Automated Solar Hot Water Power Shower using Black Plastic Pipes

by **Tecwyn Twmffat** on June 15, 2016

Table of Contents


Author:Tecwyn Twmffat Goat Industries
I live on an island in the Irish sea called Ynys Mon which was once inhabited by the Romans, the Vikings and is still inhabited by Druids. Me, I'm just a bloke who likes inventing things and doing the whole 'Alternative' thing. Most rewarding experience: Playing music to large audiences at festivals. Most dangerous experience: Mixing psychoactive drugs with Buddhist meditation. Most difficult experience: Trying to work out what the hell I'm here for! Plans for the future: Would like to build a 'Passiv Haus'. Best advice: Get off the well trodden paths.

Intro: Automated Solar Hot Water Power Shower using Black Plastic Pipes

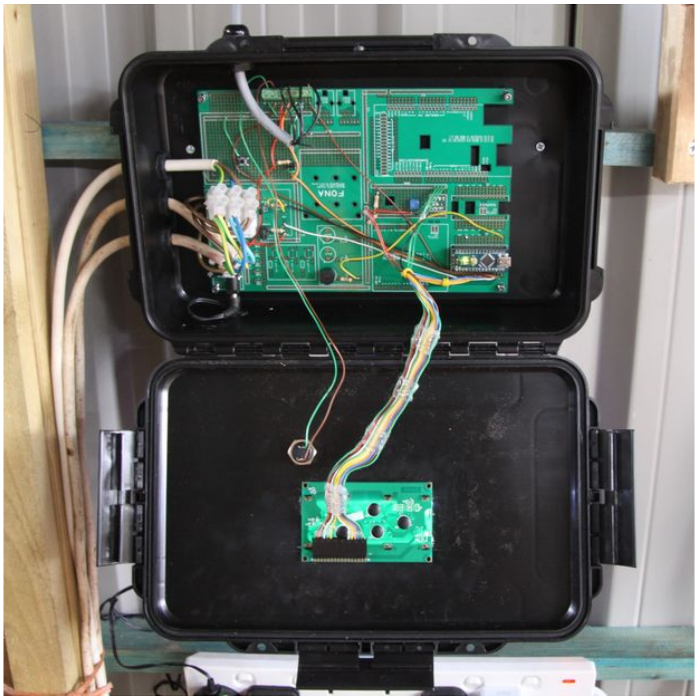
Black 50mm pipes filled with water are arranged in a herring bone pattern on a roof to pick up heat from the sun for a nice warm shower. The features of this design are as follows:

- Batch process so never goes cold
- Easy to build
- Self draining - no frost damage in winter
- No complicated plumbing
- Durable and robust against wind damage
- Non destructive tappings for level sensors and thermometers
- No special tools or skills required
- 30 litres gives 5 minutes of power shower

The pipes are mounted on a roof facing to the south west with an elevation of about 30 degrees and supply a 350 watt electric pump for a proper 'power' shower.

Heating water is a great way of using solar energy and water needs lot of energy to make it hotter and we all need a shower every now and again, right?

If this system takes 2 hours to warm up from 15 degrees C to 35 degrees C then, because it contains 30 litres of water, the power works out as 350 watts.



