TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA
ROMANIA

Faculty of Automation
and Computer Science

Arduino Solar Tracker

-Project prepared for the *Design with Microprocessors* Course-

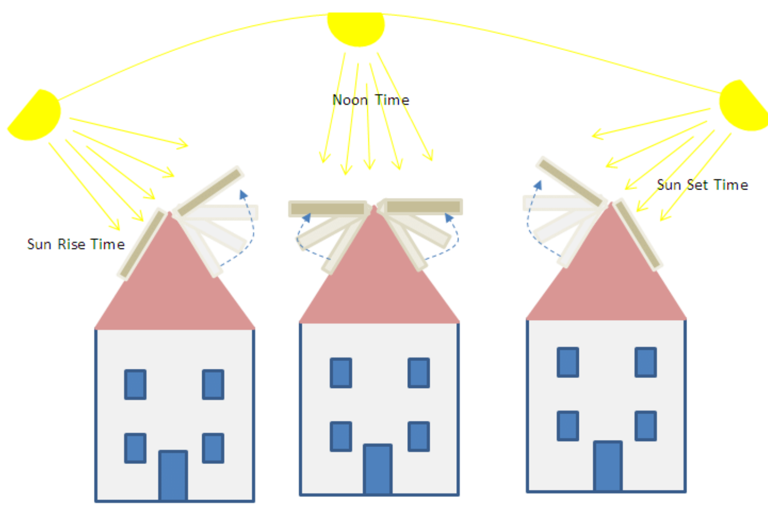
Student: Enoiu Diana, Keresztes Beáta, Fazakas Borbála
Supervisor: Radu Dănescu
Group 30432

2021-2022

Background. Problem Statement

Due to the growing population and more and more industrialized societies, the energy requirements of humankind have drastically increased in the past century. Unfortunately, the primary energy sources that we relied on so far (coal, gas, petroleum, …) are not sustainable: their usage contributes to climate change, and some of them are even available only in a limited amount.

One of the most promising alternative energy resources is the Sun: an unlimited, renewable, environment-friendly energy resource, at least at first sight. However, for converting the solar energy into electrical power, we need a. photovoltaic cells and b. appropriate weather, with a good amount of light falling on the solar panels. It is evident that solar panels represent an alternative worth to be considered only in the sunny environments, but even there, it is an extra challenge to position the solar panels such that they reach their maximum potential. Ideally, the sunlight should fall perpendicularly on the solar panel to produce the maximum amount of energy.



The Goal of the Project

The goal of our project is to implement a system which automatically repositions a solar panel, depending on the position of the Sun, such that the amount of energy produced is always maximum.

Design

We approached the problem from two different perspectives:

- **The Sensor Mode**: Firstly, we tried to come up with an algorithm that is based on the angle of the light rays falling on the solar panel. The objective of the algorithm is to position the panel such that the light rays are falling perpendicularly on it.
  - Measurements:

        - to decide from which direction the light rays are falling, we used 4 photoresistors to measure the light intensity at the 4 sides of the solar panel
    - Main concept:
        - the relative values of the light intensities define the direction into which the solar panel needs to be moved
    - Output:
        - the position of two motors, which control the position of the solar panel, are adjusted according to the light intensities. The process is then repeated again from the measurements part, and the panel's position is step-by-step adjusted to the optimal one.
- **The Search Mode**: Secondly, we developed an algorithm that works based on the voltage generated by the solar panel and tries to find the position where this voltage is maximal.
    - Measurements:
        - the voltage generated by the solar panel in multiple positions is measured
    - Main concept:
        - the position with the maximum voltage is selected as the optimal one
    - Output:
        - the position of two motors, which control the position of the solar panel, are adjusted to reach the previously found optimal position

