Design of Non-contact Thermal Inspection System Based on Embedded Linux and Cloud Service

Yousong Guo[a], Jun Xu

Academy of Army Armored Forces, Beijing 100072, China
[a]gyousong@163.com

Keywords: Infrared Thermometer, Embedded Linux, Cloud, Raspberry Pi.

Abstract: In many industrial fields, it is necessary to measure environmental parameters, including temperature, pressure, humidity, etc. These measurements are inseparable from the selection of suitable sensors. This paper proposes and designs a non-contact infrared thermal inspection system. The system collects and processes data from the thermal sensor through a single-board computer running the Linux operating system, and then uploads the measurement data to the cloud server using a Wi-Fi wireless network or a cellular network. The data can be viewed by the user through a web browser.

1. Introduction

A thermometer is the main instrument used to measure temperature. However, in the case of forest fire prevention, it is impossible to detect the ignition point in advance through a thermometer placed on site. In this case, remote monitoring is usually carried out using satellite or aerial remote sensing equipment. Aden, S.T. et el analyzed the advantages of various monitoring methods in detail, and designed a low-cost solution for using UAV for hotspot monitoring [1]. In another case, to ensure the normal and safe operation of the substation, the condition monitoring is often performed according to the temperature change of the equipment cabinet. This work is usually done by people, not only the task is very heavy, but also inefficient. L. Chapman et el designed a low-cost, non-contact road surface temperature sensing system [2] that can be easily installed on both sides of the road and can send temperature data to the server over the wireless network. Guo L. et el designed the temperature and humidity remote measurement system, where the temperature values are collected first, and then displayed through the client program interface [3]. In this way, remote automatic monitoring can be realized, which brings great convenience to the actual work.

The main purpose of this paper is to design a non-contact infrared thermal inspection system which is capable of sending data to the cloud server and presenting it to the user in a more convenient and flexible way. The measurement data is usually provided to the user in three ways. First, the data is directly stored on the memory card, and after the measurement is completed, the memory card is transferred to a computer, and displayed through a specially designed client program. Obviously, users have to wait a long time before they can see the data. Even if a small display is installed on the

Published by CSP © 2018 the Authors
DOI: 10.23977/icmit.2018.008

measuring device, it is not possible for the user to see the data directly when the device is on a balloon, UAV or mobile robot. Needless to say, the payload of an aircraft such as a UAV or a balloon is usually small, and an overly heavy measuring device is likely to result in a reduction in endurance time and a reduction in cruising range. The second way is to run a web server on the measurement device, so the user can view the data through a web browser in a wireless LAN. But this method is mainly affected by the coverage of the wireless LAN, and is only suitable for indoor or short-distance applications. In the third way, the measurement device first sends the data to the Internet server, and the user views the measurement data through a web browser in any environment with an Internet connection. Although this method has a longer path of information transmission, with the popularity of the Internet and the expansion of the coverage of mobile networks, efficiency can be improved, which is also one of the mainstream directions of technology development. The design of this paper is based on the third approach described above.

2. System Design

The system mainly includes three parts: measurement controller, cloud server and client. The measurement controller consists of peripheral devices such as a single board computer (SBC), sensors, and wireless communication modules. Cloud servers leverage popular commercial cloud computing platforms such as Microsoft's Azure and Amazon's AWS. The client is usually a desktop, a laptop, a mobile phone, a tablet, and the like. Considering compatibility and convenience, the client directly connects to and views sensor data using the device's own web browser, eliminating the need to develop a separate client application.

