rad)) - 7.53 *
Math.Cos(B_rad) - 1.5 * Math.Sin(B_rad);
Console.WriteLine(e_xronou + "exronou");

//Υπολογισμός Tlocal

//Υπολογισμός της ώρας τώρα

TimeZone tz = TimeZone.CurrentTimeZone;
Console.WriteLine(tz.GetUtcOffset(DateTime.Now));
int mytz = (int)ΔTgmt;
int wra_twra = DateTime.Now.Hour -
(tz.GetUtcOffset(DateTime.Now).Hours - mytz);
int min_twra = DateTime.Now.Minute;
int sec_twra = DateTime.Now.Second;
double gonia_d;
double sin_imeras;
sin_imeras = Math.Sin(((360 * (imera_xronou + 284)) /
365) * 3.14159265 / 180);

//υπολογισμός της γωνίας δ

gonia_d = 23.45 * sin_imeras;

//μετατροπή της γωνίας δ σε rad

double rad_gonia_d;
rad_gonia_d = (3.14159265 * gonia_d) / 180;

```

```

// ΥΠΟΛΟΓΙΣΜΟΣ ΣΤΟΙΧΕΙΩΝ ΓΙΑ ΑΝΑΤΟΛΗ ΚΑΙ ΔΥΣΗ

// Α Π Ο   Ε Δ Ω ...

//υπολογισμός γωνίας ώρας για τις ώρες ανατολής και
δύσης

//πρώτα υπολογισμός του cosω

double cosω;
cosω = (-((Math.Sin(rad_gonia_d) * Math.Sin(geopla *
3.14159265 / 180)) / (Math.Cos(rad_gonia_d) *
Math.Cos(geopla * 3.14159265 / 180))));

//υπολογισμός της ωριαίας γωνίας ω, σε rad

double ω_rad_anatolis = Math.Acos(cosω);

//υπολογισμός της ωριαίας γωνίας ω, σε μοίρες

double ω_maires_anatolis = (ω_rad_anatolis * 180) /
3.14159265;

//υπολογισμός Tsolar ( για τη δύση )

double Tsolar_anatolis;

```

```

Tsolar_anatolis = ω_maires_anatolis / 15 + 12;
Console.WriteLine("dysis solar " + Tsolar_anatolis);

//υπολογισμός Tlocal, της τοπικής ώρας για την ανατολή
και τη δύση του ηλίου

//Lst : ο κεντρικός μεσημβρινός για την ώρα τοπικής
ζώνης, -30 μοίρες για την Ελλάδα

//Llocal : το γεωγραφικό μήκος της τοποθεσίας όπου
προσδιορίζεται η ώρα

double Tlocal_dysis;
double Tlocal_anatolis;
Tlocal_dysis = Tsolar_anatolis - (e_xronou / 60) - ((4
* (Lst - geomik)) / 60);
Tlocal_anatolis = (12 - (ω_maires_anatolis / 15)) -
(e_xronou / 60) - ((4 * (Lst - geomik)) / 60);
Console.WriteLine(Tlocal_anatolis + " anatolis");
Console.WriteLine(Tlocal_dysis + " dysis");

//πραγματική ώρα ανατολής

double h_anatolis;
double h_dysis;
double ypol;
double min_anatolis;
double sec_anatolis;
double min_dysis;
double sec_dysis;
double gonias;

```



```
//Math.Truncate(number) για επιστροφή του ακέραιου
μέρους. Με την κάτωθι διαδικασία μετατρέπουμε την ώρα
ανατολής
```

```
// από δεκαδικό αριθμό σε μορφή hh:mm:ss
```

```
h_anatolis = Math.Truncate(Tlocal_anatolis);
ypol = Tlocal_anatolis - h_anatolis;
min_anatolis = Math.Truncate(ypol * 60);
ypol = ypol * 60 - min_anatolis;
sec_anatolis = Math.Truncate(ypol * 60);
string time_anatolis = (h_anatolis.ToString() + ":" +
min_anatolis.ToString() + ":" + sec_anatolis.ToString()
+ " πμ");
Console.WriteLine("ora anatolis: " +
h_anatolis.ToString() + ":" + min_anatolis.ToString() +
":" + sec_anatolis.ToString() + " πμ");
```

```
//Ακολουθούμε την ίδια διαδικασία για την ώρα δύσης
```

```
h_dusis = Math.Truncate(Tlocal_dusis);
ypol = Tlocal_dusis - h_dusis;
min_dusis = Math.Truncate(ypol * 60);
ypol = ypol * 60 - min_dusis;
sec_dusis = Math.Truncate(ypol * 60);

string time_dusis = (h_dusis.ToString() + ":" +
min_dusis.ToString() + ":" + sec_dusis.ToString() + "
```