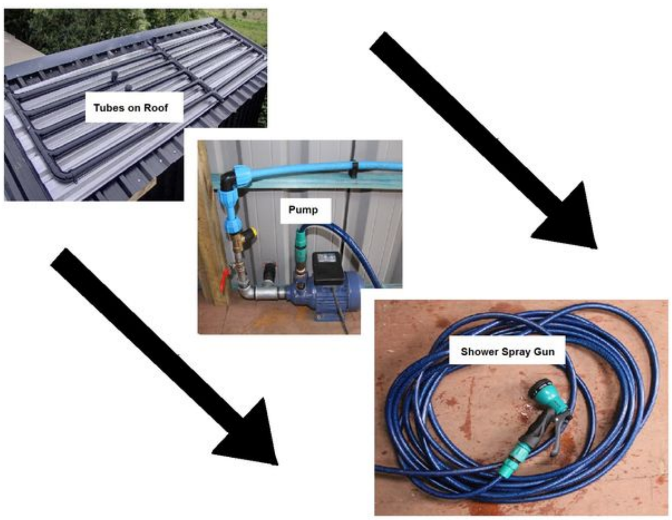
Step 1: How it Works

The main secret behind this design is the use of waste water pipes and fittings to get the pipes to run downwards at 2 degrees to the horizontal into a central spine. The whole system needs to be self draining so that every last drop of warm water comes down into the pump at a good rate of flow so that the pump is adequately supplied.

It might be tempting to use pipe of a smaller diameter to catch more heat from the sun, but this might cause the pump inlet to be restricted and the pump output pressure to be significantly reduced and it would not then be a proper 'power' shower. Also, smaller diameter pipe would mean less water storage unless very long lengths were used, which would make self draining very tricky.

Self draining is critical as we don't want to introduce cold water into the system until the person has finished showering - there's nothing worse than a nice warm shower suddenly going cold! Also, in the winter the shower must be completely free of water or the frost will damage the pipes.

This system is fully automated, which means that the pipes are automatically refilled with fresh water after the 'dirty person' has finished showering.



Step 2: Materials

Steel box profile sheet, 3.5m x 1.1m 1

1/2" BARREL NIPPLE, MILD STEEL, GALVD 6

6469 1" BARREL NIPPLE, MILD STEEL, GALVD 1

8088 1" x 1/2" BSP HEX BUSH, GALV 2

8243 1/2" BSP EQUAL TEE, GALV 1

8291 1" BSP EQUAL TEE, GALV 1

8498 1" BSP M x 1" BSP F, ELBOW, GALV 1

9689 1/2" BSP, BALL VALVE, LEVER, FULL BORE, PLATED 2

9865 20 mm MDPE x 1/2" BSP TM (BS 21), ADAPTOR 2

9870 20 mm COMP, MDPE ELBOW 6

9883 20 mm MDPE, EQUAL TEE 1

9922 20 mm, 12 bar PLASTIC PIPE STIFFENER 16

11796 32 mm KNUCKLE BEND, BLACK, SOLVENT WELD 2

15819 20 mm x 1/2" PP BLACK COBRA PIPE CLIP 10

18176 7 dial spray gun with cushion grip 1

18185 1/2" - 3/4" adaptor and reducer 1

19802 3/4" 2

9697 1/2" BSP, WATER SOLENOID VALVE, DIRECT... 1

11114 32 mm PIPE CLIP, WHITE 4

11133 32 mm x 2 m, WASTE PIPE, WHITE 1

11137 32 mm KNUCKLE BEND 90°, UNIVERSAL COMP 1

11778 32 mm x 3 m, WASTE PIPE, BLACK, SOLVENT WELD 1

11784 50 mm x 3 m, WASTE PIPE, BLACK, SOLVENT WELD 3

11820 50 mm SWEPT TEE, BLACK, SOLVENT WELD 16

11847 50 mm STRAIGHT COUPLING, BLACK, SOLVENT WELD 5

11865 50 mm SCRWD ACCESS PLUG, BLACK, SOLVENT WELD 4

11883 32 mm PIPE CLIP, BLACK 6

11889 50 mm PIPE CLIP, BLACK 25

11901 50 mm CROSS TEE, BLACK, SOLVENT WELD 4

17570 Multi-fit waste bend, 32 mm grey 1

11784 50 mm x 3 m, WASTE PIPE, BLACK, SOLVENT WELD 3

11793 50 mm SWEPT BEND, BLACK, SOLVENT WELD 4

11820 50 mm SWEPT TEE, BLACK, SOLVENT WELD 4

11865 50 mm SCRWD ACCESS PLUG, BLACK, SOLVENT WELD 1

11868 32 mm THREADED COUPLING, BLACK, SOLVENT WELD 1

11877 50 mm x 32 mm, REDUCER, BLACK, SOLVENT WELD 1

11889 50 mm PIPE CLIP, BLACK 8

11901 50 mm CROSS TEE, BLACK, SOLVENT WELD 1

11922 32 mm ADAPT0R M IRON, BLACK, SOLVENT WELD 1

22007 250 ml 1

Reflective paint 2.5 litre 1

Pump, TAM105 1

Most of the fittings were obtained from a company called BES in the UK.

