```
// Solar Charging Monitor by Glen Popiel - KW5GP
/*

 This program is free software: you can redistribute it and/or modify
 it under the terms of the version 3 GNU General Public License as
 published by the Free Software Foundation.

 This program is distributed in the hope that it will be useful,
 but WITHOUT ANY WARRANTY; without even the implied warranty of
 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 GNU General Public License for more details.

 You should have received a copy of the GNU General Public License
 along with this program.  If not, see <http://www.gnu.org/licenses/>.

 */

#include <LCD5110_Basic.h>  // Use the Nokia 5110 LCD Library

#define Amps A0  // Define the Analog Input pin for the current sensor
#define Solar_In A1  // Define the Analog Input pin for the Solar Cell
Voltage
#define Battery_In A2  // Define the Analog Input pin for the Battery
Voltage
#define calibration_value 2410  // Define the calibration value used to
map the solar cell and Battery voltages
#define amp_calibration 5120  // Define the calibration value used to map
the charging current to milliamps

/*
 It is assumed that the LCD module is connected to
 the following pins.
      CLK - Pin 12
      DIN - Pin 11
      DC  - Pin 10
      CE  - Pin 8
      RST - Pin 9
*/

float solar_voltage, battery_voltage;  // Variables to contain the solar
cell and battery converted values
int charge, solar, battery, charge_current;  // Variables to contain the
sensor data

LCD5110 glcd(12,11,10,8,9); //  Assign the Nokia 5110 LCD Pins

extern uint8_t SmallFont[];  // define the Nokia Font

void setup()
{
  glcd.InitLCD(70);  // Initialize the Nokia 5110 Display, set the
Contrast to 65
  glcd.setFont(SmallFont);  // Set the Font to Small Font
  glcd.print("KW5GP", CENTER, 0);  // Display the Startup screen
```

```
  glcd.print("Solar Cell", CENTER, 8);
  glcd.print("Charging", CENTER,16);
  glcd.print("Monitor", CENTER,24);
  delay(3000);
  glcd.clrScr();  // Clear the LCD screen
  glcd.print("Status", CENTER, 0);  // Set up the LCD screen for the data
  glcd.print("Solar: ",0,16);
  glcd.print("Battery: ",0,24);
  glcd.print("Current: ",0,32);

}  // End Setup Loop

void loop()
{
  // Read the sensors and display the voltages and current

  charge = analogRead(Amps);  // Read the current sensor
  solar = analogRead(Solar_In);  // Read the Solar cell voltage divider
  battery = analogRead(Battery_In);  // Read the Battery voltage divider

  solar_voltage = map(solar,0,1023,0,calibration_value);  // Map the
Solar Cell A/D value to voltage
  battery_voltage = map(battery,0,1023,0,calibration_value);// Map the
Battery A/D value to voltage
  charge_current = map(charge,0,1023,0,amp_calibration);  // Map the
Current Sensor A/D value to milliamps

  glcd.printNumF(solar_voltage/100,2,48,16);  // Display the data
  glcd.printNumF(battery_voltage/100,2,48,24);
  glcd.printNumI(charge_current,48,32,4);

  glcd.print("V",72,16);
  glcd.print("V",72,24);
  glcd.print("ma",72,32);

  if (charge_current > 0)  // Charging Status Indicator
  {
    glcd.print("  Charging  ",CENTER, 40);
  } else {
    glcd.print("Not Charging", CENTER,40);
  }
  delay(5000);

}
```