

Improve The Effectiveness In The Solar Tracking System Using Nodemcu Microcontroller

Dr.V.Shanmugasundaram¹, Anjali Nighoskar², Sampathkumar R³, Dr C Balakrishnan⁴, K.Suresh⁵, Dr Sudharsan Jayabalan⁶

¹Assistant Professor, Department of EEE, Sona College of Technology, Salem, Tamil Nadu-636005 ²Assistant Professor, School of Engineering and Technology, Jagran Lakecity University,

 Assistant Professor, School of Engineering and Technology, Jagran Lakecity University, Mugaliyachap, Ratibad 462044.
 ³Assistant professor (Sl.G), Department of EEE, KPR Institute of Engineering and Technology, Coimbatore-641407
 ⁴Assistant Professor, Alagappa Institute of Skill Development, Alagappa University, Karaikudi-63003.
 ⁵Associate Professor, Hyderabad Institute Of Technology And Management, Gowdavelly(Village), Medchal-501401.

⁶Associate Professor, Department of Electronics and Communication Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad-500090, India

Abstract

Solar energy is the only renewable energy that has swiftly acquired reputation and relevance. We can create a large quantity of energy using the solar tracker, which increases the efficiency of the solar panels. The effectiveness of the solar panel is due to its parallel proportion with the sun's beams. Financially, its setup fee is expensive, especially when cheaper alternatives are available. The planning and engineering method of a model for a solar tracking system with a fixed dimension of flexibility are explained in detail in this work. The NodeMcu microcontroller is used in the primary control circuit. The programming of this system is done in such a way that the LDR sensor, in response to the identification of sun rays, will guide the DC Motor in the position in which the solar panel would rotate. The solar panel is then positioned in such a way that it receives the large quantities of sun rays possible. Even though a boost in solar panel effectiveness resulted in a significant gain in ineffectiveness, the ideal was still a long way off.

1. INTRODUCTION

The Sun had long been regarded as a vital source of energy. Solar energy is a more ecosustainable source than its rivals. The growth of technology has resulted in the development of ways for putting this energy into productive usage. It might be thermal energy, fuel generation. Photovoltaic (PV) and focused solar power (CSP) systems convert solar energy captured by the earth into electricity. The nationalized solar power is used by the solar tracking device through a channel of solar panels, an aligned scaffolding of photo-voltaic cells [1]. Solar cells made of materials other than silicon are somewhat more expensive to manufacture, making silicon the ideal material for solar tracking systems [2]. Amorphous silicon cells can operate at high temperatures, although their effectiveness is lower than other silicon types. [3]. The PV units are grouped in a solar panel or a PV array in such a way that



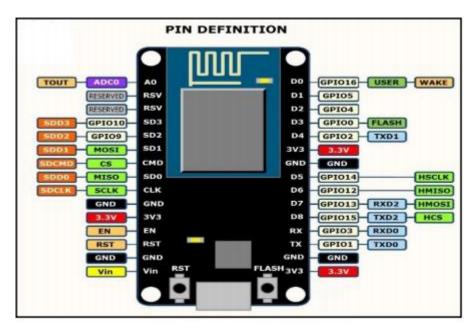
photons are used to excite the electrons of the material contained within the solar cells. The average quantity of sunlight obtained by solar panels is determined by the sun's location [4].

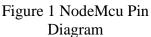
The manipulation of the energy source is encouraged because of the numerous blooming ways in which it may be utilized to push for a change in the conservation of other resources [5]. The mechanism is responsible for providing precise motions to follow the sun's path during the day. The device's prototype has been designed to endure adverse weather conditions. The solar tracking system's operation is divided into two categories: linear scale tracker and dual-axis sensor [6]. In terms of small-scale solar power plants, single-axis monitoring might be regarded as one of the useful systems or a primary option. The tracker is set up to track a single row of data. The many configurations where a single axis tracker can be constructed are described below. All of the techniques that are maintained have the same basic operation. The relationship of the sunlight with the collectible surface is calculated and evaluated, and the collectors are then charged with tracking the sun's motion to capture a higher proportion of solar radiance [7].