```

μμ");
Console.WriteLine(h_dusis.ToString() + ":" +
min_dusis.ToString() + ":" + sec_dusis.ToString() + "
μμ");
//Υπολογίζουμε την γωνία ανατολής για την συγκεκριμένη
ημέρα του χρόνου

gonia = Math.Acos(-Math.Sin(rad_gonia_d) /
Math.Cos(geopla * 3.14159265 / 180));
gonia = gonia * 180 / 3.14159265;

//// ..... Ω Σ      Ε Δ Ω !!!

double[] parameters = new double[11];
parameters[0] = h_anatolis;
parameters[1] = min_anatolis;
parameters[2] = sec_anatolis;
parameters[3] = h_dusis;
parameters[4] = min_dusis;
parameters[5] = sec_dusis;
parameters[6] = Lst;
parameters[7] = ΔTgmt;
parameters[8] = e_xronou;
parameters[9] = geomik;
parameters[10] = geopla;
return parameters;

}

public static int[] ReadTime(int mytz)

```

```

{

    TimeZone tz = TimeZone.CurrentTimeZone;
    Console.WriteLine(tz.GetUtcOffset(DateTime.Now));
    int wra_twra = DateTime.Now.Hour -
    (tz.GetUtcOffset(DateTime.Now).Hours - mytz);
    int min_twra = DateTime.Now.Minute;
    int sec_twra = DateTime.Now.Second;
    int[] time = new int[3];
    time[0] = wra_twra;
    time[1] = min_twra;
    time[2] = sec_twra;
    return time;

}

private static ZaberPortFacade CreatePortFacade()

{

    var packetConverter = new PacketConverter();
    packetConverter.MillisecondsTimeout = 50;
    var defaultDeviceType = new DeviceType();
    defaultDeviceType.Commands = new List<CommandInfo>();
    var portFacade = new ZaberPortFacade();
    portFacade.DefaultDeviceType = defaultDeviceType;
    portFacade.QueryTimeout = 1000;
    portFacade.Port = new TSeriesPort(
    new System.IO.Ports.SerialPort(),
    packetConverter);
    portFacade.DeviceTypes = new List<DeviceType>();
    return portFacade;
}

```

```
}
```

```
static void Main(string[] args)
```

```
{
```

```
    ZaberPortFacade facade = CreatePortFacade();
```

```
    facade.Open("COM3");
```

```
    Conversation conversation = facade.GetConversation(1);
```

```
    var topic = conversation.StartTopic();
```

```
    //Διαβάζουμε τις παραμέτρους της περιοχής όπου θα  
    τοποθετήσουμε τον ιχνηλάτη
```

```
    readGPS();
```

```
    //Άνοιγμα αρχείου όπου αποθηκεύονται τα δεδομένα της  
    ημέρας
```

```
    StreamWriter file = new StreamWrit-
```

```
er(@"C:\Users\reitz\Documents\stick\Zber1\Zber\csv\" +
```

```
DateTime.Now.DayOfYear.ToString() + ".csv", true);
```

```
while (true)
```

```
{
```

```

//Υπολογισμός της ημέρας του χρόνου

Console.WriteLine(ΔTgmt);
double[] stoixeia = readParams();
string yo;
DateTime dy = DateTime.Now;
Console.WriteLine("Date is " +
dy.DayOfYear.ToString());
yo = dy.ToString("yyyy");

//αποθήκευση ημέρας του χρόνου σε μεταβλητή int

Int32.Parse(dy.DayOfYear.ToString());
int imera_xronou;
imera_xronou = Int32.Parse(dy.DayOfYear.ToString());

//Υπολογισμός της ώρας τώρα

int[] wras = ReadTime((int)stoixeia[7]);
int wras_twra = wras[0];
int mins_twra = wras[1];
int secs_twra = wras[2];

//Μετατρέπουμε την ώρα ανατολής και δύσης σε μορφή η
οποία, μας βοηθάει στους ελεγχούς των βρόχων

DateTime dateAnatolis = new
DateTime(DateTime.Today.Year, DateTime.Today.Month,
DateTime.Today.Day, (int) stoixeia[0], (int)
stoixeia[1], (int) stoixeia[2]);

```

```

Console.WriteLine("Date Anatolis: " + dateAnatolis.ToString());
DateTime dateDysis = new DateTime(DateTime.Today.Year,
DateTime.Today.Month, DateTime.Today.Day,
(int)stoixeia[3], (int)stoixeia[4], (int)stoixeia[5]);
Console.WriteLine("Date Dysis: " + dateDysis.ToString());
Console.WriteLine("Date Now: " + DateTime.Now);
Console.WriteLine("Ticks: " + DateTime.Now.Ticks);