The implementation

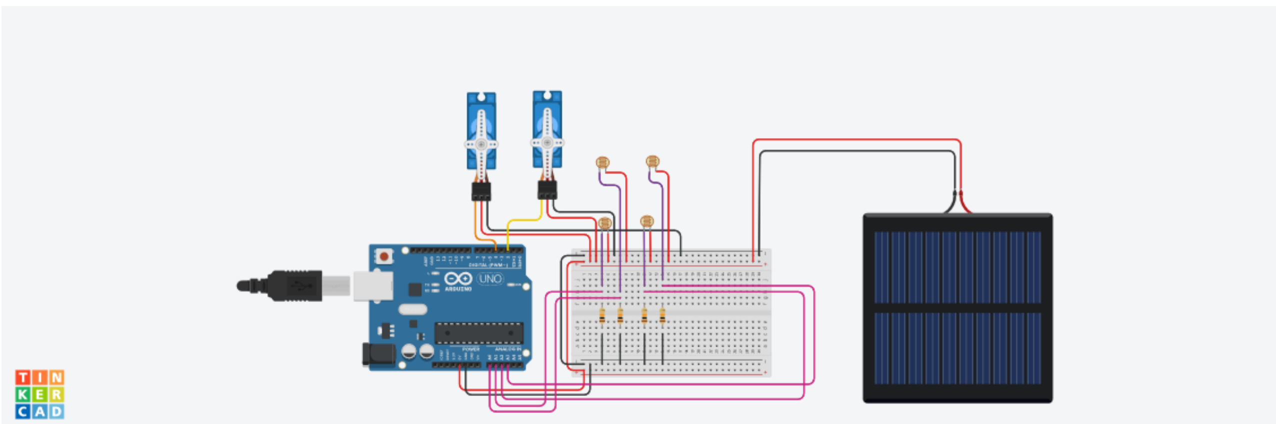
The Hardware

Components

- **Arduino Mega** microcontroller board
- **Servomotors** - https://cleste.ro/motor-servo-sg90-9g-180grade.html
    - Used to make the panel movement. They can rotate up to 180 degrees and they operate with a voltage of 4.8 ~ 6.0V.
    - Operating voltage: 4.8 ~ 6.0V
    - Current: 0.19A@5V, 0.24A@6V
    - Functioning speed: 0.12sec / 60 grade (4.8V) ~ 0.1 sec / 60 grade (6.0V)
    - Temperature interval: -30 ~ + 60 ℃
    - Dimension: 23X12.2X29MM
- **Photoresistors** - https://cleste.ro/fotorezistor-5528-ldr.html
    - They measure the light intensities that are used in the sensor mode.
    - Maximum voltage (V-dc): 150
    - Maximum power consumption (mW): 100
    - Temperature: - 30°C - +70°C
    - Spectral value (nm): 540
    - Light resistance (10Lux) (KΩ): 10 - 20