2. IMPLEMENTATION

The "Automatic Solar Tracking System" project is created by assembling different components such as a solar panel with a 12-volt output, a NodeMcu as an MCU, a motor controller - an IC L293D, two LDR sensor modules, a 10 r.p.m. basic DC motor, a current sensor, and a 9 V batteries. The stated project's structure is made up of a wooden foundation put on the ground, iron rods fastened on both sides in a cross-shaped way, and a hollow cylindrical rod connected on both sides with a Motor drive clinging to one edge of the hollow rod. The circuit of the solar tracking system is broken into three parts. The input stage contains two LDR modules that are arranged to form a voltage divider circuit, the microcontroller is programmed using the Arduino ide software that is installed in the system, and the drive circuit, which includes a DC motor, aids in the rotation of the solar cell. The motor driver is equipped with three terminals: two for motors outputs and inputs, and a third for power input. The motor input module is connected to two of the Arduino UNO's 14 digital input/output pins, while the motor output circuit is connected to the DC motor. With NodeMcu analog inputs, the two LDR sensor components are attached to the scaffolding. The light-dependent resistors are therefore attached to each surface of the panel along its span. Figure 1 displays the NodeMcu Pin Diagram.







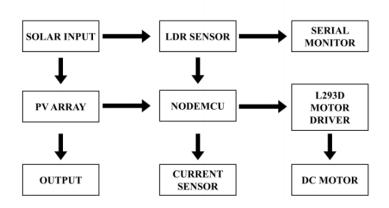


Figure 2 Sketch of Automatic Solar Tracking setup

The embedded software is composed of a basic hardware language is called processing, which is comparable to the C language put into the Arduino UNO. Three separate phases are engineered individually before being merged into one device. This method, which is comparable to incremental refinement in modular programming, was chosen because it ensures a precise and logical approach that is basic and easy to comprehend. This often guarantees that any faults are thoroughly investigated and remedied separately. The block diagram of the Automatic Solar Tracking setup is shown in Figure 2

3. WORK DESIGN

The Controlling Circuit

A microcontroller is used in the controlling circuit. The controller is programmed with a technique that compares and detects the direction of the highest light intensity. The picture sensor inputs are processed by the microprocessor, and the results are sent to the driving module after calculation. NodeMcu, an IoT-based system, is the microprocessor we utilize in our architecture. Figure 3 shows the circuit diagram for the solar tracking system.



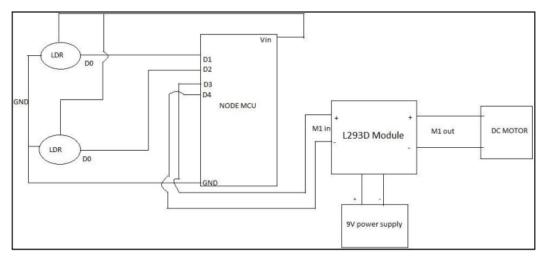


Figure 3 Circuit Diagram

Software Design

The controller is programmed using an IoT platform. In the Arduino Integration Development Environment, the methodology is created (IDE). The setup's upload frequency is increased at 9600, and it's linked to the ESP8266 board's COM5. Figure 4(a-c) shows the various set up of the NodeMcu controller for the solar tracking system.