Step 3: Layout the pipes

The 50mm pipes are cut into the following pieces:

- 1585mm x 1of
- 755mm x 8 of
- 1510mm x 5 of
- 60mm x 3 of
- 115mm x 8 of

.. and laid out on the box section sheet on the ground and assembled without the cement to check that all the pieces fit together.

Notice that most of the fittings are 92 degrees and not 90 as they are designed to produce waste water drainage running at 2 degrees to the horizontal, which is perfect for our requirements. Also, the box profile sheet is used UPSIDE DOWN as this gives more flat surfaces to attach the plastic clips to clamp the structure down.

Step 4: Central Spine

All the 'horizontal' pipes slot slightly downwards into the central spine at 2 degrees.

Disassemble all of the pipes and reassemble the central spine, taking care to keep all the other pipes as near to their original positions as possible to prevent getting them mixed up with one another. Some of the fittings that I used were a bit loose and required extra cement to glue them together. Before gluing, the surfaces need to be clean and sanded to give a rough finish before coating generously with cement , pressing together and giving a 90 degrees twist. It says on the tin that it is cured in ten minutes, but I found that this was not true and the joints needed at least one hour at 15 degrees C to cure before doing the next lot of pipes.

Step 5: 'Horizontal' Sections

Though not truly horizontal, these four sections are fitted with inspection caps to house level sensors at a later stage. I positioned the caps vertically as it made them easier to get to for the future.

Step 6: End Fittings

It's important to do the assembly in the right order so that the final parts can be slotted in and glued 'double ended' without too much difficulty. The small sections of pipe in the photo will be the very last bits to glue in place.

Step 7: Final assembly

After a few hours, with breaks between each of the major stages,the whole structure can be laid out on the upside down steel sheet. It is positions carefully so that all the 'horizontal' pipes are at 2 degrees to the horizontal and all the vertical pipes are dead on vertical. The final array of pipes forms a really smart and tidy herring bone pattern, with most of the drainage going to the central spine and the bottom most fitting, exactly in the optimum point for feeding the pump for the power shower.

Step 8: Testing

Now is the time to test the shower and check for leeks. Make a temporary attachment to the pump and fill the whole structure with water and look for drips and dribbles. If the sun is shining, the water will get nice and warm after about 2 hours.

Leaks can be fixed with some pipe cement and jute string.

Step 9: **Clamping the Pipe Array to the Sheet**

With the pipe array in exactly the right position on the sheet, the clamps can be screwed on using self tapping screws. A couple of cordless drills is very handy for this job, one for drilling and the second for screwing in.

Step 10: Painting the Sheet

All the clamps are now unscrewed again and everything is removed and the sheet brushed down. It is then painted with aluminium reflective paint to get a little bit extra sun light on the black pipes through reflection. It is important to be continually stirring the paint pot during the painting process.

Step 11: Assembling on the Roof

Firstly, the steel roofing sheet is screwed onto the roof with self drilling and self sealing screws especially designed for this purpose. The sheet must be exactly horizontal to get the best possible drainage into the central spine.

Step 12: Screw the Pipes onto the Sheet

The pipes are then screwed back onto the sheet.

Step 13: Plumbing the Downward Run

The main outlet from the pipe array has a reducer fitted to 32mm, which then forms the start of the feed line to the pump. The first fitting - the white one - is a standard 90 degree compression bend. Notice how it runs over the gutter clip, allowing plenty of space for the gutter, which is not yet in place.

The third photo shows how the 30 degree roof slope is resolved into true vertical. A special fitting is used, shown in the last photo above, called a 'universal knuckle bend', which is a compression fitting that can be positioned in any angle from 90 to 180 degrees - very useful!




