ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-PCaPAC2018-WEP05

WIRELESS INTERNET OF THINGS (IoT) APPLICATION IN THE TLS

C. H. Kuo, C. J. Wang, T. Y. Lee, W. U. Lin, Y. K. Lin, S. Fann, H. H. Chen, B. Y. Chen, H. C. Chen. S. J. Huang, C. S. Huang, J. A. Li
NSRRC, Hsinchu 30076, Taiwan

Abstract

The internet of thing is applied in the accelerator more frequently than before. There are many advantages in data acquisition and control-oriented applications, for example, easy to distribute remotely and less cables needed, low noise generated, many commercial interfaces for choosing. The stable wireless communication is also applied in the measurement system. The high reliability and security of wireless communication with server and client structure is introduced. The structure design and implementation of IoT are summarized in this report.

INTRODUCTION

The applications of IOT are based on Raspberry Pi 3 and Intel Celeron N3050 system in the operation group.

The Raspberry Pi 3 is applied in the temperature measurement with angalog/digital interface. The Raspberry Pi3 system is show in Figure 1.
Hardware specifications are listed:

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- Full-size HDMI
- 4 USB 2.0 ports
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display

Figure 1: Raspberry Pi 3 system.

The Intel N3050 system is apllied in the various displays these include of alarm, security monitor system. The systme is shown in Figure 2. Intel Celeron N3050 Kit is an entry-level, small-footprint. Equipped with a dual-core Celeron® processor, this Intel NUC includes new features. These features include items such as a VGA port, SDXC card slot and TOSLINK audio output. It also continues to offer 4x USB 3.0 ports, 4K video support, and 802.11ac Wi-Fi. The hardware specifiaction are lists:

(a) Model A (b) Model B

Figure 2: Intel Celeron N3050 system. There are two difference form factors are applied. Model A is suppoted in 4 K display and IR remote control, model B is applied in the weak wireless signals area with high gain antenna.

- Soldered-down dual-core Intel® Celeron® processor N3050 with up to 6 W TDP
- 4K HDMI display
- USB 3.0/SATA6.0
- IR Receiver Sensor
- Intel® Wireless-AC 3165 + Bluetooth 4.2
- 10/100/1000 ethernet

This system are applied in the 4K display. Alarm and security monitor are adapted in this application.

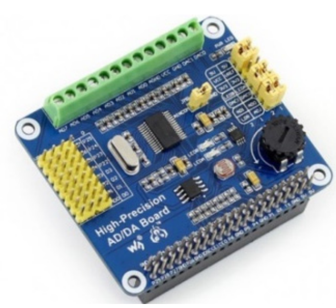


Figure 3: High-precision AD/DA convert.

Wireless lan and bluethooth interface are built in this low cost production that can provide amout of arrangement with simple DC power and manpower. In most time, the doors of storage ring tunnel are close. This is useful in the temporary measurement and system diagnostic for this short period time.

INTERFACE OF IoT

Analog Interface

Since there are no AD/DA functions on Raspberry Pi GPIO interface, this may restrict the application development on Raspberry Pi sometimes. Hence, we have chosen high-precision AD/DA board from Waveshare https [1] to compensate for this drawback. The features of this AD/DA board includes on-board ADS1256 (from Texas Instrument) of 8 channel 24 bit high-precision ADC (4 channel for differential inputs) with up to 30K samples per second data rate, on-board DAC8532(also from Texas Instrument) of 2 channel 16bit high-precision DAC, and 4 pins dedicated for digital I/O which are individually configurable as either inputs or outputs by the GPIO control register. The usage of this expansion board enables one to implement those data acquisition and external device control tasks on Raspberry Pi for applications of Internet of Things (IoT) field. This hardware board is shown in Figure 3.

Figure 4: Graphic user interface of AD/DA board in the control console.