Active Settich Active Settich Fit Encoding & Reload Fit Encoding &	0 🖬 🛙	Auto Format	Ctrl+T						l
I deta Ol port 1/01 I deta Ol port 1/01 I deta Ol I deta Ol		Archive Sketch							
put: put: put: put: put: 1:cep() put: put: put: 1:cep() put: Base: Put: Upload put: Debug put: Put: put: Put:<	rtch_may09	Fix Encoding & Reload							
1 conc) (Secial workship Cuti-series with workship 1 conc) (WF101 / WF40NA Farmware Update************************************		Manage Libraries	Ctrl+Shift+I						
Lange () 1 pre: "joon" Band: "NedANIAL Farmware Updater Band: "NedANIAL () 1.0 (ESP-12E Medule)" Upload Speed: "9000" CPU Frequency: "900 M-4" CPU Frequency: "900 M-4" CPU Frequency: "900 M-4" CPU Frequency: "900 M-4" Debug net: "Danibed" Debug net: "Dan	but Nons	Serial Monitor	Ctrl+Shift+M						
Alegn 10 Pore: 1/007 Ugload Speed: 1/000 CPU Frequency: 100 M4/2 Flash Size: 1/4M (co SPIFS) Debug pert: Toxebieff Debug pert: Toxe		Serial Plotter	Ctrl+Shift+L						
puer your Beard: "NodeMCU L0 (ESP-12E Module)" > Upload Speed "NoDO" > CPU Frequency "NodeMCU L0 (ESP-12E Module)" > CPU Frequency "NodeMC" > Plansh Size: "MM (no SPIFFS)" > Debug peet: "Nome" > Debug peet: "Nome" > Debug peet: "Nome" > Debug peet: "Nome" > Not P Variant: "V2 Lower Memory" > VTables: "Flan" > Free Flant: "COMS' \$ Ecception: "Usabled" > Free Flant: "COMS' \$ Ecception: "Usabled" > Free Flant: "COMS' \$ Forgarmmer: "AVRSP mikl" \$ COMS	loop() {	WiFi101 / WiFiNINA Firmware Updater							
CPU Frequency: 300 MB-4" > Flash, Sice: "M4 (oo SPF5)" > Debug Lexel: "None" > Intel Debug Lexel: "None" > Mol Debug Lexel: "None" > Value: "Rach" > Exception: "Diabled" > Free Text" COMS* Secial ports Get Scotts" Secial ports Free Text" COMS* COM4 Programmer: "AVRSP mAil"		Board: "NodeMCU 1.0 (ESP-12E Module)"							
Flash Sice: "Mid (no SPFS)" > Delug port: "Disklord" > Delug port: "Disklord" > Idf Variant: "Viz Lower Memory" > Vizbles: "Flash" > Exceptions: "Disklord" > Exceptions: "Disklord" > Exceptions: "Disklord" > Frame Flash: "Only Statch" > Get Board Info Secial ports Get Board Info COM4 Programmer: "AVROP mikl"		Upload Speed: "9600"	>						
Debug Cent: "Diabled" > Debug Lent: "Nonn" > belf Vaintit: "V2 Lover Memory" > VTbable: "Flach" > Exception: "Diabled" > Exception: "Diabled" > Free Teach "Only Statch" > Get Board Info COM4 Programmer: "AVESP mkl"		CPU Frequency: "80 MHz"	>						
Debug Level: "Nona" > Mol Vainet: "V2 Lover Memory" > VTable: "V2 Lover Memory" > Exceptions: "Disabled" > Exceptions: "Disabled" > Free: "COMS" > Get Board Info COM4 Programmer: "AVROP mik!"		Flash Size: "4M (no SPIFFS)"	>						
baff Variant: "v2 Lower Memory" > VTables: "Rath" > Exceptions: "Doabled" > Erne Rath: "COMS" > Peet: "COMS" > Get Board Info COM4 Programmer: "AVRISP mixil"		Debug port: "Disabled"	>						
VTables: "Rah" > Exceptions: "Disabled" > Frame Rhah: "Coldy Sketch" > Peet: "COMS" Steinal ports Get Bloard Info Programmer: "AVRSP mABI"		Debug Level: "None"	3						
Exceptions "Disabled" Frame Flash" COMS Peter "COMS" Get Board Info COM4 Programmer "AVRISP mkl" COM5		IwIP Variant: "v2 Lower Memory"	>						
Ernse Flash: "Only Sketch" > Ped: "COM5" 3 Gett Bard Info COM4 Programmer: "AVRSP mk8"		VTables "Flash"	>						
Port: "COMS" Serial ports Get Board Info COM4 Programmer: "AVRISP mk8" 2		Exceptions "Disabled"	>						
Get Board info COM4 Programmer "AVRSP måll" COM5		Erase Flash: "Only Sketch"	>		_				
Programmer "AVRSP miki" 2 COM5		Port: "COM5"	1	Serial ports					
Programmer: "AVRISP mkl"		Get Board Info		COM4					
		Department "AUDICE cold"		COM5					
pum policiend									
		ban boodado							
	at cc	.arduino.utils.network.FileDownloa	der.download(F	leDownloader	.java:132)				
st cc.ardsin.stils.network.fileDowloader.dowload (fileDowloader.jevs:132)									
st co.arduin.utile.network.FileDownineder.downined (FileDownineder.jevs132) at co.arduin.contribution.DowlinedbleContributionsDowlineder.downined(DowlinedbleContributionsDownineder.jevs137)									