Step 14: Plumb in the Pump

The mains water comes in from the blue pipe, through a 90 degree bend, through a 1/2" BSP solenoid valve, through a 1/2" isolation valve, into a 1" to 1/2" reducer, into a 1" male female bend, into a 1" 'T' fitting and into the inlet of the pump via a 1" barrel nipple. The black fitting going through the wall goes to the array of pipes on the roof for the warm water.

The outlet from the pump goes vertically upwards and has a 1" to 3/4" reducer which goes to the garden hose via a 3/4" barrel nipple. The garden hose has a hand operated spray gun for the actual shower head.

Step 15: Automation

Parts:

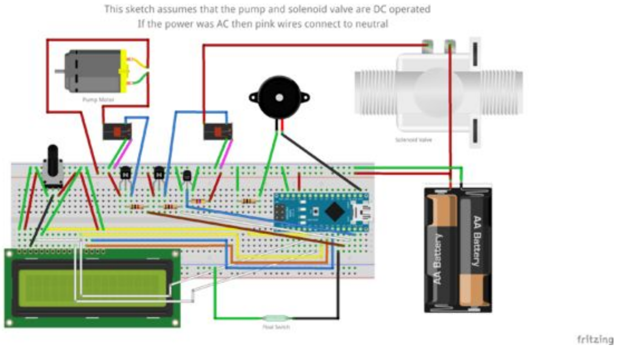
1. DS1 DS18B20 1-Wire Temperature Sensor part # DS18B20
2. J1 Piezo Speaker
3. K1 Relay package THT; switching circuit SPDT; voltage 5V; contact rating 125VAC / 30VDC @ 1 AMP; part # FRS1B-S
4. K2 Relay package THT; switching circuit SPDT; voltage 5V; contact rating 125VAC / 30VDC @ 1 AMP; part # FRS1B-S
5. LCD1 LCD screen type Character; pins 16
6. M1 DC Motor
7. Part1 Arduino Nano (Rev3.0) type Arduino Nano (3.0)
8. Q1 NPN-Transistor package TO92 [THT]; type NPN (EBC)
9. Q2 NPN-Transistor package TO92 [THT]; type NPN (EBC)
10. R1 Rotary Potentiometer (Small) size Rotary - 9mm; package THT; maximum resistance 100k?; track Linear; type Rotary Shaft Potentiometer
11. R2 100? Resistor bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 100?
12. R3 1k? Resistor bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 1k?
13. R4 4.7k? Resistor bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 4.7k?
14. R5 1k? Resistor bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 1k?
15. S1 Reed switch package THT
16. VALVE1 Plastic Solenoid Valve
17. Hackable Prototyping Board
18. IP67 case

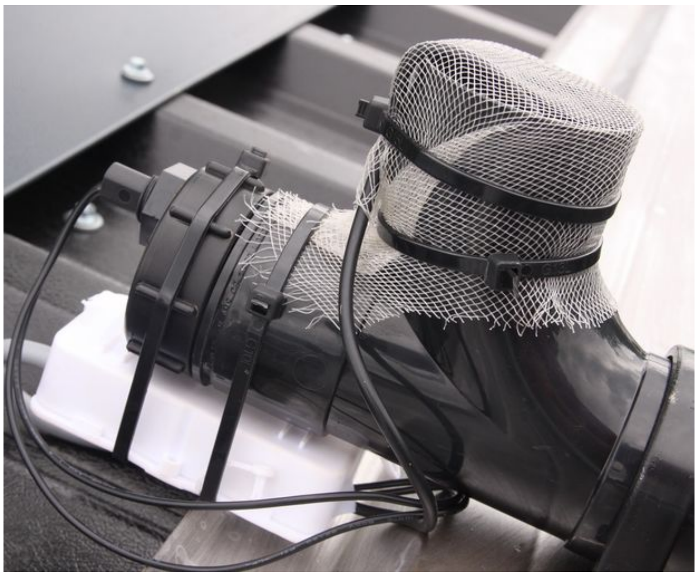
Please be aware that this particular system uses AC 240v power which can kill due to electric shock. All connections must be secured with strain relief grommets and the case must be labelled with a 'danger of shock' sticker and be secured against tampering with a suitable lock. All cables must be armoured or encased in appropriate sheaving. For legal compliance in many countries, this installation will need to be signed off by a qualified electrician. A safer way to do this is with DC pumps and solenoid valves.