The communication between the expansion board and Raspberry Pi is handled over an SPI-compatible serial interface that can be operated with a 2-wire connection. For the Raspberry Pi (SPI master), we have installed the Raspbian Jessie 4.9.24-v7+ operating system and all applications of software are built on it. EPICS (Experimental Physics and Industrial Control System) 3.15.5 release environment has been downloaded and established on the Raspberry Pi [2-3]. Consequently, the channel access technique is implemented on the expansion board and Raspberry Pi system. The C++ based asynPortDriver [4-5] device support is utilized to obtain duplex data interchanged between SPI master and slave. In EPICS database, waveform records are used for octet commands and responding data between both sides. For the purpose of initializing required commands sent to ADS1256, array subroutine records of EPICS are utilized. The sequence records of EPICS have been constructed for ADC (single ended and differential inputs), DAC and GPIO control logic sequences. The graphic user control interface of AD/DA board is shown in the Figure 4.

Digital Interface

Another application focuses on one-wire temperature and humidity sensors which communicate with the raspberry Pi using a single GPIO pin. The AM2302 and DHT22 sensors from Aosong [6] is utilized to implement the application. The DHT22 is shown in the Figure 5. It is digital connection with SPI interface. Because of better control over size and precise timing of sensors, C/C++ programs are compiled into *.so dynamic library. A software input output controller of EPICS is used for storing sensor readings whose values are updated by pyEpics (channel access for Python) [7] in the database. The method to link Python with C/C++ is to use a wrapper called Boost.Python [8] that binds C/C++ and Python. It simplifies the coding and provides methodology for calling to C++ objects in Python. The C++ namespace and Boost.Python specially defined macro are used for calling C/C++ functions within the previously mentioned *.so dynamic library [9].

Since EPICS channel access mechanism is implemented on the Raspberry Pi system, we have selected and utilized the CSS (control system studio) [10] channel access client software to build the graphic user interface for our applications. Please refer to Figures 6, which are obtained from acquired screenshots.

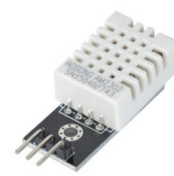


Figure 5: Temperature sensor DHT22 with digital interface.

In this context, we would like to investigate the possibility of constructing a corporate WiFi interconnected IoT network by CSS graphic user interface, database archiving tool, EPICS system, and Java EE server. This is only the initial stage, and a lot of efforts must be dedicated to in the future. We have expected more obstacles lying ahead of us, and we need to overcome those persistently.

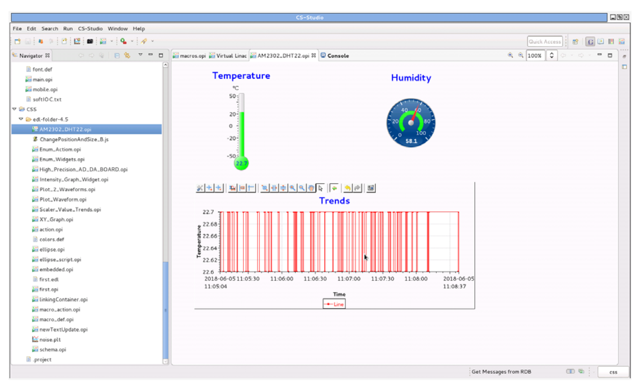
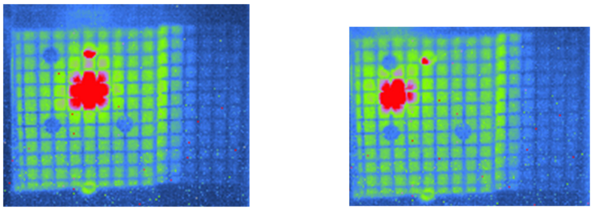


Figure 6: Graphic user interface of DHT22 temperature sensor.

The temperature isn't sense in the tunnel of storage ring. In the most case, the temperature sensor is far away from instrument. In Figure 7, one of the transport line quadrupole magnet cooling pipes is fail, the beam size is divergence. It is hard to injection from transport line to storage ring, but air temperature of tunnel is just 0.1°C difference. DHT22 supports high resolution and precision measurement. Low cost and less calibration with digital interface keep from cabling problem.