Figure 4(a): Arrangement of NodeMcu - Port link



😳 sketch_may09a 4	Arduino 1.8.8															-	0	×
File Edit Sketch To	ols Help																	
	Auto Format	Ctrl+T																ø
	Archive Sketch																	_
sketch_may09	Fix Encoding & Reload																	
<pre>void setup()</pre>	Manage Libraries	Ctrl+Shift+I																^
// put your	Serial Monitor	Ctrl+Shift+M																- 1
3	Serial Plotter	Ctrl+Shift+L																
void loop() {	WiFi101 / WiFiNINA Firmware Updater																	
// put your	Board: "NodeMCU 1.0 (ESP-12E Module)"	- >																
1	Upload Speed: "9600"		115200															
	CPU Frequency: "80 MHz"	1	9600															
	Flash Size: "4M (no SPIFFS)"	1	57600															
	Debug port: "Disabled"	1	256000															
	Debug Level: "None"	1	512000															
	IwIP Variant: "v2 Lower Memory"	1,	921600															
	VTables: "Flash"	>																
	Exceptions: "Disabled"	>																
	Erase Flash: "Only Sketch"	>																
	Port	>																
	Get Board Info																	
	Programmer: "AVRISP mkll"	>																
	Burn Bootloader																	
																		×
at cc.	arduino.utils.network.FileDownloa	der.download(Fi	leDownloade	r.java:132)														^
at cc. 5																		
																		~
		-	-	-		esten/CU 1	ilo (Estr-12E	Medule), 80 MP	na, Flash, G	material, 46	100 51475	1 VZ LOWE	wenney, D				3.4	1 PM
🖬 🖸 l'm 🤇		Ø	е 📮	é	8									~ 1	10 2	48) E		/2019