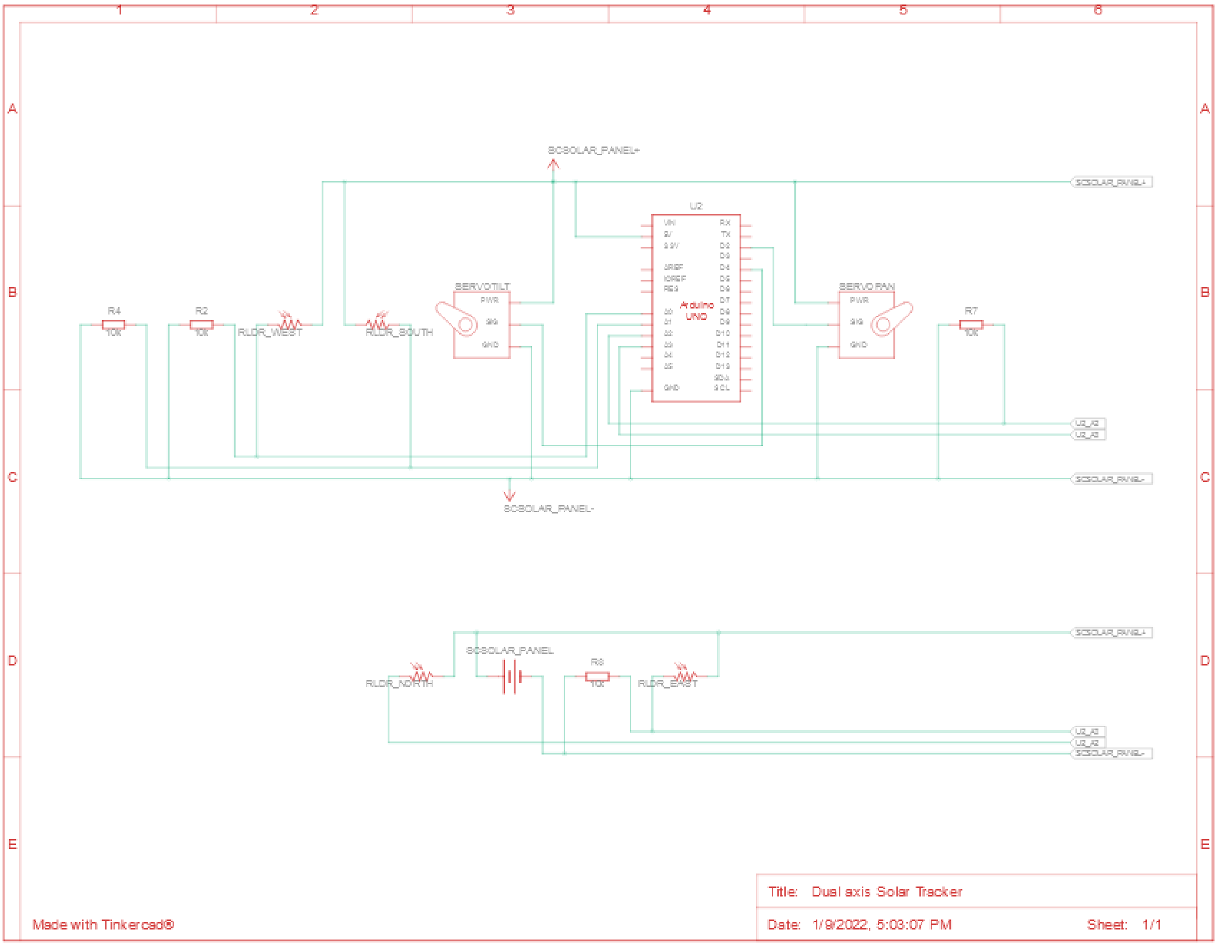
- **Solar panel** - https://cleste.ro/panou-solar-mini-5v-1-25-w-100mah.html
    - The component that we try to make receive as much light as possible from the sun. The voltage obtained by the solar panel is also used when performing the search mode. The panel generates up to 5V
    - Dimension 13.7cm x 8cm
    - Operating voltage: 5V
    - Operating current: 0-220mA
- **Tilt-pan support** - https://cleste.ro/suport-camera-cu-servomotor-pan-tilt.html
    - Helped us with the mechanical part of the tilt-pan movement.
- **Wires and resistors** of 10K ohms (for the measurements of light intensities)

Circuit design



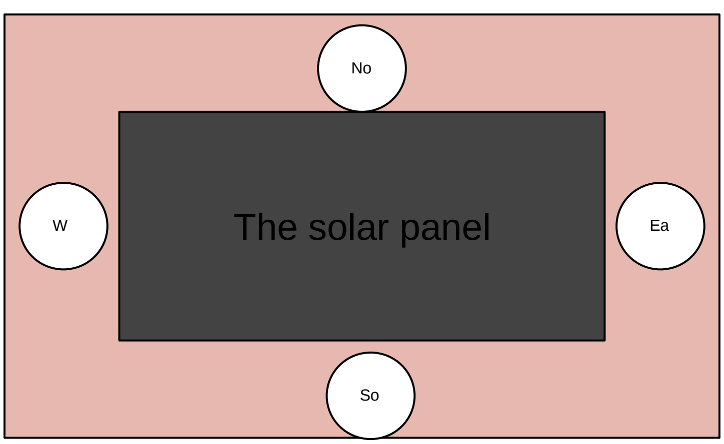
* The above circuit contains only the basic components for a functioning project. The complete diagram of the circuit with the components used for the interface, display and buttons, in order to interact with the user, can be found here: Interface .

Schematic view



The Software

Sensor mode



The sensor mode in fact represents a hill climbing algorithm, in which the position where the light rays are falling perpendicularly on the solar panel is found step-by-step, by taking the local optimum (i.e. the step that brings the angle closer to 90 degrees) at each step. Fortunately, since there is only one light source to be considered, the Sun, meaning that there are no local maximums, just one global maximum, this greedy algorithm takes us to the global maximum after all.

To decide which direction the next step needs to be taken, we placed 4 photoresistors around the solar panel, 1 to the North, one to the East, 1 to the South and one to the West, as can be seen on the above picture.

