DEVICE FOR AUTOMATIC TESTING OF ULTRASONIC GAS LEAK DETECTOR

Petr Skryja

Doctoral Degree Programme (1), FEEC BUT

E-mail: xskryj01@stud.feec.vutbr.cz

Supervised by: Otakar Wilfert

E-mail: wilfert@feec.vutbr.cz

Abstract: In this paper a device for automatic testing of the gas leak detector is presented. The device allows testing product which is an embedded device that is improved during a development. This requires testing the product after each implementation of a new feature, as there may be situations where this implementation will stop some part of the product from being functional. For realization mini PC Beaglebone Black was used and software architecture of this device is introduced. The presented device is able to do automatic testing that is more accurate and faster than human testing.

Keywords: testing device, automatic testing, Beaglebone Black, software architecture

1 INTRODUCTION

When developing a new product, it requires more people to cooperate on its firmware. Each team member is responsible for his / her part of a code. The working procedure is such that the product is programmed with the basic functionality, which then develops to the final state. This approach to effective team collaboration on complex products is called Scrum. In order for developers to work on such a project, they use version control systems such as GIT or SVN. When creating a new firmware version, the product should be subjected to a series of tests to prove that the product's existing features is not stopped being functional by the newly created part of the code. A series of tests can be performed by a person, which can be a time-consuming task where human can introduces error. For this purpose a device that can provide series of tests automatically and clearly display occurred errors is good to use. In this paper the device for automatic testing of ultrasound gas leak detector will be introduced. The first part of this paper introduces the detector and hardware solution of the device for automatic testing. Then software architecture and graphic user interface (GUI) application for test controlling are presented. Finally, the principle of writing tests is described.

2 TESTING OF ULTRASOUND GAS LEAK DETECTOR (UGLD)

There are several parts that need to be tested on the UGLD. These parts are shown in Figure 1. The UGLD uses an acoustic sensor (piezoelectric microphone) to identify noise fluctuations that are imperceptible to human ear [1]. If the gas moves from a high pressure zone through a hole to a low pressure region, ”hissing” is generated [2]. This ” hissing ” is a broad band noise, ranging in frequency from Hz to tens of MHz [3]. To indicate an alarm relays, a mA loop and an LED are used. The detector is able to communicate by using RS-485 interface or Bluetooth. Setting of the device is located in system data store (SDS) block. The functions of the detector must be met within the temperature range from -55°C to +75°C [4]. For testing it, temperature chamber, generator of gas leak, temperature sensors, RS-485 communicator, mA loop sensor, bluetooth, relays (for power on/off the detector and simulation of disconnected mA loop) and circuit for checking relays are required.

For realization of the testing device development board Beaglebone Black was selected, because of GPIO pins number, USB port, UARTs peripherals and ADC that are on board. This development board is extended by board with DAC which is implemented by microcontroller STM32F051K6T6 and serves to generate/simulate signal of the gas leak.

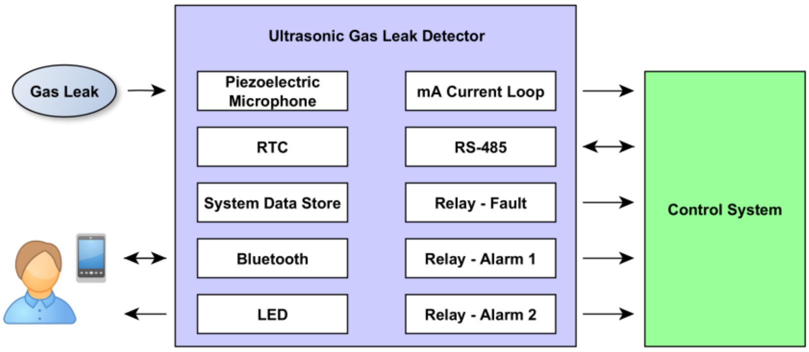


Figure 1: Block scheme of ultrasonic gas leak detector

In Figure 2 block scheme of workplace for testing is shown. Testing is controlled from PC which is connected to Beaglebone black via USB/UART converter and provides energy for this device. For powering the detector a power supply of 24V is used. For communication with detector RS-485 and Bluetooth is presented. Tests require two sensors to measure temperature of the chamber and temperature of the detector. A physic realization is shown in Figure 3.