Figure 4(b): Arrangement of NodeMcu – Speed

File Edit Satch Tools Help Void setting (// put your setup oods here, to run rep / // put your setup code here, to run rep / // put your setup code here, to run rep - // put your setup code here, to run rep - - - - - - - - - - - - -	8.8				- 0	×
Skith_map/Sia void secup() { // pit your secup code here, to run rep // pit your main code here, to run rep						
widd getug() { // put your setup oode here, to run rep } I vidd loeg() { // put your sain code here, to run rep } I vidd loeg() { // put your sain code here, to run rep } I vidd loeg() { // put your sain code here, to run rep } I vidd loeg() { // put your sain code here, to run rep } I vidd loeg() { // put your sain code here, to run rep } exectors rep vidd loeg() { // put your sain code here, to run rep } exectors rep vidd loeg() { // put your sain code here, to run rep } exectors rep vidd loeg() { // put your sain code here, to run rep } vidd loeg() { // put your sain code here, to run rep } exectors rep vidd loeg() { // put your sain code here, to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put your sain to run rep } vidd loeg() { // put						ø
<pre>// put your setup code here, to run rep } vaid loop() { // put your main code here, to run rep } rep A // put your main code here, to run rep } rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep Rearts included in the package rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep Rearts included in the package rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here, to run rep rep A // put your main code here rep A // put your rep A // put yo</pre>						
Source Section Manager * * ************************************						^
J vidi Loop() { // put your main code here, to run rep // put your run rep // put your main code here, to run				~		- 1
<pre>veid1 app() { // put your main code here, to run rep Packanze // put your main code here, to run rep Packanze Packanze</pre>				^		- 1
// put your main code here, to run rep Industrimo SAND Bards (32-bits ANI Cortes-109-) by Industruino Bards included in this patiage Industruino SAND Bards (32-bits ANI Cortes-109-) by Industruino Bards included in this patiage Industruino SAND Bards (32-bits ANI Cortes-109-) by Industruino Bards included in this patiage General field More Infe General F30026 Bords (0.5-bits Ani (0.5-bits More Infe General F30026 Bords (0.5-bits Sands included in this patiage General F30026 Bords (0.5-bits Sands included in this patiage General F30026 Bords (0.5-bits Sands included in this patiage General F30026 Bords (0.5-bits Sands included in this patiage General F30026 Bords (0.5-bits Sands included in this patiage General F30026 Bords (0.5-bits Sands General F30026 Bords		lter your search				- 1
Industration SAND Baards (32-bits ARM Cortex-190+) by Industruine Baards Inducated in this package: Industruine D218. Colline. Intel Marse Links Baards Inducated in this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated in this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: Description of SEPERGE Community version 2.3.6 INSTALLED Baards Inducated In this package: SEPERGE Community Version SEPERGE Community Version 2.3.6 Induced Vision, SEPERGE Community Version 2.3.6 Induced Vision 2.				^		
Beards included in this package: Industrue D210. Colling. Mile More with exp8266 by ESP8266 Community version 2.3.0 INSTALLED Beards included in this package: Owner: ESP8226 Moldule. Generic ESP82265 Module, ESP0:ins (25P-13 Module). Adefnuit Peather HUZZAH ESP8266, Invent One, Xinakou (WH), ESPeate Lite 2.6, SSPeaters Lite 2.6, Peaceting 1.0, Robert D. 0, RobertOJ D. 10 (SSP-13 Module). NoderHOJ 1.0 (SSP-13 Module). NoderHOJ 1.0 (SSP-13 Module). SSPEater S	Here, co can rep					- 1
Industruino D210. Collise.http: More Life: exp8266 by 55P8266 Community vention 2.3.0 1167ALLED Bearls included in this package General: ED8226 Followide, General: Sector 200 Molecular, Addinut Prasterr HUZZAM ESPESIO. Invest: One, General: ED8226 Followide, General: Sector 200 Molecular, Addinut Prasterr HUZZAM ESPESIO. Invest: One, General: ED8226 Followide, Bearls (Sector 200 Molecular, Addinut Prasterr HUZZAM ESPESion, Invest: One, General: ED8226 Followide, D100 Molecular, D100 Molecular, D100 Molecular, Molec			uino			