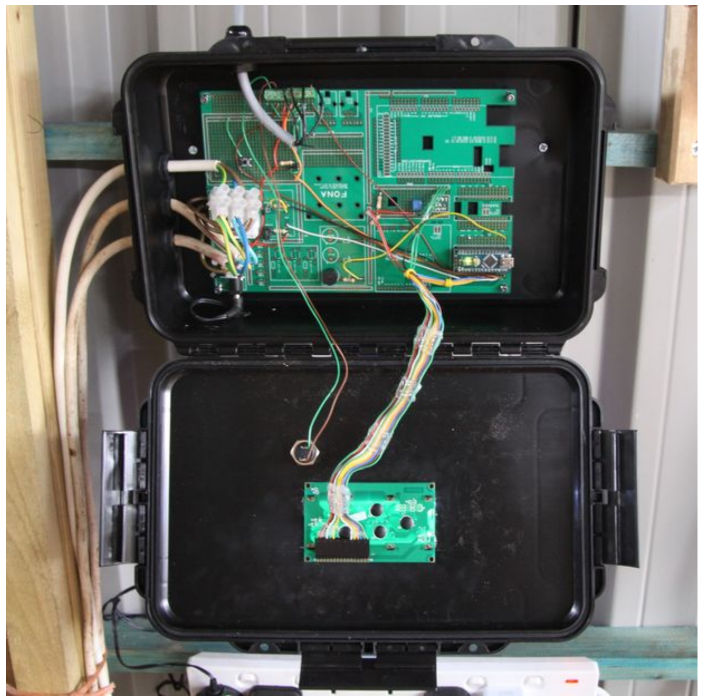
How it Works:

There are two sensors in the system, one telling us if the pipes are full and the other if the water in the pipes is hot enough. The temperature is not part of the automation, but it's very useful to know how warm the water is going to be. The level switch is screwed into the upper most inspection cap as shown in the photo and wired through a junction box back to the control panel, which activates a solenoid valve or a pump, depending on the status of the system as a whole. The mesh over the breather pipe is to stop leaves from getting into the water.

The nano is programmed with six main sets of logic which take care of all the possible combinations of level switch, panel switch, pump and solenoid valve. There are some instances where we need to prevent the solenoid valve coming on, even if the level switch indicates lack of water. Also, we need to be able to use the main panel switch to turn the pump off as well as turning it on.

