SASTA OLIPHANT SCIENCE AWARDS

South Australian Science Teachers Association

Highly Commended

Programming, Apps & Robotics

Year 5-6

Bradley Ayles

Heathfield Primary School

Rowe Scientific Pty. Ltd. www.rowe.com.au

Australian Government Department of Defence

Government of South Australia Department for Education

SASTA South Australian Science Teachers Association

Measurement of Solar Energy for Control of Loads

By Bradley Ayles

Why I chose this as a project:

I like to work with electrical circuits (making my own) and am interested in solar energy, my Dad is an electrician and he works a bit with solar energy for houses and pumps. I have a toy where a small solar cell runs a small motor that makes a person on a bike turn the pedals. I really like Robotics and I've been attending Robotics classes using the Arduino. I have an Arduino board and the Arduino program on a computer that I use to program different things. I recently used an infrared receiver and speaker to make a noise whenever the TV remote control button is pressed.

I know that things like pumps need to be controlled so that they don't break and I wanted to look into the control program and make one of my own.

What I did to make the project:

I already had the Arduino board, a solar cell, motor and load through the person pedaling a bike.

I needed to add in a load resister and 5Volt control relay. I also used wires from my electricity kit and soldering iron to attach everything together.

I first measured the exact voltage the motor needed to make the bike pedals turn. I did this by using my multi meter and a light shining slowly onto the solar panel, making it brighter and brighter. I saw that the multimeter read 0.15Volts the exact moment before the motor started turning and then the instant the motor started turning it jumped to 0.3Volts keeping the motor turning.

I already knew the Voltage, but needed to work out the Current in Amps. To do this, I put the multi meter in series with the motor then shone a light slowly onto the solar panel to almost start the motor and recorded the Current shown as 40 Milliamps. Then I did the same thing to measure when the motor was running and found it to be 30 Milliamps.

My Dad helped me to work out the resistance using Ohm's Law where current going through a conductor between two points is directly related to the voltage across the two points. This is called the Resistance. I had to convert the Milliamp reading to Amps (1 Amp = 1000 mA).

To work out the load resistance setting, I used Ohm's law formula of:

V (Voltage) = I (Current) x R (Resistance).

This was calculated by:

V = I x R

then

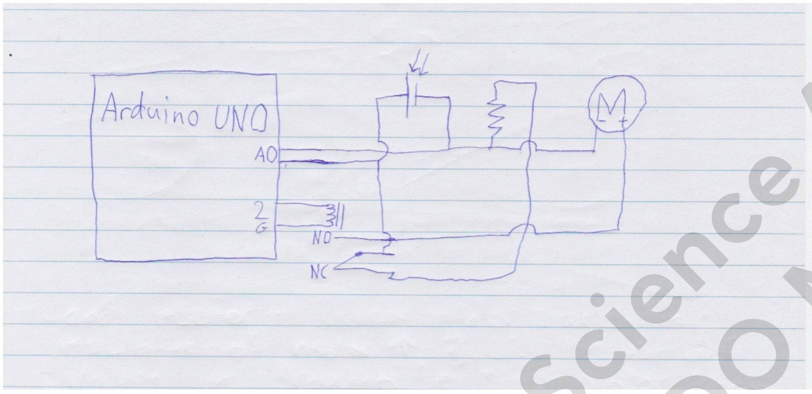
V ÷ I = R

0.15 ÷ 0.04 = 3.75Ω (Ohms)

0.3 ÷ 0.03 = 10Ω (Ohms)

My Dad helped me make a variable load resistor using some resistors and a potentiometer all in parallel. I could turn the knob on the potentiometer and see the resistance changing on my multimeter. I set the resistance to 3.75 Ohms. I only needed the 3.75 Ohm resistor because once the relay clicks on and the motor is running, the motor acts as the load. When the motor is not stalled and is actually running the affective motor resistance is the 10 Ohms I calculated. The difference between when motor is stalled and running is why I needed the two different numbers for my Arduino program, because one number is needed to know when to turn the relay on. And the other number is needed to know when the turn the relay off so that the motor stops before it stalls. It's very important that the motor is only connected to the solar panel power source when there is enough energy available for it to actually turn. If it's connected when there's not enough energy for it to turn in real life it can cause the motors to burn out.

I then drew the electronic circuit.



I used my soldering iron to join the wires together between all the electronic components (Arduino board, relay, motor, load resister and solar panel).

I then used the Arduino program on the computer using Ardublock to make the program.

