Solar panel temperature control system using IoT

Min-soo Kim
Department of Electrical Engineering, Pusan University, Busan, Republic of Korea

Hee-je Kim
Department of Electrical Engineering, Pusan University, Busan, Republic of Korea
E-mail: rlaalstn5122@naver.com, heeje@pusan.ac.kr
www.pusan.ac.Kr

Abstract

Solar photovoltaic systems are renewable energy sources that are used widely around the world. On the other hand, the efficiency decreases as the temperature of the solar panels increases. To prevent this phenomenon, a cooling fan can be installed on the back side of the solar panel to increase the efficiency. The solar system efficiency also decreases due to weather conditions and unexpected situations. To overcome this problem, an IoT (Internet of Things) system was used to monitor the state of the solar system and control the cooling fan. The core microprocessor used in IoT systems was Arduino. Using Arduino, an IoT system can be implemented simply and inexpensively. The entire system was designed and tested and the efficiency increased by approximately 4.7%. Although it is a small 30W capacity photovoltaic system, its efficiency is expected to be increased by applying it to a photovoltaic system of more than 1kW in the near future.

Keywords: Arduino, Cooling fan, IoT, Smart phone, Solar panel, Wi-Fi

1 Introduction

Nowadays, IoT (Internet of Things) is getting popular in home and industry [1]. IoT is a future network technology that can share information and control things between people, things, things and things [2]. This allows remote control and monitoring from a distance. Based on this IoT, it can be applied to solar power systems currently used worldwide.

Solar power generation system uses solar energy to produce electricity, so if the temperature of the solar panel rises, the power generation efficiency becomes low [3]. In particular, at noon time, the surface temperature of the solar panel reaches from 60 to 70 degrees. There are several ways to prevent such degradation of efficiency. First, cooling water is sprayed on the solar panel surface to lower the temperature. Second, lower the temperature by attaching a heat sink. Third, use a cooling fan to lower the temperature. Experimental results show that the last of these three methods is optimal for lowering the temperature of the solar panel.

However, this cooling fan must be activated when the temperature of the solar panel rises, and turned off when the temperature is low. Although the temperature can be controlled automatically, it is impossible to control the entire system when the temperature sensor fails or is not ideal. For this reason, the system is manually controlled through monitoring.

The microcontroller used in this paper is Arduino. Arduino is popular among engineers because it is easy to use and inexpensive [4]. However, Arduino alone can't communicate over distance. So it controls the cooling fan through Wi-Fi communication between Smartphone and Arduino. Arduino is combined with a Wi-Fi shield, and there is a server in the Wi-Fi shield. Through this server, various sensor signals and switch signals can be sent to smart phone and Arduino. This makes it possible to turn on and off the switch directly after checking the solar panel temperature.

After making the actual model, we confirmed the normal operation between Arduino and smart phone using Wi-Fi communication and also confirmed that the temperature of solar panel is lowered. Through the monitoring and control as above, the solar power generation system can make more efficient electricity generation.

2 Major System Components

The system consists of several important parts. Arduino which acts as a brain, Wi-Fi shield for Wi-Fi communication, infrared temperature sensor for receiving infrared signals, and relays that activate the cooling fan are important components. Sub-components include voltage, current sensor, ambient temperature sensor, and LCD. The temperature of the solar panel can be appropriately maintained by the organic operation of these parts.

2.1 Arduino

Fig. 1 shows the Arduino Mega used in this system. The main chip is atmega2560, which has 54 digital input / output pins, 16 analog input pins, 256kb flash memory and 8KB SRAM.

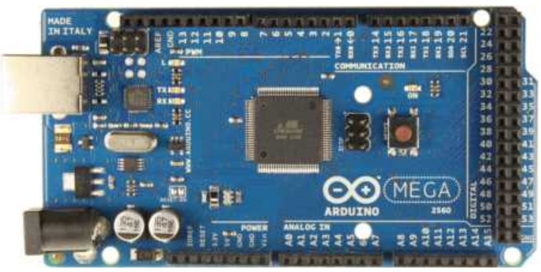

Fig. 1. Arduino MEGA 2560

And Arduino is programmed in assembly language like c language and Java.

2.2 Wi-Fi shield

The Wi-Fi shield is used in conjunction with Arduino to communicate with Arduino and the smart phone. The jsn270 model was used and the specifications are shown in Table 1 below.

Table 1. JSN270 Specification

Wireless Standard	IEEE 802.11 b/g/n
Frequency Range	2.412~2.477 GHz
Data Rates	Max 72Mbps
Modulation Type	OFDM, DSSS
Wi-Fi Security	WEP 64/128bit, WPA/WPA2 personal

2.3 Infrared sensor

The infrared sensor is a device that measures the temperature by detecting the infrared rays coming from the object. Remote measurement is possible and failure does not occur. In this system, non-contact infrared sensor (MLX90614) is used and can be measured from -70.01℃ to +382.19℃. It has resolution of 0.01℃ and communication method is I2C. It is shown in Fig. 2.

Fig. 2. MLX90614

2.4 Relay

Relay receives signal from Arduino and turns cooling fan on and off. The maximum input voltage is 250V for AC and 30V for DC.

2.5 Other sensors and devices

There are other sensors except for the main components. For example, voltage and current sensor, ambient temperature sensor, cooling fan, LCD. Other devices include AC-DC converter, 30W solar panel, cradle, and electric box.