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-PCaPAC2018-WEP05



(a) Cooling water is fail (b) Cooling is recovery

Figure 7: Beam profile before injection in end of the transport line. (a) Cooling effect is gradually failed from water pipe congestion. (b) External fan to replace cooling water.

OTHER APPLICATIONS

The following applications are used in the operation and system diagnostics:

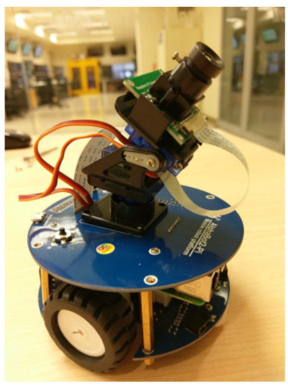


Figure 8: AlphaBot2 is tested in the TPS control room.

AlphaBot2

AlphaBot2 includes of multi interface in Raspberry PI3. It features rich common robot functions including line tracking, Bluetooth/infrared/WiFi remote control.

- 5-ch infrared sensor, analog output, combined with PID algorithm, stable line tracking
- Onboard modules like line tracking, obstacle avoiding, needs no messy wiring
- TB6612FNG dual H-bridge motor driver, compared with L298P, it's more efficient, more compact, and less heating
- N20 micro gear motor, with metal gears, low noise, high accuracy
- LM2596 voltage regulator, provides the Pi with stable 5V power
- TLC1543 AD acquisition chip, allows the Pi to use analog sensors
- PCA9685 servo controller, make it more smoothly to rotate the pan head
- CP2102 UART converter, easy for controlling the Pi via UART

The software programming is still going for auto pilot application. This system is shown in Figure 8.

Alarm and Security System Display

Alarm system of operation is based on the BEAST [11]. The css software with Intel N3050 platform is applied in the TV display. The alarm GUI is shown in Figure 9.

CONCLUSION

Wireless IoT is tested and applied in the various applications with low cost platform and easy distribution. The system reliability is still struggling with wireless interface. In the control room, the application is very stable. In the tunnel of storage ring, this wireless system isn't reliable. It still takes some efforts to improve in the future, such as radiation shielding, schedule system reboot, and so on.

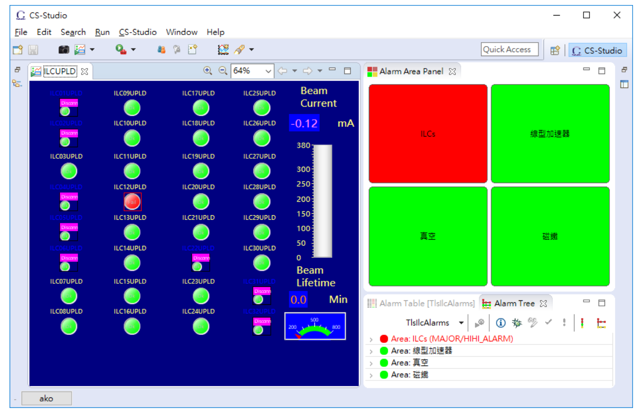


Figure 9: CSS alarm display with IoT.

ACKNOWLEDGEMENTS

Thank for Y. S. Cheng and Y. T. Chang of instrument and control group. They provide for network and database supporting. From these the resources, we can establish wireless control interface and data achieving for system analysis.

REFERENCES

[1] https://www.waveshare.com/wiki/High-Precision_AD/DA_Board

[2] http://prjemian.github.io/epicspi/

[3] https://epics.anl.gov/

[4] https://epics.anl.gov/modules/soft/asyn/R4-12/asynPortDriver.html

[5] https://github.com/ffeldbauer

[6] http://www.aosong.com/

[7] http://cars9.uchicago.edu/software/python/verview.html

[8] https://www.boost.org/doc/libs/1_66_0/libs/python/doc/html/index.html

[9] Molloy Derek, *Exploring Raspberry Pi Interfacing to the Real World with Embeded Linux*, John Wiley & Sons Inc., pp. 214-215.

[10] https://github.com/ControlSystemStudio

[11] https://github.com/ControlSystemStudio/cs-studio/wiki/BEAST