I used the Analogue input (A0) to monitor the voltage across the load resistor. The analogue inputs of the Arduino are a 0-5 Volt input where 0 Volts is equal to 0 and 5 Volts is equal to 1023. I had to work out the Analogue figure that I needed to turn the relay on. So, knowing that 0.15Volts is the amount required to start the motor and 0.3Volts is required to run the motor, I used a formula to work out the amount I need to put into the Ardublock code for the program.

(1023 ÷ 5) X 0.15 = 30.69 (rounded up to 31)

(1023 ÷ 5) X 0.3 = 61.38 (rounded down to 61)

I tested these amounts by uploading the program to see if it worked, but the motor was stalling when the relay clicked on, therefore there was not enough solar energy when the relay turned on to drive the motor. I then adjusted the Analogue input numbers I used in the program until I found that the motor would start successfully every time when I used 42 as the analogue input level and the motor would continue to run down to 70. Any time below 70 was used, the motor would sometimes stall in low light before the relay cut the motor off.

Aim of the entry and its scientific purpose and potential applications:

To show how solar energy can be measured and then used to control different types of loads.

An example of this is the solar pump on our farm. When there is enough solar energy the pump will run, but if there isn't enough solar energy (such as when the sun is setting), the pump will not run. The pump controller allows the pump to only run when there is enough energy to safely do so. This is because you can't run a water pump when there is not enough solar energy, because the pump motor won't turn but it will have power from the solar panels on it and it will get very hot and can burn out. The solar panel can only be connected to the motor when there is enough sunlight that the pump can actually turn.

Another example is on a house that has a solar system. When there's more energy from the solar panels that what is being used, the controller can switch on the hot water heater to make use of the available energy rather than this energy being wasted.

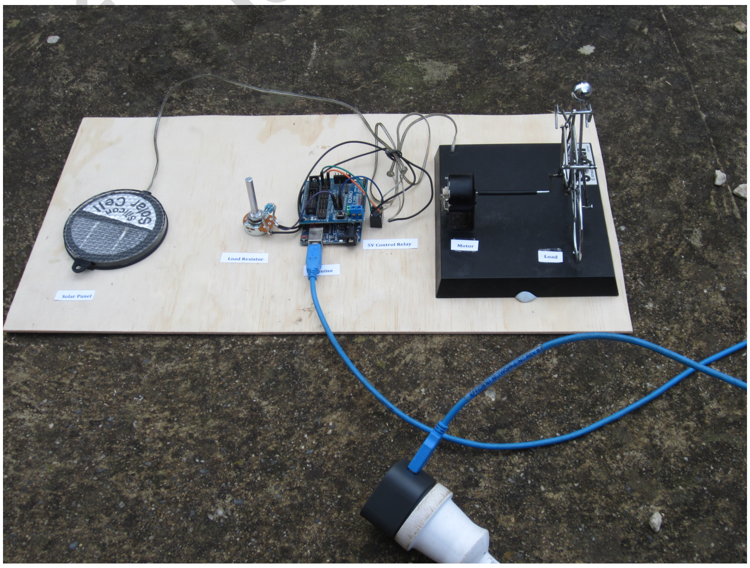
When sun rises the solar energy gradually increases. The motor should not be connected to the power source when there is not enough energy to get it to turn. It should only be connected once there is enough energy to make it turn immediately. When the sun sets the solar energy gradually decreased. The motor needs to be disconnected from the power source while there is still just enough energy to keep it turning, but not too little to make the motor stall. This stops the motor from getting too hot by being connected to power but not turning. I have shown this concept using the Arduino to measure the available solar energy and using the Arduino to control the relay to switch the motor on and off. The person riding the bike represents the load which could be either a pump in a well or pool or a hot water system etc.

The type of robot or computer/device required to run the program:

The type of device is the Arduino and the program I used to program the Arduino is called Ardublock.