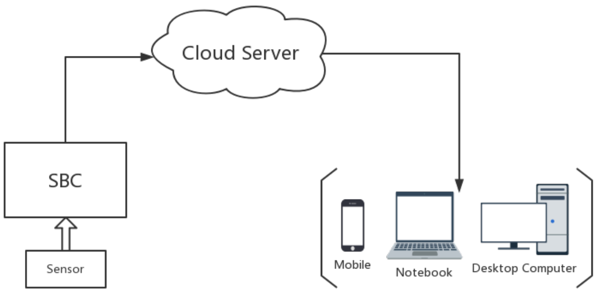

Fig. 1 Block Diagram of Complete System

3. Hardware Components

3.1 Measurement Controller

The measurement controller is used to connect the sensors and send the measurement data to the cloud server. There are many control boards capable of doing this, such as the current popular open source hardware, including Arduino series, Raspberry Pi series, BeagleBone Black and so on. Based on an 8-bit AVR microcontroller, Arduino has an easy-to-learn development environment. However, Arduino does not have an operating system and is not suitable for complex applications such as network communication. Both the Raspberry Pi and the BeagleBone Black are based on a 32-bit ARM processor with integrated high-capacity RAM and powerful audio and video processing units, as well as a rich expansion interface. These boards usually run the Linux operating system and are capable of performing network communications, graphics and image processing, and other tasks that require high computational performance. They are smaller, more energy efficient, and less expensive than traditional computer and professional industrial motherboards. At the same time, rich community resources make learning and problem solving very easy. Of course, enough GPIO port resources make it possible to connect directly to peripheral devices.

The Raspberry Pi was developed by the Raspberry Pi Foundation of the United Kingdom. It is essentially a single-board computer based on an ARM processor. It has the CPU, RAM, and peripheral interfaces that a computer system should have. The SD/MicroSD card slot allows it to be connected to a large-capacity memory card. The HDMI interface allows it to connect to a TV and display, and the USB interface gives it the same scalability as a desktop computer. Running the Linux operating system means having a wealth of software resources for it to use. The Foundation initially hoped to facilitate computer science education in schools through this product, which is priced at only a few tens of dollars. Today, however, it has become one of the most popular IoT product development boards in the world.

This article is based on the latest version of the Raspberry Pi 3 generation B plus. Compared to the BeagleBone Black, which has not been upgraded for many years, it is superior in CPU performance and RAM. The integrated Wi-Fi module also makes wireless networking more convenient.

3.2 Infrared Thermometer

Considering that in some cases, it is necessary to carry out tasks from the air by using an aircraft such as a UAV to carry the measuring device, the weight of the measuring controller needs to be small enough and the power consumption is low enough. Therefore, the design uses a lightweight infrared thermometer and discards the larger and heavier thermal imager.

The MLX90614 is a high-precision, high-resolution, non-contact infrared thermometer from Melexis with a 17-bit ADC and a powerful DSP unit. It has been calibrated at the factory and does not require additional calibration by the user. The measured temperature value will be output via a PWM signal or SMBus bus. With SMBus, the resolution of the measurement may reach to 0.02 degrees Celsius. According to the experiment [4], the measurement error does not exceed plus or minus 0.5 degrees Celsius, which is almost equivalent to the high precision PT100 temperature sensor. Fig.2 shows the typical application schematic of MLX90614 [5].

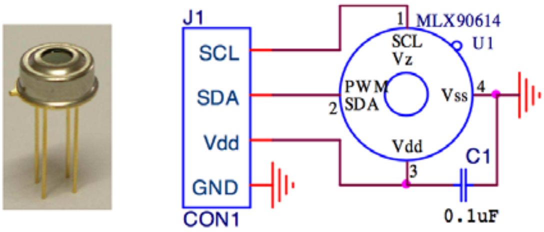

Fig. 2 MLX90614 and the typical application schematic

3.3 Communication Module

The system relies on the Internet to upload sensor data to the cloud server. Outdoors, the server can only be connected via Wi-Fi or cellular networks. The Raspberry Pi 3B integrates a Wi-Fi module but does not have a cellular communication module. This design uses SIM800C communication module produced by SIMCom. It is a quad-band GSM/GPRS module with a working frequency of 850/900/1800/1900MHz. The external size of the module is only 17.6*15.7*2.3mm, and the price is low, the performance is stable, the power consumption is low. It is useful in the voice, SMS and Data information transmission and other needs, especially suitable for the design needs of various compact products.

SIM800C supports a variety of TCP/IP connection protocols. The function setting and data transmission requires AT commands sent by the computer through a serial port. Since the serial port of the Raspberry Pi 3 is occupied by the Bluetooth module, an “USB to TTL” conversion module may be needed during actual application.