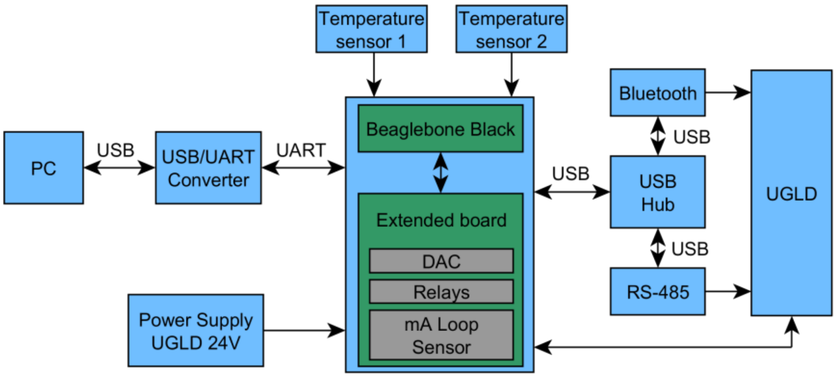


Figure 2: Block scheme of device for testing

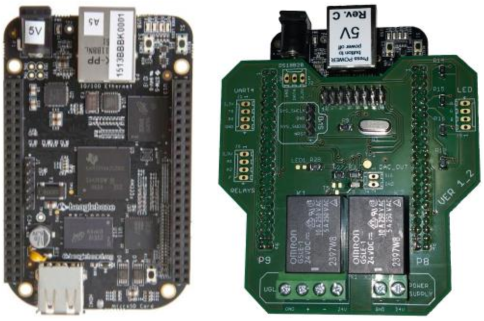


Figure 3: Development board Beaglebone Black and extension board

2.1 SOFTWARE ARCHITECTURE

The software architecture shown in Figure 4 was designed for the testing device. Each segment uses FP2 protocol for communication. The tests are controlled from PC by using GUI application. To control Beaglebone Black, there is BBB drivers block. It allows working with USB, UARTs, GPIO pins, ADC and timers of the development board. UGLD drivers support to control peripherals that are required for detector tests. For example, it is power supply controller or Real Time Clock (RTC) driver. Running, stopping and forwarding test information is done by a Test manager block. It uses test cases which are located in Test buffer. To generate the necessary signals DAC driver in STM32F051K6 is presented. For completeness, there is also a general detector software architecture consist of UGLD software and block for FP2 communication. This realization of architecture allows switch to testing another type of product. This is achieved by changing product drivers block (UGLD drivers) and possibly modifying the extension board. Communication, tests control and basic drivers for development board remain the same and thus form a fixed core.

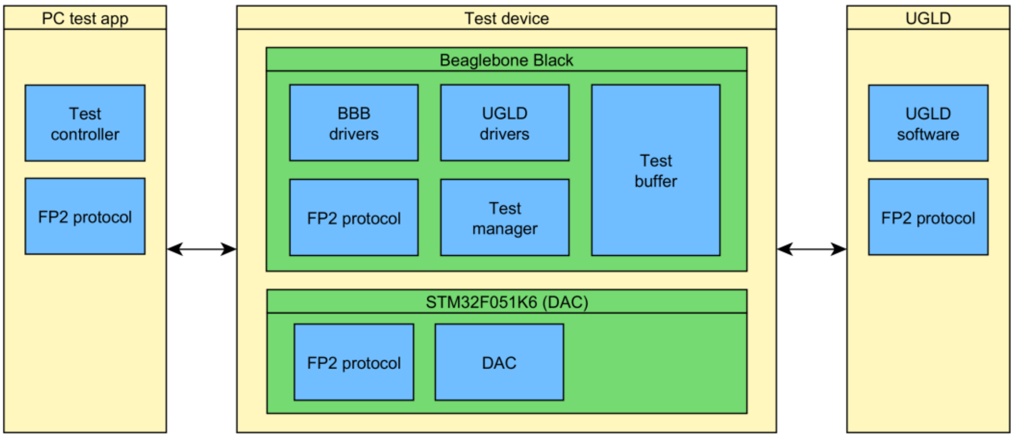
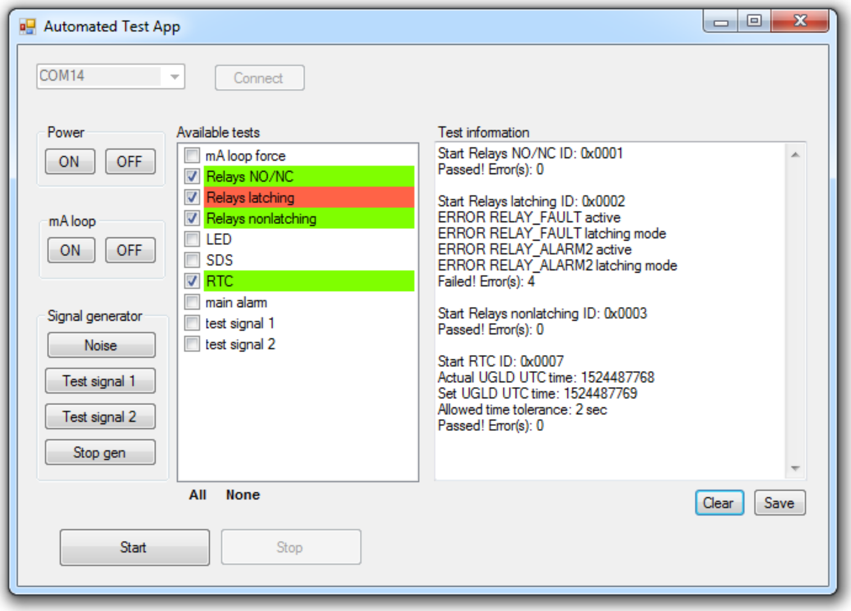