Clear instructions on loading or using the entry

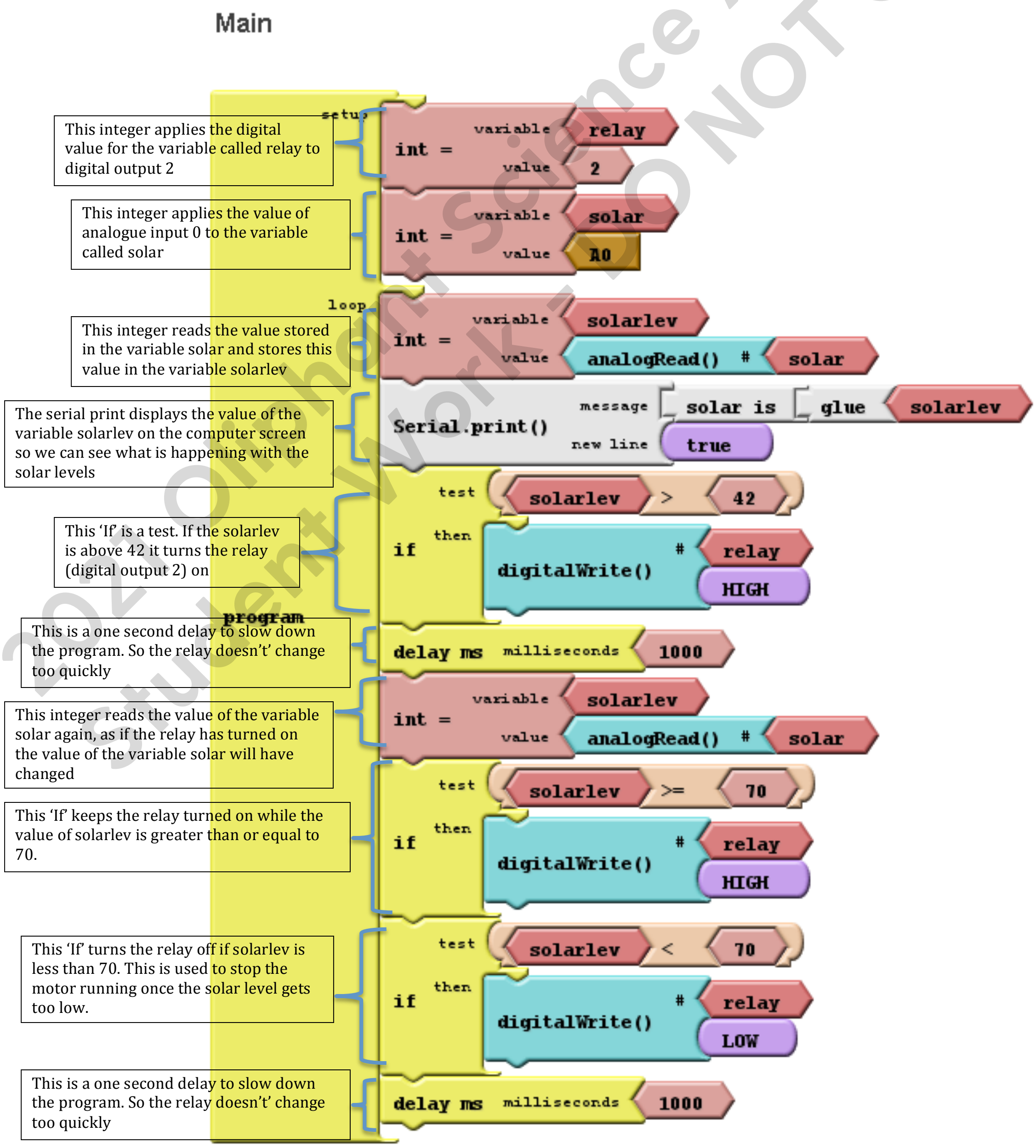
You need to plug a USB cable into the port on the Arduino and the other end to a USB socket or power adaptor to provide a power source. This is to provide power to the Arduino board and run the program. See picture below:



You also need a strong light to simulate the sun (or use the sun). I used the sun and slowly moved a piece of cardboard across the solar panel to simulate sunlight coming and going. I could hear the click of the relay and saw the bike pedals immediately moving. If the sun is not strong enough (behind a cloud), you can use a strong light. I could do this using a spot light.

I have demonstrated how it all works in the video link (this is a public link, but has also been shared with jennah@sasta.asn.au):
https://drive.google.com/file/d/1AZYP73_AtdrCr5w8TUE_ueWbSFEMCocY/view?usp=sharing

A hard copy of the program and an explanation of what the sections of the program do:



Acknowledgement of any external support provided to the entry:

My Dad helped me to work out the resistor values. My Dad explained Ohm's law (Voltage = Current X Resistance) that I used to calculate the needed value of the load resistance. He also brought me the resistors and potentiometer that I made up into the load resistor.

I did everything else myself, including drawing the wiring diagram, soldering the wires together to each electric component, using my own multimeter to measure the required voltages and currents and writing the program using the analogue input figures I had calculated.

Bibliography

https://en.wikipedia.org/wiki/Ohm%27s_law

https://www.solarquotes.com.au/blog/using-fronius-inverter-smart-meter-relay-make-solar-electric-hot-water/