4. Software Design

4.1 Sensor Data Acquiring

A dedicated program is required on the measurement controller to read the sensor values and send them to the cloud server. There are a variety of programming languages to choose from, whether C, C++, Python, or Java. The measurement controller can directly detect the PWM signal sent by the MLX90614, and can also read more accurate measurement results through the SMBus bus. This article chooses the second way, and writes a communication program based on python-smbus library.

4.2 Send Data to Cloud

Measurement data needs to be uploaded to the server before it can be viewed by the user. With the development of technology, cloud computing has become popular. Compared with traditional servers, cloud servers not only improve service reliability and data security, but also allow users to pay on demand, which is cheaper for small websites.

When the network terminal connects to the server, the most commonly used protocol is HTTP. While with the rapid development of the Internet of Things(IoT) technology, cloud servers have also begun to support the MQTT protocol. MQTT is a machine-to-machine (M2M)/"Internet of Things”(IoT) connectivity protocol[6]. It is essential when response time, throughput, lower battery and bandwidth usage are on the first place for future solutions. It’s also perfect in case of intermittent connectivity[7].

Within the cloud computing market, there are several companies leading in the space, including Amazon Web Services (AWS) , Microsoft Azure, IBM, Google Cloud, Alibaba, etc[8]. In this article, the Raspberry Pi will connect to Azure cloud via the MQTT protocol and upload the collected sensor data. "Azure IoT Hub" is a feature of Azure that makes it easier for IoT developers to use Azure cloud services. "IoT Hub" supports communications both from the device to the cloud and from the cloud to the device [9].

4.3 User Interface

To visualize the real-time sensor data that the IoT hub receives, a web application that is hosted on the web app should be run. The Azure provided an interface to make the progress easier.

Follow these steps to get the sensor data visualized [10]:

(1) Create a web app.

(2) Add a consumer group to this IoT hub.

(3) Configure the web app to read data from the IoT hub.

Now a web application can be uploaded. A beginner can start with a sample web application from GitHub (https://github.com/Azure-Samples).

5. Conclusion

This article describes an infrared thermal measurement system based on embedded linux and cloud services. A complete solution for temperature value measurement, data reading, transmission and storage is given. In order to overcome the shortcomings of the contact temperature sensor, the non-contact sensor MLX90614 is used, the sensor data is read by the Raspberry Pi 3, and the data is stored and retrieved through the cloud server. The measuring system can be installed on flight platforms such as balloons and UAVs, and has potential application value in various field operating systems. Measurement data can be presented to users through a common web browser and can be compatible with various computer terminal products.

References

[1] Aden,S.T., Bialas,J.P., Champion,Z., et al. Low cost infrared and near infrared sensors for UAVs. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (2014), XL-1,1-7

[2] L. Chapman, S.J. Bell. Low cost Road Surface Temperature sensing enabled by the Internet of Things. Proceedings of the 18th SIRWEC Conference, Ft. Collins, Colorado, USA, (2016) pp.1-5

[3] Guo L., Ye A., Lin T.. Fan R., Design of Remote Real-time Measuring System of Temperature and Humidity Based on Raspberry Pi and Java Language. Computer Measurement & Control, 2017, 25(6):4-8

[4] Jovanović U, Mančić D, Jovanović I, et al. Temperature Measurement of Photovoltaic Modules Using Non-Contact Infrared System. Journal of Electrical Engineering & Technology, 2017, 12(2):904-910

[5] Information on http://www.melexis.com

[6] Information on http://mqtt.org

[7] Marina Serozhenko. MQTT vs. HTTP: which one is the best for IoT. https://medium.com /mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105

[8] Cynthia Harvey. Top 7 Cloud Computing Companies in 2018. https://www.datamation.com /cloud-computing/slideshows/top-7-cloud-computing-companies-in-2018.html

[9] Nicole Berdy, Robin Shahan, Dominic Betts. What is Azure IoT Hub. https://docs.microsoft. com/en-us/azure/iot-hub/about-iot-hub

[10] Ranga Vadlamudi, Robin Shahan , Xin Shi, et al. Visualize real-time sensor data from your Azure IoT hub by using the Web Apps feature of Azure App Service. https://docs.microsoft.com/en-gb /azure/iot-hub/iot-hub-live-data-visualization-in-web-app