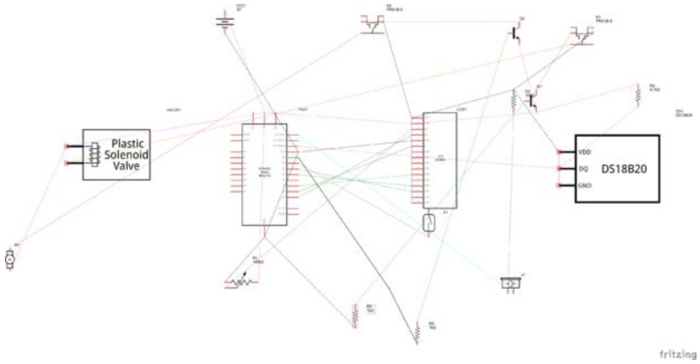






Assembly List

Label	Part Type	Properties
DS1	DS18B20 1-Wire Temperature Sensor	part # DS18B20
J1	Piezo Speaker	
K1	Relay	package THT; switching circuit SPDT; voltage 5V; contact rating 125VAC / 30VDC @ 1 AMP; part # FRS1B-S
K2	Relay	package THT; switching circuit SPDT; voltage 5V; contact rating 125VAC / 30VDC @ 1 AMP; part # FRS1B-S
LCD1	LCD screen	type Character; pins 16
M1	DC Motor	
Part1	Arduino Nano (Rev3.0)	type Arduino Nano (3.0)
Q1	NPN-Transistor	package TO92 [THT]; type NPN (EBC)
Q2	NPN-Transistor	package TO92 [THT]; type NPN (EBC)
R1	Rotary Potentiometer (Small)	size Rotary - 9mm; package THT; maximum resistance 100kΩ; track Linear; type Rotary Shaft Potentiometer
R2	100Ω Resistor	bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 100Ω
R3	1kΩ Resistor	bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 1kΩ
R4	4.7kΩ Resistor	bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 4.7kΩ
R5	1kΩ Resistor	bands 4; package THT; tolerance ±5%; pin spacing 400 mil; resistance 1kΩ
S1	Reed switch	package THT
VALVE1	Plastic Solenoid Valve	



Step 16: Nano code

The Nano uses quite a few digital pins to run the LCD:

- Pin (1) VSS Connect to - 0 volts (ground)
- Pin (2) VDD Connect to + 5 volts
- Pin (3) V0 Connect to center of 10K pot
- Pin (4) RS Connect to Pin D2
- Pin (5) RW Connect to - 0 volts (ground)
- Pin (6) E Connect to Pin D3
- Pin (7) D0 Not Used N/A
- Pin (8) D1 Not Used N/A
- Pin (9) D2 Not Used N/A
- Pin (10) D3 Not Used N/A
- Pin (11) D4 Connect to Pin D4
- Pin (12) D5 Connect to Pin D5
- Pin (13) D6 Connect to Pin D6
- Pin (14) D7 Connect to Pin D7
- Pin (15) A to +5 volts
- Pin (16) K Connect to - 0 volts (ground)

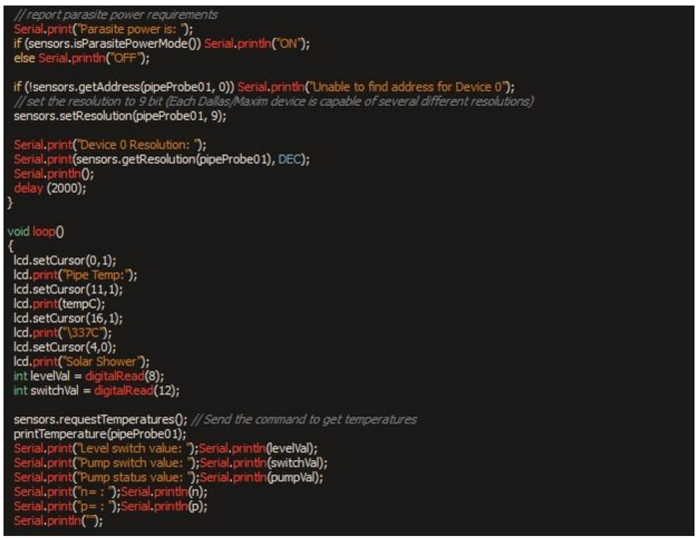
Other than this, the code is essentially divided up into six logic statements that take care of the following instances:

1. The pump is running and we want to stop the solenoid valve from introducing cold water into the system as we don't want cold water coming through into the pump.
2. The dirty person has previously turned on the pump but decides that they don't want a shower and press the button again to stop the pump.
3. The pump is off and the dirty person presses the button to turn it on.
4. When the pump is turned on a count down timer is initiated to automatically turn it off again.
5. If the pump is off and the level switch senses a lack of water then the solenoid is opened.
6. If the pump is on and the level switch senses that water is being lost then the dirty person must be actually having a shower and the count down timer is reset to allow them plenty of time to use all the water up.