Figure 4: Software architecture of the testing device

2.2 GUI APPLICATION FOR TEST CONTROLLING

In Figure 5 there is graphic user interface application that allows controlling the device for testing. A user can select only tests which need to be done in "Available tests" section. Then by using "Start" button are the tests executed. A test progress is displayed in "Test information" section and information about success is passed in the form of a colored test name, green for passed and red for failed test. The test information can be stored or cleared by using buttons "Clear" or "Save". There is a possibility of prematurely ending tests with the help of a button "Stop". There is also a panel for selected tasks such as turning the detector power on / off.



□ mA loop force
☑ Relays NO/NC
☑ Relays latching
☑ Relays nonlatching
□ LED
□ SDS
☑ RTC
□ main alarm
□ test signal 1
□ test signal 2

```
Start Relays NO/NC ID: 0x0001
Passed! Error(s): 0

Start Relays latching ID: 0x0002
ERROR RELAY_FAULT active
ERROR RELAY_FAULT latching mode
ERROR RELAY_ALARM2 active
ERROR RELAY_ALARM2 latching mode
Failed! Error(s): 4

Start Relays nonlatching ID: 0x0003
Passed! Error(s): 0

Start RTC ID: 0x0007
Actual UGLD UTC time: 1524487768
Set UGLD UTC time: 1524487769
Allowed time tolerance: 2 sec
Passed! Error(s): 0
```

Figure 5: GUI application for test controlling

When the GUI is run, names and IDs of tests are downloaded. The process of downloading is shown in Figure 6. The PC sent command “Get test list”, test device acknowledge the packet and starts sending names and IDs of tests. End of the list is indicated by ID = 0xFFFF and test name “End”. After that the names of available tests will be displayed.

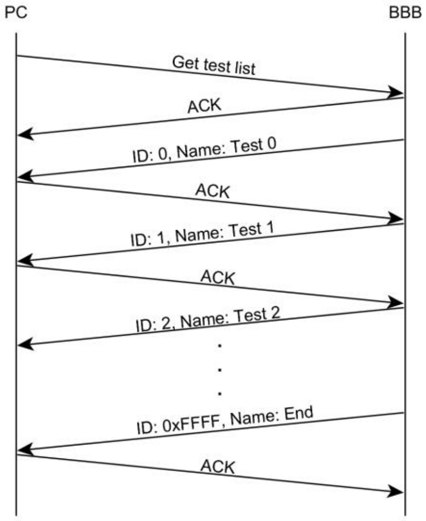


Figure 6: Downloading of test list

There are examples of communication in Figure 7 when a test is run and when a test is stopped. On the left picture packet with command “Run test, ID 0” is sent by PC. The testing device acknowledges the packet and starts the test. During the test execution there can be sent test information packets which the PC acknowledges. Finally, a packet with “End of test” and number of error message is sent. After acknowledge the PC runs next test. On the right picture there is a situation when a test is run but a user wants to stop the test. For that a packet with command “Stop test” is sent. This packet is not acknowledged. The acknowledgment is done like send a packet “End of test” with number of errors = 0xFFFFFFFF that is an indicator of force stop.