3 Fabrication

3.1. Design

The total system was designed using the parts described above. Various sensors and relays are attached to the center of Arduino, and finally one system is completed. In addition, a smart phone application for operating the Arduino is also designed. The app was designed with easy-to-use App Inventor 2 interface. Fig. 3 and Fig. 4 show the system schematic and App Inventor 2 interface.

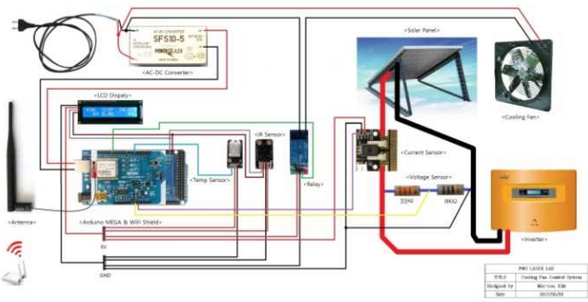

Fig. 3. The Schematic of the System

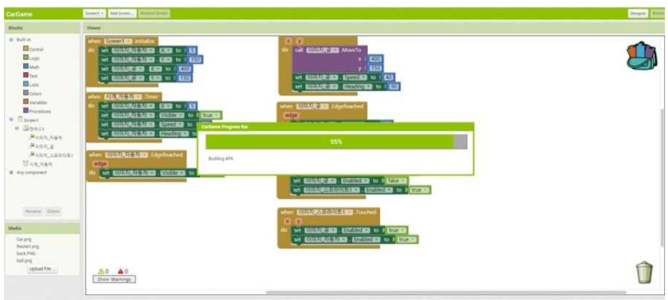

Fig. 4. App Inventor 2 interface

3.2 Program Code

Finally, Arduino requires source code for device operation. The source code is divided into four parts: Wi-Fi connection, LCD display, temperature and voltage and current sensors, and switch operation. The code below shows only some important parts. The first is a header part where you set the header file to use in your code. The second part is for Wi-Fi connection, where you can set the SSID, Password and encryption method.

```
/* Wi-Fi connect part */
  #define SSID     "air"
  // your wifi network SSID
  #define KEY       ""
  // your wifi network password
  #define AUTH       "NONE"
  // your wifi network security (NONE,
  WEP, WPA, WPA2)

/*Server part*/
  // HTTP headers always start with a
  response code (e.g. HTTP/1.1 200 OK)
  // and a content-type so the client
  emknows what's coming, then a blank
  line:
  JSN270.println("HTTP/1.1 200 OK");
  JSN270.println("Content-
  type :text/html");
  JSN270.println("Refresh: 30");
  JSN270.println();
```

3.3 Final model

A final model was created as shown in Fig. 5. Six cooling fans were attached to the solar panel and a converter was added to change the input voltage to 12V. An LCD was attached to the main box to verify the data directly. Finally, the completed application is shown in Fig. 6. There is a cooling fan switch button and added function to check various data in real time.

Fig. 5. Final model

Fig. 6. Smart phone application

4 System Test

We confirmed that the manufactured system works normally. The On / Off of the cooling fan was controlled by a smart phone and Heater was used to simulate sunlight. And heat the solar panel using a heater as shown in Fig.7.

Fig. 7. Testing set-up

The temperature of the solar panel was maintained at over 57°C and the ambient temperature was maintained at 28°C.

After running the cooling fan through the smart phone, the temperature dropped by 40°C. At this time, it takes about 20 minutes. The temperature change of the solar panel is shown in Fig. 8. The temperature of the solar panel is generally lowered.

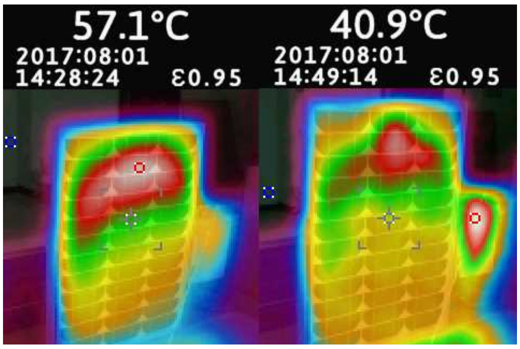

Fig. 8. Temperature change after cooling fan operation

5 Conlusion

In this paper, we designed and manufactured a solar panel temperature control system. With Arduino and Wi-Fi shield, it is now possible to control the temperature of the solar panel at anywhere and anytime. By lowering the temperature of the solar panel, the power generation efficiency can be increased. In addition, this system control via smart phone is very simple. In the near future, we need to research more about technologies that can significantly lower the temperature of solar panels while reducing costs. It's possible to maximize power generation efficiency by applying this system to general household and large-scale photovoltaic system.

References

[1] K. J. Park (2017), *"Arduino makes your imagination a reality,"* Edu-I, pp.198

[2] O. Vermesan et al.,"Internet of things strategic research roadmap," *Internet of Things-Global Technological and Societal Trends,* vol. 1, pp.9-52, 2011.

[3] S. Dubey, J. N. Sarvaiya, B. Seshadri, "Temperature Dependent Photovoltaic(PV) Efficiency and Its Effect on PV Production in the World – A Review," *Energy Procedia*, vol. 33, 2013, pp.311- 321.

[4] Adriansyah A, Dani, A.w., "Design of Small Smart Home system based on Arduino," in *Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS),* 2014, pp.121-125,27-28 Aug. 2014