DateTime ypnos = new DateTime(dateAnatolis.Ticks -
DateTime.Now);

```

```

// Συγκρίνουμε την ώρα τώρα με την ώρα ανατολής

```

```

while (dateAnatolis > DateTime.Now)

```

```

{

```

```

//Υπολογισμός της χρονικής διάρκειας που
απομένει μέχρι την ανατολή του ηλίου

```

```

//για αυτή την χρονική διάρκεια τοποθετείται το
tracker σε κατάσταση ύπνου

```

```

//αυτό το γίνεται για λόγους οικονομίας στη
λειτουργία του επεξεργαστή και κατανάλωσης
ενέργειας

```

```

TimeSpan ypnos = new
TimeSpan(dateAnatolis.Ticks -
DateTime.Now.Ticks);

```

```

        Console.WriteLine("sleep for: " + ypnos.TotalMilliseconds);
        Thread.Sleep(ypnos);
    }

while (dateDysis > DateTime.Now)
{

    //Υπολογισμός της ώρας τώρα

    int[] wra = ReadTime((int)stoixeia[7]);
    int wra_twra = wra[0];
    int min_twra = wra[1];
    int sec_twra = wra[2];

    //Μετατροπή τοπικής ώρας σε μορφή δεκαδικού

    double Tlocal = sec_twra / 60;
    Tlocal = (min_twra + Tlocal) / 60;
    Tlocal = wra_twra + Tlocal;
    Console.WriteLine(Tlocal + " Tlocal ");

    //υπολογισμός της γωνίας ημέρας B

    double B_maires;
    double B_rad;
    double b_maires;
    double Z_maires;
    B_maires = 360 * (imera_xronou - 81) / 365;

```

```
B_rad = B_maires * 3.14159265 / 180;
```

```
//υπολογισμός της εξίσωσης χρόνου E
```

```
double e_xronou;
```

```
e_xronou = 9.87 * Math.Sin(2 * B_rad) - 7.53 *  
Math.Cos(B_rad) - 1.5 * Math.Sin(B_rad);
```

```
//Υπολογισμός Tsolar
```

```
double Tsolar = Tlocal + (e_xronou / 60) + ((4  
* (stoixeia[6] - stoixeia[9]))/60);
```

```
//Υπολογισμός γωνίας ω
```

```
double ω_maires;
```

```
ω_maires = 15 * (Tsolar - 12);
```

```
double ω_rad = (ω_maires * 3.14159265) / 180;
```

```
if (ω_maires > 0)
```

```
{
```

```
    double gonia_d;
```

```
    double sin_imeras;
```

```
    sin_imeras = Math.Sin(((360 * (imera_xronou  
+ 284)) / 365) * 3.14159265 / 180);
```



```
//υπολογισμός της γωνίας δ
```

```
gonia_d = 23.45 * sin_imeras;
```

```
//μετατροπή της γωνίας δ σε rad
```

```
double rad_gonia_d;
```

```
rad_gonia_d = (3.14159265 * gonia_d) / 180;
```

```
//Υπολογισμός ύψους ηλίου β
```

```
double sin_b = (Math.Sin(rad_gonia_d) *  
Math.Sin(stoixeia[10] * 3.14159265 / 180))  
+ (Math.Cos(rad_gonia_d) *  
Math.Cos(stoixeia[10] * 3.14159265 / 180) *  
Math.Cos(ω_rad));  
double b_rad = Math.Asin(sin_b);  
b_maires = b_rad * 180 / 3.14159265;  
Console.WriteLine(b_maires + " υψος σε  
μοιρες");
```

```
//Υπολογισμός αζιμούθιας γωνίας Z
```

```
double Z_rad = Math.Acos((sin_b *  
Math.Sin(stoixeia[10] * 3.14159265 / 180) -  
Math.Sin(rad_gonia_d)) / (Math.Cos(b_rad) *  
Math.Cos(stoixeia[10] * 3.14159265 /  
180)));  
Z_maires = Z_rad * 180 / 3.14159265;
```

```

//Μετατροπή αζιμούθιας γωνίας σε
"μικροβήματα" για συμβατότητα

Console.WriteLine(Z_maires + " αζιμούθια σε
μοίρες");
double az_gonia = (768000 / 180) *
Z_maires;
int azimuthia_gonia;
azimuthia_gonia = (int)az_gonia;
Console.WriteLine(az_gonia + " αζιμούθια
γωνία ως προς νότο");

//Μετακίνηση του tracker στην αζιμούθια
γωνία

int x = (768000 + azimuthia_gonia);

conversation.Request(Command.MoveAbsolute,
x);
Console.WriteLine(x + " απόλυτη θέση σε
μικροβήματα");