The main idea is that the light rays are falling perpendicularly on the panel if and only if $LightIntensity_{North}$ = $LightIntensity_{South}$ and $LightIntensity_{East}$=$LightIntensity_{West}$.

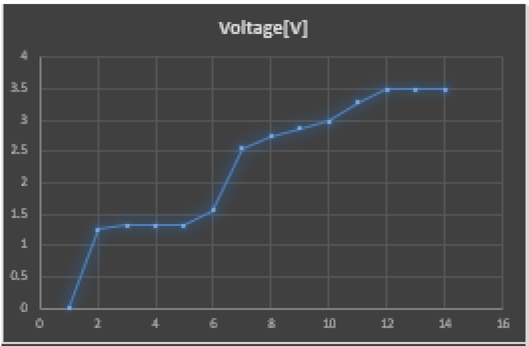
Furthermore, if this optimal state was not yet reached, then the next action needs to be taken based on the following rules:

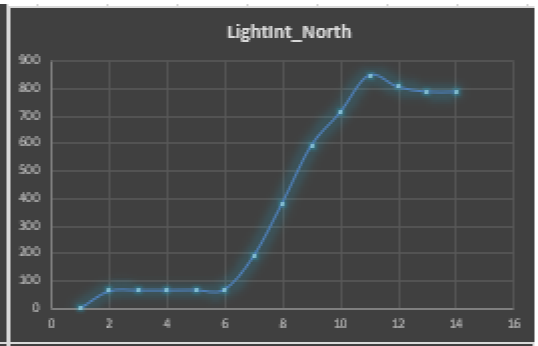
- if $LightIntensity_{North}$ > $LightIntensity_{South}$, then move the panel to the North (i.e. upwards)
- if $LightIntensity_{North}$ < $LightIntensity_{South}$, then move the panel to the South (i.e. downwards)
- if $LightIntensity_{East}$ > $LightIntensity_{West}$, then move the panel to the East (i.e.to the right)
- if $LightIntensity_{East}$ < LightIntensity, then move the panel to the South (i.e. downwards)

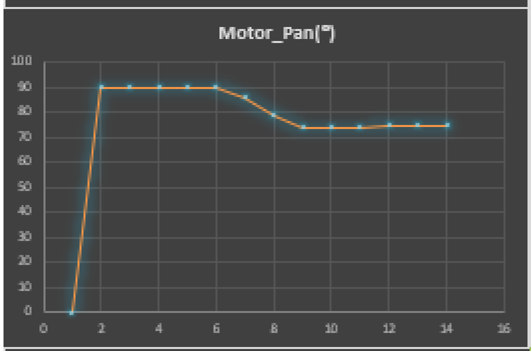
In practice, the perfect equality is almost impossible to reach, so we made the solar panel stop moving once the light intensities get close enough. Moreover, the magnitude of the step taken into a certain direction depends on the difference between the two light intensities causing that step: the higher the difference between the two of them, the larger the step should be.

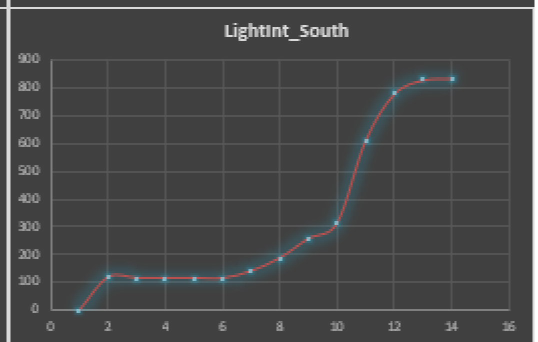
Demo

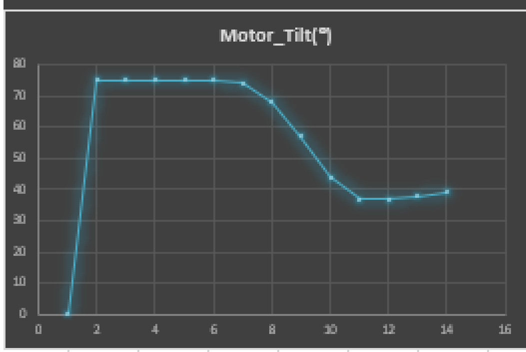
For an example of how the Sensor Mode works, you may refer to the below figure (see the Data Acquisition section for a better understanding of how these plots were generated):