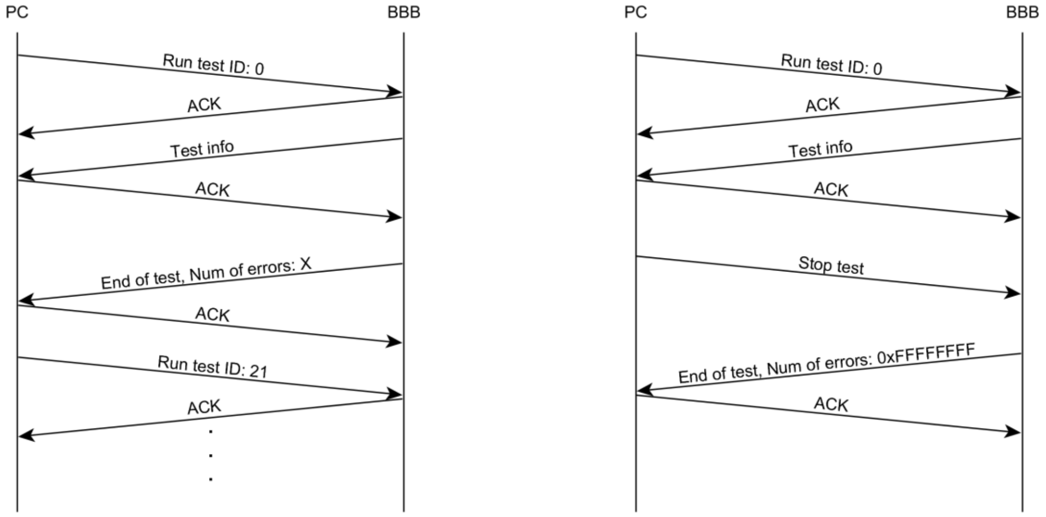


Figure 7: Example of communication when the test is run (on the left) and when the test is stopped (on the right)

2.3 CREATING TEST CASE

Test case is created as function in C language which is located in Beaglebone Black Test buffer block. It is written into function that has 32-bit unsigned return value which represents number of errors. In this function UGLD drivers should be used for detector and Beaglebone Black controlling. An error is detected using if(…) condition. When error is detected a message will be sent using a function of the test manager block. For delay instruction there is a macro that realizes the delay and at the same time reads serial line from PC to check request for force stop command.

3 CONCLUSION

In this paper a device for automatic testing was presented. The device consists of development board Beaglebone Black, extension board and PC for test control and was designed for testing an ultrasonic gas leak detector. The proposed software architecture can be adapted and used for testing of another product. In last subsections GUI application for test controlling, principle of communication and creation of new tests were introduced. Possible improvement of the testing device would be to move the tests to a computer and use the development kit only to control GPIO pins and signal measurement. This would allow writing tests directly on PC, which can be much more convenient. Another option would be to use the development board as a server that would allow controlling the tests through an Internet browser. Therefore, there would be no GUI applications on a PC, and testing could be done from a mobile phone, for example.

ACKNOWLEDGEMENT

The research was financially supported by the FEKT-S-17-4707.

REFERENCES

[1] Honeywell. *GasBook* [online]. 2013 [cit. 11. 11. 2017]. Dostupné z URL: <https://www.honeywellanalytics.com/~/media/honeywell-analytics/ documents/english/11296_gas book_v5_0413_lr_en.pdf?la=en-gb>.

[2] *ULTRASONIC GAS LEAK DETECTION TECHNOLOGY* [online]. Gassonic [cit. 11. 11. 2017]. Dostupné z URL: <http://www.gassonic.com/technology/the_technology/>.

[3] *About Ultrasonic Gas Leak Detectors* [online]. Emerson, c2017 [cit. 11. 11. 2017]. Dostupné z URL: <http://www.emerson.com/en-us/automation/measurement-instrumentation/flame-gas-detection/about-ultrasonic-gas-leak-detectors>.

[4] SKRYJA, Petr. *Automatické testování detektoru úniku plynu*. Brno, 2018, 71 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Vedoucí práce: Ing. Marek Novák.