}

else

{

```

```

double gonia_d;
double sin_imeras;
sin_imeras = Math.Sin(((360 * (imera_xronou
+ 284)) / 365) * 3.14159265 / 180);

//υπολογισμός της γωνίας δ

gonia_d = 23.45 * sin_imeras;

//μετατροπή της γωνίας δ σε rad

double rad_gonia_d;
rad_gonia_d = (3.14159265 * gonia_d) / 180;

//Υπολογισμός ύψους ηλίου β

double sin_b = (Math.Sin(rad_gonia_d) *
Math.Sin(stoixeia[10] * 3.14159265 /
180)) + (Math.Cos(rad_gonia_d) *
Math.Cos(stoixeia[10] * 3.14159265 / 180) *
Math.Cos(ω_rad));
double b_rad = Math.Asin(sin_b);
b_maires = b_rad * 180 / 3.14159265;
Console.WriteLine(b_maires + " ύψος σε
μοίρες");

//Υπολογισμός αζιμούθιας γωνίας Z

double Z_rad = Math.Acos((sin_b *

```

```

Math.Sin(stoixeia[10] * 3.14159265 / 180) -
Math.Sin(rad_gonia_d)) / (Math.Cos(b_rad) *
Math.Cos(stoixeia[10] * 3.14159265 /
180)));
Z_maires = - Z_rad * 180 / 3.14159265;

//Μετατροπή αζιμούθιας γωνίας σε
"μικροβήματα" για συμβατότητα

Console.WriteLine(Z_maires + " αζιμούθια
γωνία σε μοίρες");
double az_gonia = (768000 / 180) *
Z_maires;
int azimuthia_gonia;
azimuthia_gonia = (int)az_gonia;
Console.WriteLine(az_gonia + " αζιμούθια
γωνία ως προς νότο");

//Μετακίνηση του tracker στην αζιμούθια
γωνία

int x = (768000 + azimuthia_gonia);
conversation.Request(Command.MoveAbsolute,
x);
Console.WriteLine(x + ".....");
}

Thread.Sleep(2000);
file.WriteLine(imera_xronou + ";" + wra_twra +
";" + min_twra + ";" + sec_twra + ";" +

```

```

        b_maires + ";" + (-Z_maires) + ";");
    }

    file.Flush();
    //Υπολογισμός της χρονικής διάρκειας μέχρι την αλλαγή
    της ημέρας

    //και τοποθέτηση του ιχνηλάτη σε κατάσταση ύπνου μέχρι
    την αλλαγή της ημέρας

    TimeSpan mesanyxta = new TimeSpan((new
    DateTime(DateTime.Today.Year, DateTime.Today.Month,
    DateTime.Today.Day + 1, 0, 0, 0)).Ticks -
    DateTime.Now.Ticks);
    Console.WriteLine("sleep for: " +
    mesanyxta.TotalMilliseconds);
    Thread.Sleep( (int) mesanyxta.TotalMilliseconds + 1000);

    }
}
}
}

```



## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

- 1) Κωνσταντίνος Θ Δέρβος Φωτοβολταϊκά Συστήματα: από την Θεωρία στην Πράξη, Πανεπιστημιακές Εκδόσεις ΕΜΠ
- 2) <http://www.zaber.com/wiki/Manuals/T-RSW>
- 3) <http://www.zaber.com/support/>
- 4) ΜΗΤΖΙΑΣ ΑΘΑΝΑΣΙΟΣ, Διπλωματική Εργασία: ΣΥΓΚΕΝΤΡΩΤΙΚΑ ΗΛΙΑΚΑ ΣΥΣΤΗΜΑΤΑ, Πολυτεχνική Σχολή Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
- 5) [http://re.jrc.ec.europa.eu/pvgis/cmeps/eu\\_cmsaf\\_hor/G\\_hor\\_GR.png](http://re.jrc.ec.europa.eu/pvgis/cmeps/eu_cmsaf_hor/G_hor_GR.png)
- 6) [http://www.nais.com.gr/lang\\_el/projects/view\\_details/2](http://www.nais.com.gr/lang_el/projects/view_details/2)
- 7) [http://www.energia.gr/article.asp?art\\_id=27935](http://www.energia.gr/article.asp?art_id=27935)
- 8) [http://users.sch.gr/omixara/eco\\_energy/ananeosimes/geothermia.htm](http://users.sch.gr/omixara/eco_energy/ananeosimes/geothermia.htm)
- 9) <http://ggeorgala.wordpress.com/2009/04/17/ηλιακό-διάγραμμα>
- 10) <http://www.hottowel.gr>