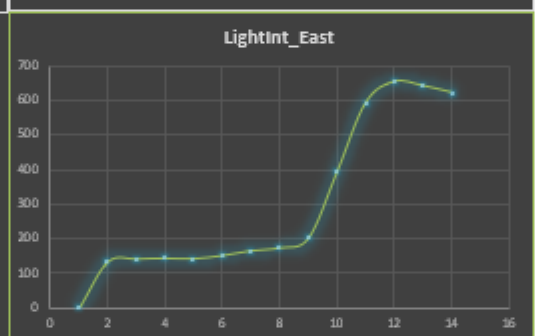












Disadvantages of the Sensor Mode:

As previously stated, the perfect equality between the corresponding light intensities can never be achieved. This makes it hard to detect the case when the motors should not be moved anymore. However, if the motors reposition themselves too often, then the efficiency of the model will decrease, as the motors use a significant amount of energy. As a workaround, we implemented the Search Mode too. For a perfect, efficient and ecological working mode, the Sensor and the Search modes should be used in turn.

Search mode

The second mode that the solar panel uses to find the sun is Search Mode. This mode has two searches – broad search and restricted search.

Broad search measures the voltage value obtained by the solar panel in any direction and then it positions the solar panel to where the maximum voltage value was obtained. To position the panel in almost any possible direction, we first tilt the panel up (at 90 degrees from the ground) and then pan the panel from left to right, slowly incrementing the rotation angle that is sent to the servomotor. Once the left to the right movement is done, we will increment the tilt angle and then perform a right to the left movement (to avoid going back again to the left position, wasting energy and time). We should be able at the end of this algorithm to find an approximate position to where the maximum voltage value is and position the solar panel there.

After a broad search is completed, we will not use it anymore. Since we can assume that the sun does not change its position by a lot as time passes, we will use the restricted search. Restricted search does almost the same thing as broad search, but it does not go in any possible direction. Instead, it goes around the position where the maximum voltage was found to adjust the panel on small changes in the position of the sun. We can use restricted search from hour to an hour for example, but to exemplify the working of this search algorithm we will use it every 30 seconds.

The searching mode attempts to be a more economic mode compared to sensor mode. In sensor mode, the servo motors may continue to adjust the panel even after the maximum voltage position was found because the photoresistors will still measure changes in the position of the sun if it is cloud outside for example. Those small adjustments do not help increase the voltage obtained by the panel, but they waste some energy.

Data acquisition

For visualizing the obtained data from the photoresistors and the solar panel, a Microsoft Excel extension tool was used, the PLX-DAQ, Parallax Data Acquisition Tool, which allows real-time monitoring of the read data and rendering them to a graph.

It allows communication with the Arduino board using the Serial interface. The Arduino board sends through the Serial line, on port 9600, the format of the columns as well as the actual data.