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);

// Data wire is plugged into port 11 on the Arduino
#define ONE_WIRE_BUS 11
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
DeviceAddress pipeProbe01 = { 0x28, 0x1E, 0x28, 0x08, 0x00, 0x00, 0x80, 0x00  };
float tempC;
int pumpVal =0;
int levelVal =0;
int switchVal =0;
int n=0;
int p=0;
int z=0;

void setup()
{
 tone(13,500,500);
 Serial.begin(9600);
 lcd.begin(20, 4);
 lcd.setCursor(4,0);
 lcd.print("Solar Shower");
 pinMode(8, INPUT_PULLUP);
 pinMode(9, OUTPUT); // Relay
 pinMode(10, OUTPUT); // Relay
 pinMode(12, INPUT_PULLUP);
 pinMode(13, OUTPUT);

 // locate devices on the bus
 Serial.print("Locating devices...");
 sensors.begin();
 Serial.print("Found ");
 Serial.print(sensors.getDeviceCount(), DEC);
 Serial.println(" devices.");

 // report parasite power requirements
 Serial.print("Parasite power is: ");
```

```
 if (sensors.isParasitePowerMode()) Serial.println("ON");
 else Serial.println("OFF");

 if (!sensors.getAddress(pipeProbe01, 0)) Serial.println("Unable to find address for Device 0");
 // set the resolution to 9 bit (Each Dallas/Maxim device is capable of several different resolutions)
 sensors.setResolution(pipeProbe01, 9);

 Serial.print("Device 0 Resolution: ");
 Serial.print(sensors.getResolution(pipeProbe01), DEC);
 Serial.println();
 delay (2000);
}

void loop()
{
 lcd.setCursor(0,1);
 lcd.print("Pipe Temp:");
 lcd.setCursor(11,1);
 lcd.print(tempC);
 lcd.setCursor(16,1);
 lcd.print("\337C");
 lcd.setCursor(4,0);
 lcd.print("Solar Shower");
 int levelVal = digitalRead(8);
 int switchVal = digitalRead(12);

 sensors.requestTemperatures(); // Send the command to get temperatures
 printTemperature(pipeProbe01);
 Serial.print("Level switch value: ");Serial.println(levelVal);
 Serial.print("Pump switch value: ");Serial.println(switchVal);
 Serial.print("Pump status value: ");Serial.println(pumpVal);
 Serial.print("n= : ");Serial.println(n);
 Serial.print("p= : ");Serial.println(p);
 Serial.println("");

 z=0;
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
 if ((levelVal == LOW)||(pumpVal == 1))               // Level switch is on but valve won't open if pump is on.
 {
   digitalWrite(10, LOW);
   lcd.setCursor(0,2);
   lcd.print("Fill Valve: Closed");
 }
 else
 {
   digitalWrite(10, HIGH);
   lcd.setCursor(0,2);
   lcd.print("Fill Valve: Open");
 }
///////////////////////////////////////////////////////////////////////////////////////////////////////////////////
 if ((switchVal == LOW)&&(pumpVal == 1))             // Switch turned on whilst pump is running.
 {
   digitalWrite(9, LOW);
   tone(13,500,500);
   lcd.setCursor(0,4);
   lcd.print("Pump: Off ");
   n=0;
   p=0;
   z=1;                                              // Tell us that the pump needs to be off.
   pumpVal =0;                                       // Tells us that the pump is off.
   delay(10000);                                     // Pump won't come on again for 10 seconds.
 }
///////////////////////////////////////////////////////////////////////////////////////////////////////////////////
 if ((switchVal == LOW)&&(z==0))                      // Switch turned on, unless pump has been turned off.
 {
   digitalWrite(9, HIGH);
   tone(13,500,500);
   lcd.setCursor(0,4);
   lcd.print("Pump: On ");
   n=-600;
   pumpVal =1;                                       // Tells us that the pump is on.
 }
 else
 {
   digitalWrite(9, LOW);
 }
///////////////////////////////////////////////////////////////////////////////////////////////////////
 if (n<0)
 {
   digitalWrite(9, HIGH);
   tone(13,500,500);
   lcd.setCursor(0,4);
   lcd.print("Pump: On ");
   n++;
 }
 else
 {
   lcd.setCursor(0,4);
   lcd.print("Pump: Off");
   pumpVal =0;
 }
///////////////////////////////////////////////////////////////////////////////////////////////////////
 if ((levelVal == HIGH)&&(pumpVal==0))             // Valve is open and pump is off.
 {
   n=0;
 }
///////////////////////////////////////////////////////////////////////////////////////////////////////
 if ((levelVal == HIGH)&&(pumpVal==1)&&(n==0))     // Somebody is actually having a shower.
```