Collean.bells Marse.Info exept206b by CSPP206 Community version 2.3.6 INSTALLED Board ESP206 Module, Onexeit CSPP225 Module, ESPCURO (SEP-13 Module), Adafruit Peather HUZZAM ESP2266, Envect One, StraBaro (WDI, SFIPASEO Instal, Onexeit CSPP225 Module, SEPCURO (SEP-13 Module), Adafruit Peather HUZZAM ESP2266, Envect One, StraBaro (WDI, SFIPASEO Instal, Dev. StraBaro (Lev.), Specifyrus ESP2260, Dimit Dev. StraBaro (SEP-12 Module), NodeHCU 1.0 (SEP-12 Module), Dimite Net, Wohl (SEP-12 Module), Dimite Net, Wohl (SEP-12 Module), StraBaro (SEP-12 Module), Dimite Net, Wohl (SEP-12 Modul		ikage:				- 1
eseR364 by ESPR366 Community version 2.3.6 INSTALLED Baards Included in this parkage: Owners ESPR366 Module, Owners ESPR365 Module, ESPCINO (ESP-13 Module), Adefnut Peartner HUZZAH ESPE366, Invest Ome, XinaBar (VM), ESPR3656 ILb 10, ESPR365 Like 30, Peacetti 30, Nodelk 30, Nodelk 30, Nodelk 31, Iones Module, Nodelk 31, 30 (ESP-125 Module), Olimec MOD-VITY-ESPE366 (CV), Sankhur ESPR3650 Tring, Sparkfun ESPR366 Tring Dev, SmedPae ESP-210, LOLIN(WENDS) DI R 28 minut, LOLIN(WENDS) DI mile Ibe, WendBar I EL, ESPINO (ESP-12 Module), ThataveTeet a ESPino, Withfo, Anduno, 4D Bystems gene IoD Kange, Digistump Ceir, Withduno, Amperia WH Sist, Community International Internati	Online help					
Rearts included in this package: General ESR226 Module, General ESR2255 Module, ESPDuino (ESP-13 Module), Adafruit Rearber NUZZ4H ESR2266, Invent Che, Stration CWD1, ESRess (Jat 3), ESRess (Jat 2), Otherwise 10, Discourt 20, Module), M	More info					
Rearts included in this package: General ESR226 Module, General ESR2255 Module, ESPDuino (ESP-13 Module), Adafruit Rearber NUZZ4H ESR2266, Invent Che, Stration CWD1, ESRess (Jat 3), ESRess (Jat 2), Otherwise 10, Discourt 20, Module), M						
Beards Included in NHI package General: 539226 Moulus, General: 539255 Mouluie, ESPDuino (ESP-13 Mouluie), Adaftuit Reather NUZ24H ESPE266, Invent Chee, 30x88ux (MV), ESPears (Jan J.), ESPears (Jan 22, Mouers), 10, Mouluie), Adaftuit (Fashber NUZ24H ESPE266, Invent Chee, 30x88ux (MV), ESPears (Jan J.), ESPears (Jan 22, Mouers), 10, Mouluie), 30, Mouluie), Mouluie), Mouluie), ESP-151, LOLIN(WHEND) SID 182 a min. LOLIN(WHEND) SID 10 min Ihou. NoIII, 100, Mouluie), SID 181, ESPINO, ISBN 21 Mouluie), Tastasyties's ESPino, Withfo, Anduron, 4D Systems gen4 IsD Range, Digistump Oak, Withshino, Amperka With Side, Espe-164, ESPears (Jan 20, Mouluie), SID 182, Mouluie), SID 184, SID 184, SID 184, SID 184, SID 184, Side Min Link, ESPeartin Core, Side Min Link, ESPeartin Core, Side Min Link, SID 184, Side Min Link, Side Side Min Link, Side Side Min Link, Side Side Side Side Side Side Side Side	esp8266 by E5P8266 C	munity version 2.5.0 INSTALLED				
Xinadox CW02, ESPesso Lite 1.0, ESPesso Lite 2.0, Process 1, DP. Pocenti 2.0, ModerNU 0.9 (ESP-12 Module), ModerNU 1.0 (ESP-122 Module), ModerNU 0-XIVTEREDBG(-EVI), Spatial 2014, ISEB026 Thins, Spatial Distribution, DBR-Smether ESP-210, LOLIN(VENDS) DI R 28 mini, LOLIN(VENDS) DI mini Pio, LOLIN(VENDS) DI mini Lite, WeMso DI R 1, ESPino (ESP-12 Module), ThatEasyleicit eStano, Wifrido, Adduno, 4D Systems gen4 IoD Kange, Digistump Oak, Wifriduino, Amperka Wifr Slot, Seeed Wio Link, ESPectro Core, Colling, Inter More, Info	Boards included in this	dicage				
EEE-210, LOLDINUEHOS) D1 R2 & mini, LOLINUWEHOS) D1 mini Pio, LOLINUWEHOS) D1 mini Lite, WeMas D1 R1, ESPino (ESP-12 Module, Thaitasylicitie's ESPino, Wifnio, Adduno, 4D Systems gen4 IoD Kenge, Digistump Oak, Wifsbulno, Amperka Wifi Slot, Seeed Wio Link, ESPectro Core, Colling, Infio						
Module), Thatlasytiev's ESFino, Writinio, Arduino, 4D Systems gen4 IzD Renge, Digistump Oak, Wifibulno, Amperka Wifi Slot, Seeed Wio Luke, ESPectro Core, Colona India More Infia						
Sections India Mana John						- 1
Marce softe		Core.				
						- 1
Errer downloading https://downloads.arduins.cc/padiages/padiage_index.json				~		- 1
	Error downloading https://do	oads.arduino.cc/packages/package_index.jso	n	OK		
	L					
						~
at cc.arduino.utils.network.FileEowsloader.download(FileEownloader.java:132)						^
at cc.ardmino.contributions.ExemioadableContributionsExemioadflowelloadableContributionsExemioadflowelloadar.java;137), 5 prote						
··· b ROLE						~
NadeWCU 1.0 (CSP-1.2E Medule), ED MHz; Flash, Otabled, 444 (no SPIF75), V2 Lower Memory, Databled, Rone, Only Seatch, 4600 on C			NodeMCU 1.0 (ESP-12E Module), 80 NHz, Flash, Disabled, 4M (n	o SPIFFS), v2 Lower Memory, Disab	ed, None, Only Sketch, 9500 on 0	ONB
	Ask me anything.	e 🖿 🛍 🥥 🥯				1 PM /2019