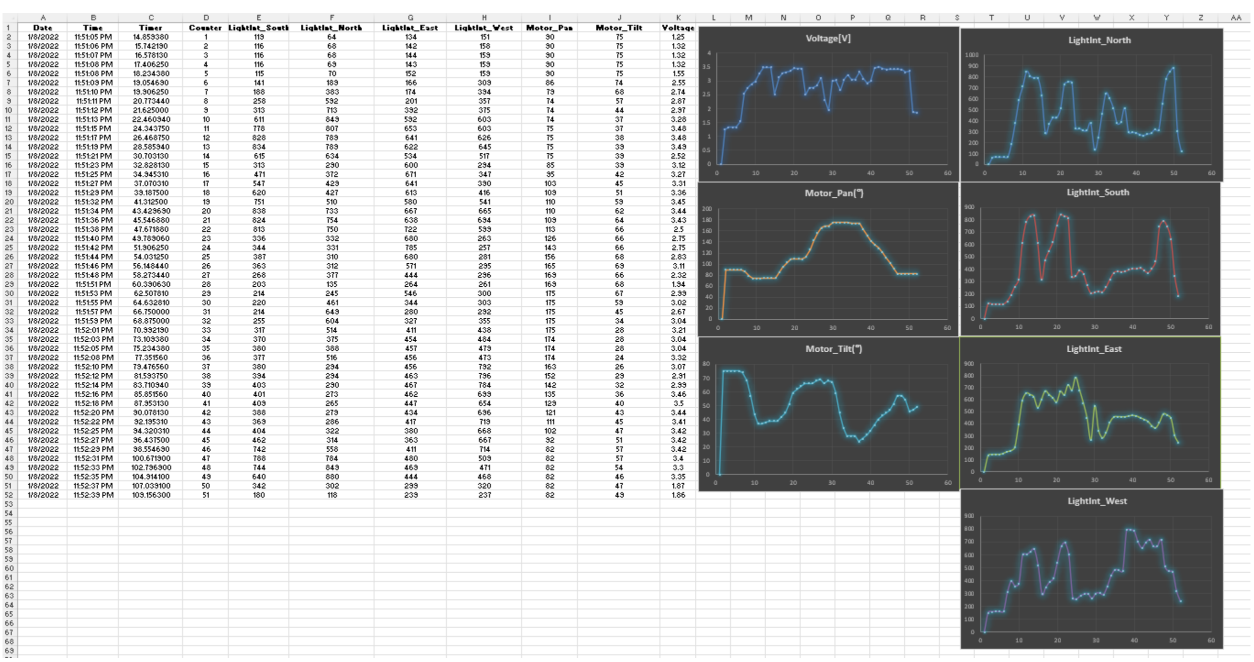
Seven charts were added for monitoring real-time the read values:

· Voltage: measured by the solar panel over time

· Motor_Pan and Motor_Tilt: which show the angles of the 2 motors for their current positions

· LightInt_North, LightInt_South, LightInt_East and LightInt_West: which illustrate how the intensity of the light changes depending on the orientation of the panel (towards the light or away from it), also they indicate the current direction of the light source (where the max value is measured).

An example of the collected data in the Sensor modes (from the 4 photoresistors and the solar panel):