```
 {
   p++;                                             // Shower timer initiated.
 }
 if ((p<600)&(p>0))
 {
   digitalWrite(9, HIGH);
   tone(13,500,500);
   lcd.setCursor(0,4);
   lcd.print("Pump: On ");
   pumpVal =1;                                      // Tells us that the pump is on.
   p++;
 }
 else
 {
   p=0;                                             // Shower timer reset to zero.
 }

 delay (1000);
////////////////////////////////////////////////////////////////////////////////////////////////////
}
void printTemperature(DeviceAddress deviceAddress)
{
 tempC = sensors.getTempC(deviceAddress);
 Serial.print("Temp C: ");
 Serial.print(tempC);
 Serial.print(" Temp F: ");
 Serial.println(DallasTemperature::toFahrenheit(tempC)); // Converts tempC to Fahrenheit


}
```



```
 //report parasite power requirements
 Serial.print("Parasite power is: ");
 if (sensors.isParasitePowerMode()) Serial.println("ON");
 else Serial.println("OFF");

 if (!sensors.getAddress(pipeProbe01, 0)) Serial.println("Unable to find address for Device 0");
 //set the resolution to 9 bit (Each Dallas/Maxim device is capable of several different resolutions)
 sensors.setResolution(pipeProbe01, 9);

 Serial.print("Device 0 Resolution: ");
 Serial.print(sensors.getResolution(pipeProbe01), DEC);
 Serial.println();
 delay (2000);
}

void loop()
{
 lcd.setCursor(0,1);
 lcd.print("Pipe Temp:");
 lcd.setCursor(11,1);
 lcd.print(tempC);
 lcd.setCursor(16,1);
 lcd.print("\337C");
 lcd.setCursor(4,0);
 lcd.print("Solar Shower");
 int levelVal = digitalRead(8);
 int switchVal = digitalRead(12);

 sensors.requestTemperatures(); //Send the command to get temperatures
 printTemperature(pipeProbe01);
 Serial.print("Level switch value: ");Serial.println(levelVal);
 Serial.print("Pump switch value: ");Serial.println(switchVal);
 Serial.print("Pump status value: ");Serial.println(pumpVal);
 Serial.print("n= : ");Serial.println(n);
 Serial.print("p= : ");Serial.println(p);
 Serial.println("");
```

Step 17: Final Test

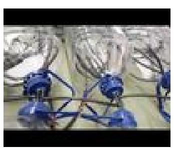
Rather than just have a shower outside in swimming shorts, we set up a 'pop up' shower tent. The system works really well on a nice sunny day and we get a nice warm shower for about 5 minutes until the water runs out. To operate the shower, the red button on the control panel is pressed and held in for one second. There's plenty of time to get into the shower tent and the system automatically detects when the dirty person has started to have a shower and gives them some extra minutes for completion. To stop the shower, the red button can be pressed in again or we can just wait for the count down timer to finish.

Please vote for this instructable in the contests! Thank you!

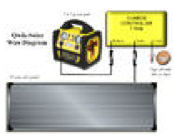
Related Instructables

Solar Tracking Water Heater by AWMT

Solar Hot Water DIY EASY Cheap by GREENPOWERS(



Solar Water Heater with Plastic Bottles by bubal



Build a Qwik-Solar Step by Step by Conserver

Homemade Solar Water Heater forced circulation with mini pump by apachebcn

Paddling pool solar water heater by manuka

Comments