Figure 4c): Arrangement of NodeMcu - Installing Library

4. EXPERIMENT RESULTS

The project's results were derived using LDRs for the solar tracking system and the fixedposition panel. The findings were kept track of for 4 days, then tallied. The LDRs' output is determined by the amount of light falling on their surfaces. Arduino features a USB monitor that connects with the computer through USB and digital pins 0 and 1. Pins 0 and 1 can be utilized for digital output devices if these functionalities are used. The built-in serial monitor in the Arduino environment may be used to interact with the NodeMcu board. To gather the data, a model was implemented that allowed data from the LDRs to be collected each hour.



	observations on a bright sunny day											
Time	Array	LDR Outpu	ts for a cloudy	LDR outputs for a bright sunny								
(Hrs) PV	Output (V)	day		day								
		LDR1(V)	LDR2 (V)	LDR1(V)	LDR2 (V)							
0800	8.05	0.21	0.22	1.56	1.55							
0900	8.85	0.51	0.51	2.90	2.91							
1000	9.34	1.88	1.67	3.21	3.33							
1100	9.98	1.6	1.88	3.97	3.78							
1200	10.56	1.99	1.86	4.22	4.54							

Table 1(a) Photovoltaic array at sunny day ; (b) LDR observations on a cloudy day; (c) LDR observations on a bright sunny day

At the specified intervals, the values from the two LDRs must be read and recorded. The LDRs detect the intensity of light and so provide a reliable estimate of the amount of electricity that reaches the solar panel's area. The solar panel's power production is directly proportional to the wavelength.

Tables 1 show that the greatest sunshine occurs around midday, with maximum values between 1200 and 1400 hours. The amount of sunlight decreases in the mornings and afternoons evening, and the obtained results are lower than those obtained during the day. The monitoring system is turned off after sundown. In the morning, it is turned back on. The LDR readings for the screen equipped with the monitoring sector are designed to be near. It is because an error is created anytime they are in various locations, allowing it to move. When the readings are the same, the panel stops moving, indicating that the LDRs get the same amount of sunshine. Since the panel is in a static position, the numbers for the fixed panel fluctuate. The LDRs do not always face the sun at the same inclination. Except for midday, when they are nearly perpendicular to the direction. The days with the least amount of cloud cover have the maximum light intensity, so the LDR outputs will become the maximum. Whenever it's sunny, the monitoring system works best. It will be able to catch the majority of solar energy and turn it into electricity. In terms of solar panel power production for monitoring and stationary systems, it is obvious that the tracking system would produce more electricity. This is because the amount of energy created by solar cells is proportional to the amount of light available. The higher the luminous source, the more electricity the photovoltaic system will create.

5. CONCLUSION

Each day brings a new task of incubating anything new and unique, requiring energy to be the ultimate source of motivation for all the hard effort. It would be more appropriate to show that commercialization has spread its wings to a degree in the pursuit of money and power that we now find ourselves in a pool of severe ignorance of the planet's resource constraints, as a result of which the whole globe has been injured. Healing the planet is the foundation cultivation with which the hour clock is ringing, and this work provides the eye, as a result, to open the corridors of decreasing pollution in the storage of energy harvested from the Sun, as well as to crank up the speed of progress.



6. REFERENCES

- [1]. Ying-Tung Hsiao, China-Hong Chen, "Maximum Power Tracking for Photovoltaic Power System," Conference Record of the 2002 IEEE Industry Applications Conference. 37th IAS Annual Meeting, ISBN-0-7803-7420-7, 13-18 Oct 2002, pp 1035-1039.
- [2]. Pankajkumar, D. K. S., & Ponnusamy, D. P. S. Design and analysis the performance of real time Content delivery network using beam scanning. Journal Of Critical Reviews ISSN-2394-5125 VOL, 7.
- [3]. Sridharan, K., & Sivakumar, P. (2018). A systematic review on techniques of feature selection and classification for text mining. International Journal of Business Information Systems, 28(4), 504-518.
- [4]. Levent Bas, "Thin Film vs. Crystalline Silicon PV Modules," December 2011.
- [5]. M. A. Panait, T. Tudorache, "A Simple Neural Network Solar Tracker for Optimizing Conversion Efficiency in Off-Grid Solar Generators", International Conference on Renewable energies and Power quality (ICREPQ), march 12-14, Santander, 2008.
- [6]. Juan Reca-Cardeña, Rafael López-Luque, Chapter 9- Design Principles of Photovoltaic Irrigation Systems, "Advances in Renewable Energies and Power Technologies", Elsevier Science, 2018
- [7]. C. Chang, Chapter 5 Tracking solar collection technologies for solar heating and cooling systems, "Advances in Solar Heating and Cooling", Woodhead Publishing, 2016