Date	Time	Timer	Counter	LightInt_South	LightInt_North	LightInt_East	LightInt_West	Motor_Pan	Motor_Tilt	Voltage
1/8/2022	11:51:05 PM	14.859380	1	119	64	134	151	90	75	1.25
1/8/2022	11:51:06 PM	15.742190	2	116	68	142	158	90	75	1.32
1/8/2022	11:51:07 PM	16.578130	3	116	68	144	159	90	75	1.32
1/8/2022	11:51:08 PM	17.406250	4	116	69	143	159	90	75	1.32
1/8/2022	11:51:08 PM	18.234380	5	115	70	152	159	90	75	1.55
1/8/2022	11:51:09 PM	19.054690	6	141	189	166	309	86	74	2.55
1/8/2022	11:51:10 PM	19.906250	7	188	383	174	394	79	68	2.74
1/8/2022	11:51:11 PM	20.773440	8	258	592	201	357	74	57	2.87
1/8/2022	11:51:12 PM	21.625000	9	313	713	392	375	74	44	2.97
1/8/2022	11:51:13 PM	22.460940	10	611	849	592	603	74	37	3.28
1/8/2022	11:51:15 PM	24.343750	11	778	807	653	603	75	37	3.48
1/8/2022	11:51:17 PM	26.468750	12	828	789	641	626	75	38	3.48
1/8/2022	11:51:19 PM	28.585940	13	834	789	622	645	75	39	3.49
1/8/2022	11:51:21 PM	30.703130	14	615	634	534	517	75	39	2.52
1/8/2022	11:51:23 PM	32.828130	15	313	290	600	294	85	39	3.12
1/8/2022	11:51:25 PM	34.945310	16	471	372	671	347	95	42	3.27
1/8/2022	11:51:27 PM	37.070310	17	547	429	641	390	103	45	3.31
1/8/2022	11:51:29 PM	39.187500	18	620	427	613	416	109	51	3.36
1/8/2022	11:51:32 PM	41.312500	19	751	510	580	541	110	59	3.45
1/8/2022	11:51:34 PM	43.429690	20	838	733	667	665	110	62	3.44
1/8/2022	11:51:36 PM	45.546880	21	824	754	638	694	109	64	3.43
1/8/2022	11:51:38 PM	47.671880	22	813	750	722	599	113	66	2.5
1/8/2022	11:51:40 PM	49.789060	23	336	332	680	263	126	66	2.75
1/8/2022	11:51:42 PM	51.906250	24	344	331	785	257	143	66	2.75
1/8/2022	11:51:44 PM	54.031250	25	387	310	680	281	156	68	2.83
1/8/2022	11:51:46 PM	56.148440	26	363	312	571	295	165	69	3.11
1/8/2022	11:51:48 PM	58.273440	27	268	377	444	296	169	66	2.32
1/8/2022	11:51:51 PM	60.390630	28	203	135	264	261	169	68	1.94
1/8/2022	11:51:53 PM	62.507810	29	214	245	546	300	175	67	2.99
1/8/2022	11:51:55 PM	64.632810	30	220	461	344	303	175	59	3.02
1/8/2022	11:51:57 PM	66.750000	31	214	649	280	292	175	45	2.67
1/8/2022	11:51:59 PM	68.875000	32	255	604	327	355	175	34	3.04
1/8/2022	11:52:01 PM	70.992190	33	317	514	411	438	175	28	3.21
1/8/2022	11:52:03 PM	73.109380	34	370	375	454	484	174	28	3.04
1/8/2022	11:52:05 PM	75.234380	35	380	388	457	479	174	28	3.04
1/8/2022	11:52:08 PM	77.351560	36	377	516	456	473	174	24	3.32
1/8/2022	11:52:10 PM	79.476560	37	380	294	456	792	163	26	3.07
1/8/2022	11:52:12 PM	81.593750	38	394	294	463	796	152	29	2.91
1/8/2022	11:52:14 PM	83.710940	39	403	290	467	784	142	32	2.99
1/8/2022	11:52:16 PM	85.851560	40	401	273	462	699	135	36	3.46
1/8/2022	11:52:18 PM	87.953130	41	409	265	447	654	129	40	3.5
1/8/2022	11:52:20 PM	90.078130	42	388	279	434	696	121	43	3.44
1/8/2022	11:52:22 PM	92.195310	43	369	286	417	719	111	45	3.41
1/8/2022	11:52:25 PM	94.320310	44	404	322	380	668	102	47	3.42
1/8/2022	11:52:27 PM	96.437500	45	462	314	363	667	92	51	3.42
1/8/2022	11:52:29 PM	98.554690	46	742	558	411	714	82	57	3.42
1/8/2022	11:52:31 PM	100.671900	47	788	784	480	509	82	57	3.4
1/8/2022	11:52:33 PM	102.796900	48	744	849	469	471	82	54	3.3
1/8/2022	11:52:35 PM	104.914100	49	640	880	444	468	82	46	3.35
1/8/2022	11:52:37 PM	107.039100	50	342	302	299	320	82	47	1.87
1/8/2022	11:52:39 PM	109.156300	51	180	118	239	237	82	49	1.86

As the data comes from the Arduino board, through the Serial interface, the table is filled with the new data, and the plots automatically refresh showing the evolution of the measured quantities over time.

This allows us to visualize the recorded data and validate the correctness of our implementation, as well as it stores the received values in an excel file.

Interface

To further extend the application, a user interface could be added to control the solar panel and to display its current working mode.

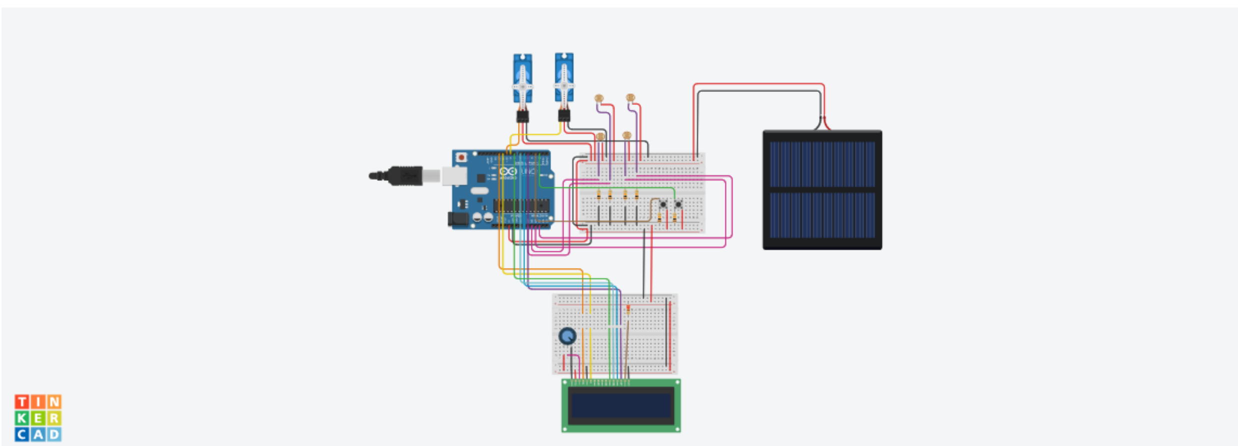
The interface consists of:

- LCD display: which displays the current functioning mode of the solar tracker, along with a small icon which represents the mode. These icons were created using custom characters.
- Buttons: a pair of buttons which allow switching from one working mode to another.

In order to instantly react to a button press, two interrupts were attached to the button pins, so that the corresponding interrupt handlers would be called to switch the mode. On such an event, the display is updated, showing the currently selected mode.

The interrupts attached to the buttons were necessary, because the main loop spends a lot of time and it is delayed considerably during the search mode, when it maps the space in all directions. Therefore, it might not react to a button press.

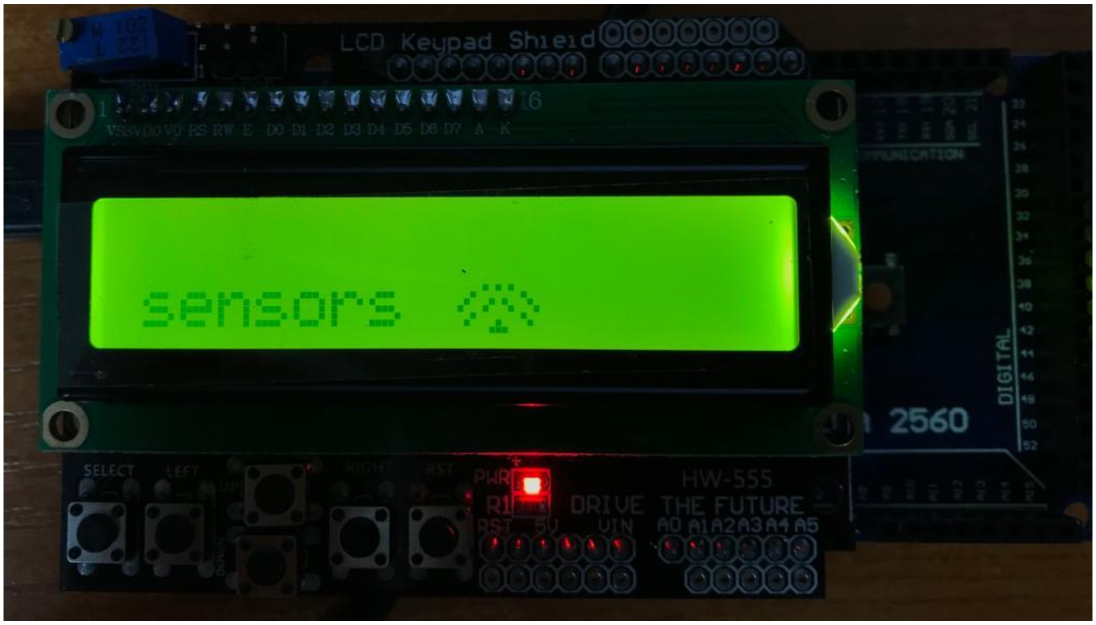
The design of the circuit is changed the following way after adding the 2 buttons and also the LCD display:



The board used in our project is an arduino Mega 2560, which has much more pins than an Uno board, it also has more external interrupt pins. However, in order to illustrate the schematic of the circuit on a simple Uno board, we can only attach external interrupts to two pins, namely 2 and 3. Therefore, the pins used for controlling the motors will be changed to pins 9 and 10, which are also pwm pins. An LCD display is attached to the circuit on the remaining pins.

In the project's design we used an LCD shield, which was mounted directly on the pins of the arduino Mega board, instead of having to connect it with additional wires. The animations shown in each of the modes are the following:

- In the sensor mode:



- In the search mode: