INDONESIAN JOURNAL OF INNOVATION AND APPLIED SCIENCES (IJIAS)
Journal Homepage: https://ojs.literacyinstitute.org/index.php/ijias
ISSN: 2775-4162 (Online)
Research Article

Volume 2	Issue 2	June (2022)	DOI: 10.47540/ijias.v2i2.506	Page: 98 – 104

Design and Implementation of High-Precision Panel for Multichannel Micro-Flow Gas Monitoring in Oil Exploitation

Weilun Tang[1*], Jian Liu[1], Yangjie Deng[1], Junli Yi[1], Yuchen Lu[1]
[1]School of Electrical and Information Engineering, Wuhan Institute of Technology, China
Corresponding Author: Weilun Tang; Email: tangwl@stu.wit.edu.cn

ARTICLE INFO

Keywords: AD Sampling, Gas Flow Panel, STM32.

Received : 17 April 2022
Revised : 30 May 2022
Accepted : 03 June 2022

ABSTRACT

Gas leakage is a common occurrence in crude oil exploitation, and may cause devastating effects or at least put the health of workers at stake if not managed properly. To address this issue, a type of high-precision panel for multichannel micro-flow gas monitoring was proposed in this paper, which can be applied to a range of scenarios including the monitoring of gas leakage in oil exploitation. Different channels for gas-flow sensors are provided on the panel, to monitor the gas flow in multiple places at the same time. The menu is enhanced with a multilevel design, which makes it possible for the functions to be customized. The research method of this paper followed these procedures: First, hardware design was illustrated. Next, the software solution was specified in the paper. Finally, based on theoretical analysis, laboratory experiments were successfully carried out, which demonstrated that the performance of this design was in line with our scheme. Overall, this design can help conduct early warning in the case of gas leakage and remind the operators to take timely response, which can significantly ensure the safety of crude oil exploitation.

INTRODUCTION

Countries around the world are tightening their regulations in the energy sector in the context of some geopolitical crisis and the skyrocketing oil price it incurred. Some countries are accelerating their pace in shifting to renewable energy due to this wave of a global energy crunch. Despite these movements, we should still ensure an adequate supply of crude oil, due to the fact that most types of renewable energy is still at the initial stage of development and is far from being applied in all scenarios. Therefore, crude oil still remains our top priority among the various forms of energy. According to a report from the United States Energy Information Administration, 98.3 million barrels per day of petroleum and liquid fuels were consumed globally in March 2022.

Given the ongoing large demand and consumption of oil, there will still be a number of staff working in the oil fields. In the exploitation of oil, incidents of gas leakage occasionally occur, which are detrimental to the health of workers and may even incur devastating explosions if not monitored properly. To refine the procedures in exploitation and ensure the facilities are in normal operations, it is necessary to monitor the real-time gas flow in some key parts, which can remind us to take timely responses when there is an excessive level of gas leakage.

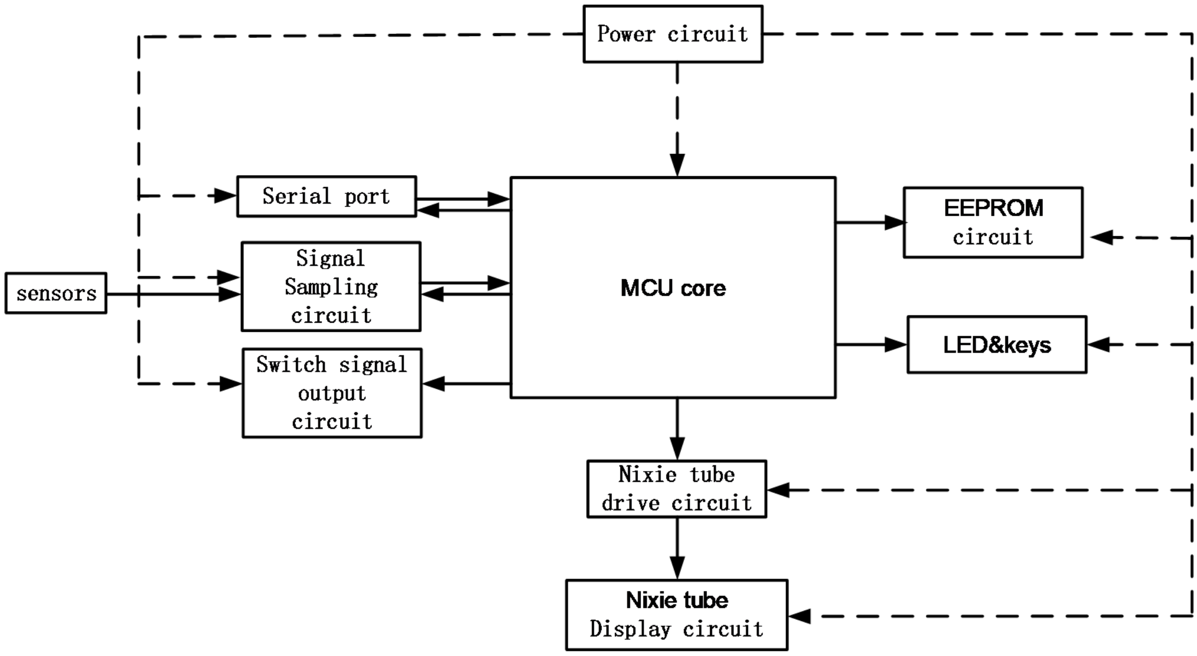
There are some existing designs of gas flow panels in the markets, however, most of these designs only have a single channel, which means more than one panel is needed when we monitor the gas flow in several different places. In addition, the majority of the existing designs in the markets are only compatible with large flow gas, and when we use them to measure micro-flow gas, they either make an inaccurate measurement or run into errors. Apart from that, the functions of existing designs are very limited. Most existing designs can display no more than the gas-flow figures.

To meet the needs mentioned above and address the constraints in existing designs, a type of multichannel and high-precision gas flow panel based on STM32 microcontroller was proposed in this paper, which can sample multichannel gas-flow data, and is compatible with the detection of micro-flow gas. Moreover, this design enriched the menu, which made it possible for a range of items to be customized.

METHODS

Hardware design

Figure 1. Hardware diagram (source: my own vision project)

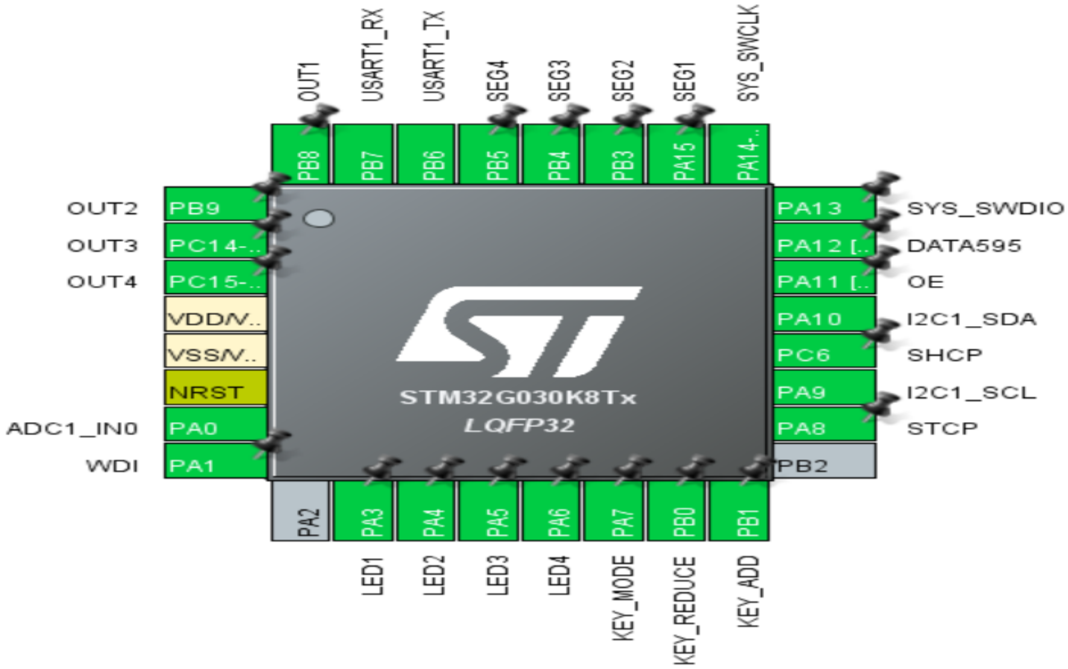


In general, the hardware circuit of this design consists of the following modules: The STM32 core and its peripherals, the power circuit, the EEPROM circuit, the LED lights and keys, the nixie tube drive circuit and its display circuit, the switch signal output circuit, the signal sampling circuit, and the serial port circuit. Apart from these modules, gas flow sensors should be installed on the corresponding channels.

Each of the modules mentioned above has its own functions. The MCU core refers to a minimum CPU circuit based on the STM32G030KxTLQFP32 chip designed by STMicroelectronics, which is a kind of microcontroller based on ARM Cortex-M0+. Just like the most commonly used STM32F1 and STM32F3 series, the STM32G0 series has a range of advantages such as a low level of consumption and rich built-in resources. In this design, the STM32 and its peripherals are in charge of data storage and processing, which means the STM32 will process the sampled data, release switch signals, drive the nixie tube circuit, judge which key is pressed, and interchange with the EEPROM circuit.

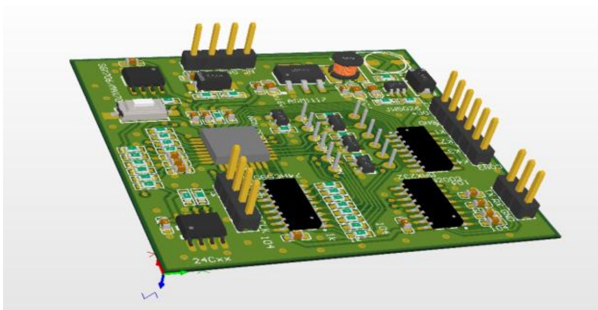
The power circuit provides a power supply for the system. The initial input voltage level is 24V, and after the transformation in the converter circuit, it can be lowered to levels of 3.3V and 5V. These three voltage levels provide power for different modules respectively; The EEPROM circuit is Electrically Erasable Programmable Read-only Memory; The LED lights are used as switch output indicators and error output indicators, and the keys are for the users to operate the menu and execute the corresponding functions. It should be noted that the switch signals indicators and error indicators can only work with the support of the switch output circuit, which can transform the analog signals into switch signals that the STM32 core can process.

Figure 2. Configuration of IO ports (source: my own stm32 Cube MX project)



The signal sampling circuit in this design is actually embedded in the STM32 chip, but to substantially implement its functions, we designed the interface as well as its peripheral circuit to install the sensors. The signal sampling circuit is actually AD converter that can transform the gas flow value the sensors detected into digital signals that the STM32 core can process.

Figure 3. PCB layout in Altium Designer (source: my own Altium Designer project)



The nixie tube drive circuit and display circuit are in charge of displaying the options in the multilevel menu. After analyzing the modules above, we configured the IO ports, as shown in figure 2, and designed the PCB layout using Altium Designer, as presented in figure 3.

Software design

To help the board fully perform its functions, the software program is indispensable. It is only when we download the software program into the STM32 microcontroller that all the data-processing procedures can be executed.

One critical part of software design is the multilevel menu. With a cyclic switching pattern, the menu in this design can be operated with only 3 keys. Key1 can be used to switch to the next option, and key2 can be used to switch to the previous option. The function of key3 varies according to different duration of press time. When it is less than 2 seconds, key3 serves as the enter button, and when it is more than 2 seconds, it serves as the quit button (return to the upper tier). The first tier of the menu includes the options of SET1, SET2, SET3, and SET4, which is switched using key1 and key2, each of these 4 options contains its sub-options, which can be accessed by pressing key3 (enter), and these sub-options forms the second tier of the menu. The sub-options serve as entries into some corresponding functions, and these functions can be categorized as the third tier of the menu. After entering the functions, there is another group of sub-options, which can be considered the fourth tier of the menu. It should be noted that the switching operation in the same tier is in a loop. When the operator reached the last option in the tier and press key1, it will jump to the first option. When the current option is the first option in the tier, and the operator presses key2, it will jump to the last option.

The workflow of the software program is specified in figure 4. It is divided into the main program and the interrupt program. In our design, the main program is mainly used to store some code for system initialization and keys configuration, while the major content, including the AD sampling algorithm, is included in the interrupt program.

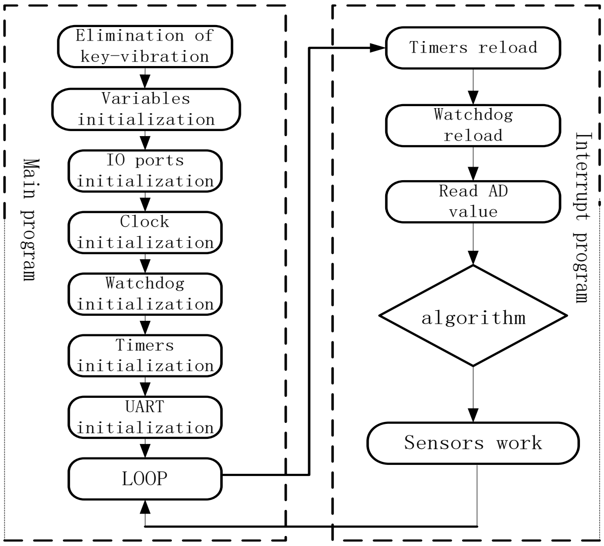
Table 1. Virtual Register 1

bit 15	
bit 14	Zero drift flag bit
bit 13	Display EF-1、 EF-2 EF-1: enable zero drift processing EF-2: not to enable zero drift processing
bit 12	Display Sd-1、 Sd-2、 Sd-3、 Sd-4。
bit 11-9	Output delay control bit 000 ： 0ms ； 001 ： 2ms ； 010 ： 20ms ； 011: 100ms ； 100: 1000ms ；
bit 8-5	Flag bits of 4 output channels
bit 4	Display REF1、 REF2、 REF3、 REF4。
bit 3-0	Output control bits

Table 2. Virtual register 2

bit 7	Filter value display bit
bit 6	Filter flag bit
bit 5	PF-1: enable the filter PF-2: not to enable the filter
bit 4	Peak holding flag bit
bit 3	Peak holding display bit ： PH-1: peak holding PH-2: real-time display
bit 2	Display dT-1、 dT-2、 dT-3
bit1-0	00: default 01: set update time as 250ms 10: set update time as 500ms 11: set update time as 1000ms

Figure 4. Software flow diagram



To display the options in the fourth tier of the menu, we defined two variables as the virtual registers, and we use the bits of the virtual registers to control the corresponding items.

To specify the scenarios in which this design can be adopted, we collected the output portrait curve of 5 gas flow sensors of different ranges. The output portrait curve is basically the curve line from a range of sample points which shows the relationship between the quantity of gas flow the sensors measured and the level of voltage they output. We extracted these sample points from the curve and with the help of the curve-fitting function in Microsoft Excel, we transformed these discrete points into formulas, and this enabled us to specify the code and algorithms in the software program.

The formulas we got which depict the relationship between the quantity of gas flow the sensors detected and the level of voltage they output are as follows:

For the sensors in a range of -3 to +3L/min:

$$V_{flow} = 0.0002U^6 - 0.0165U^5 + 0.2119U^4 - 0.8308U^3 + 0.2372U^2 + 4.7475U - 7.2955$$

For the sensors in a range of 0 to +3L/min:

$$V_{flow} = 0.0002U^6 - 0.0042U^5 + 0.0421U^4 - 0.1621U^3 + 0.1494U^2 + 0.5306U - 7.0493$$

For the sensors in a range of -0.5 to +0.5L/min:

$$V_{flow} = 0.0023U^6 - 0.0418U^5 + 0.3074U^4 - 1.1346U^3 + 2.3339U^2 - 2.3316U + 0.8674$$

For the sensors in a range of 0 to +0.5L/min:

$$V_{flow} = -0.0007U^6 + 0.0122U^5 - 0.0865U^4 + 0.31U^3 - 0.5752U^2 + 0.5713U - 0.2299$$

For the sensors in a range of 0 to +10L/min:

$$V_{flow} = 0.0741U^6 - 1.276U^5 + 8.8495U^4 - 31.291U^3 + 59.171U^2 - 55.872U + 20.3567$$

The following code example which indicates the selection of the -3 to +3L/min sensor can give a detailed look at how these formulas are applied in our software program:

```
if((flow_reg&0x07)==flow_con_1)
{
    if(Threshold_value== -3)
    {
    flow_value=0.0002*(ad_vin)*(ad_vin)*(ad_vin)
    *(ad_vin)*(ad_vin)*(ad_vin)-
    0.0165*(ad_vin)*(ad_vin)*(ad_vin)*(ad_vin)*(a
    d_vin)+0.2119*(ad_vin)*(ad_vin)*(ad_vin)*(ad
    _vin)-
    0.8308**(ad_vin)*(ad_vin)*(ad_vin)+0.2372*(a
    d_vin)*(ad_vin)+4.7475*(ad_vin)-7.2955;
    flow_value=(flow_value)*3.3/5;
    }
}
```

Results and Discussion

In an effort to verify the effectiveness of our design, we carried out laboratory experiments based on the analysis above. The realization of the multilevel menu is presented in figure 5 to figure 12. For lack of all 5 types of sensors, we used DC stabilized power supply to simulate the sensors. Figure 13 to figure 15 shows how the panel displays the AD value by the voltage that the signal sensors send to the STM32 microcontroller.

Figure 5. LED display update time setting (source: experiment)

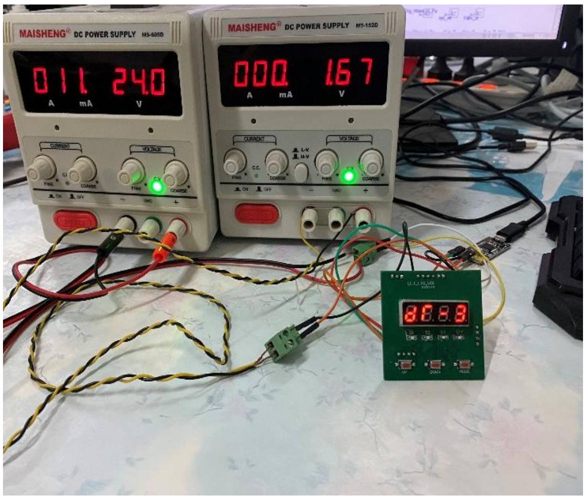


Figure 6. Selection of sensors (source: experiment)

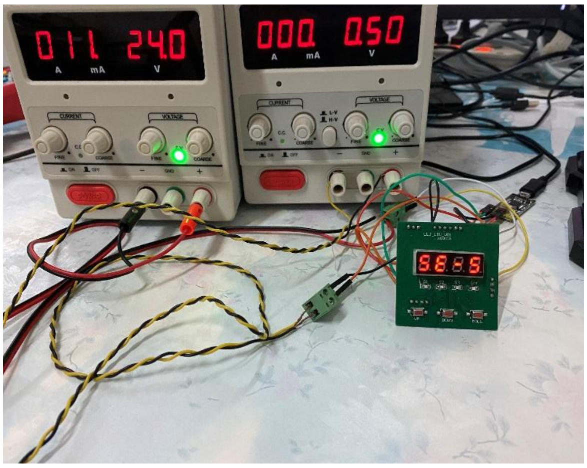


Figure 7. Peak holding setting (source: experiment)

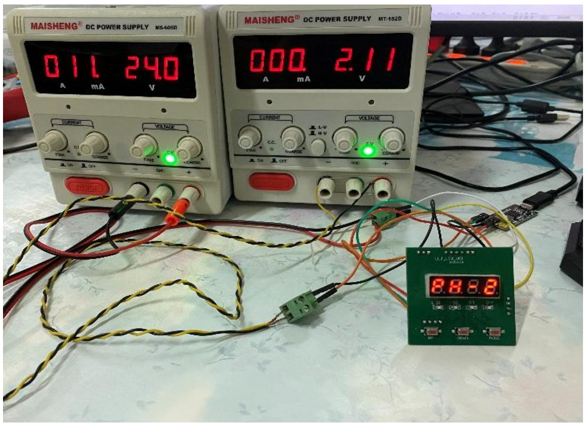


Figure 8. Threshold value selection (source: experiment)

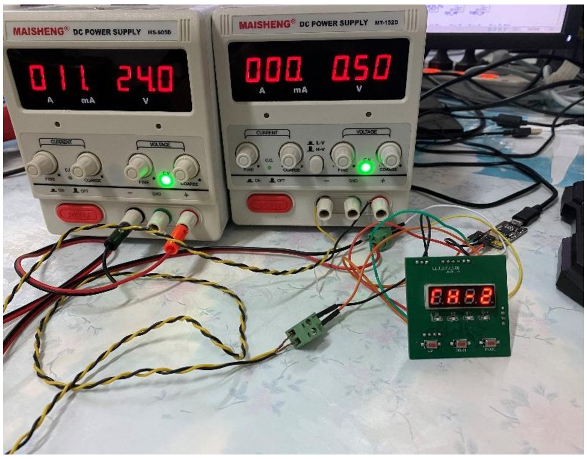


Figure 9. PV filter setting (source: experiment)

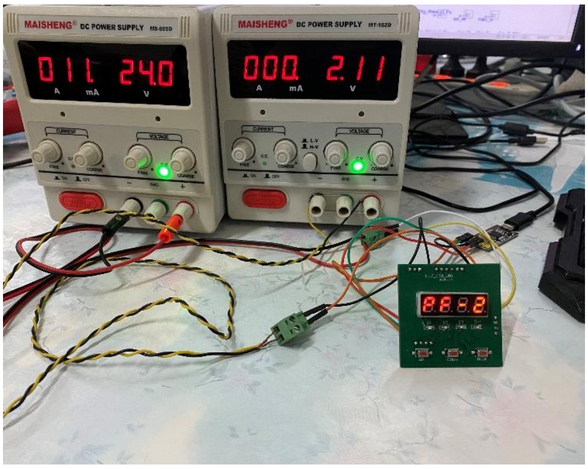


Figure 10. Output delay setting (source: experiment)

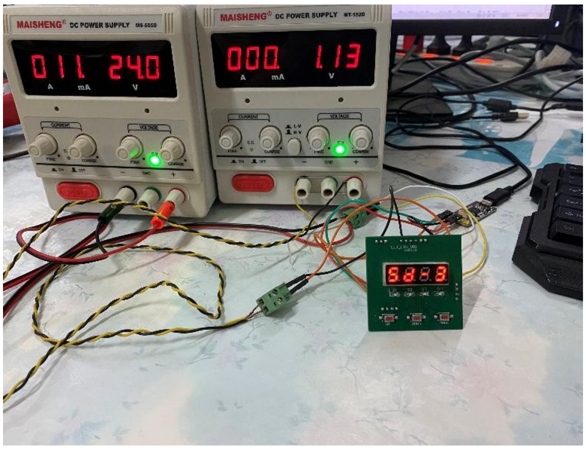


Figure 11. Switch values output flip (source: experiment)

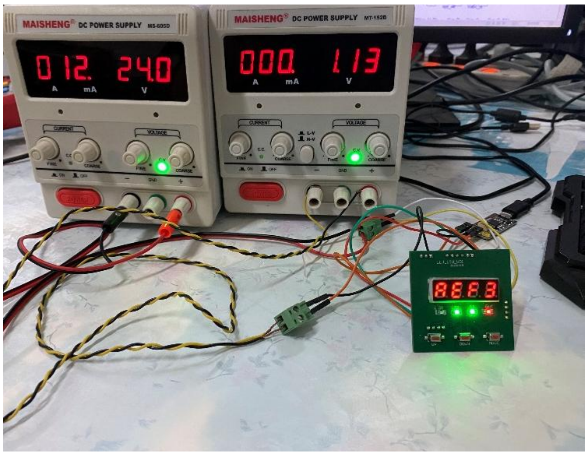


Figure 12. Zero drift setting (source: experiment)

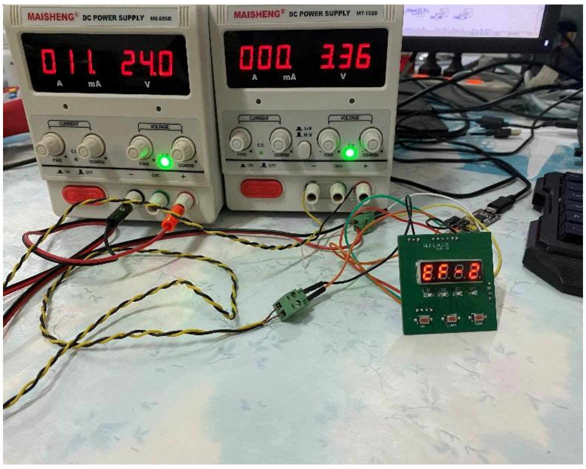


Figure 13. AD sampling value under a 5V voltage (source: experiment)

Figure 14. AD sampling value under a 3.3V voltage (source: experiment)

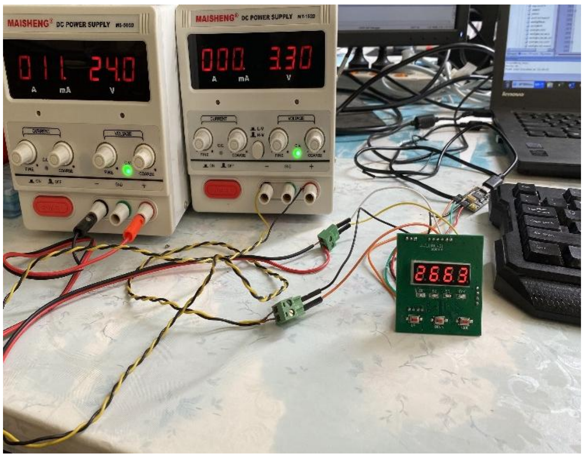
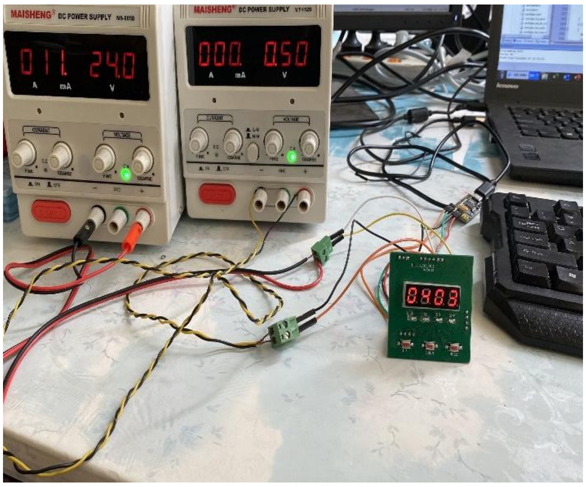


Figure 15. AD sampling value under a 0.5V voltage (source: experiment)



CONCLUSION

In this paper, a type of high-precision panel for the measurement of multichannel gas flow was proposed, which can be applied to a range of industries including crude oil exploitation. After hardware and software analysis, a laboratory experiment was successfully carried out, which verified the effectiveness and feasibility of this design. The innovative points and distinctive features in this design are as follows:

1. Multiple channels are provided for the measurement of gas flow, which enables operators to monitor gas flow in different places on a single panel.
2. This design is applicable to the measurement of micro-flow gas, which makes it possible to be adopted in a broader range of scenarios.
3. The menu is enhanced with a multilevel design, and a range of functions and settings are accessible in the menu. Instead of turning to an LCD screen alternative, we adopted the LED Nixie tube approach, which considerably lowered the cost of the design.

REFERENCES

1. Baron, C., Jean-Claude Geffroy, & G Motet. (2011). Embedded system applications. Springer.
2. Biantoro, A. W. (2020). GLEDS (Gas Leakage Early Detection System) Prototype for Early Detection of Gas Leaks Based on Microcontroller on Motor Vehicles. *Jurnal Teknik Mesin*, 9(1), 1.
3. Chen, J., Zhang, K., Wang, L., & Yang, M. (2020). Design of a High Precision Ultrasonic Gas Flowmeter. *Sensors*, 20(17), 4804.
4. Duren, R. M., & Miller, C. E. (2011). Towards robust global greenhouse gas monitoring. *Greenhouse Gas Measurement and Management*, 1(2), 80-84.
5. Fuller, S. H., & Gatherer, A. (2005). RapidIO: the embedded system interconnect. Wiley.
6. Gallagher, J. E. (2006). Natural gas measurement handbook. Gulf Pub.
7. Johannes Fink. (2021). Petroleum Engineer's Guide to Oil Field Chemicals and Fluids. Gulf Publishing Company.
8. Kernighan, B. W., & Ritchie, D. M. (2015). *The C programming language*. Pearson.
9. Lim, J.-S. (2016). Design of High Speed Data Acquisition and Fusion System with STM32 Processor. *Journal of the Korea Convergence Society*, 7(1), 9–15.
10. Miller, L. H., & Quilici, A. E. (1987). *C programming language: an applied perspective*. John Wiley & Sons, Cop.
11. Morris, J. M., & Xin, Y. (2012). Results on Gas Detection and Concentration Estimation Via Mid-IR-Based Gas Detection System Analysis Model. *IEEE Sensors Journal*, 12(7), 2347–2354.
12. Rigelsford, J. (2001). Smart mass gas flow sensor. *Sensor Review*, 21(3).
13. Sestoft, P., & Niels Hallenberg. (2017). Programming language concepts. Springer.
14. Spilsbury, R. (2014). The oil industry. Wayland.
15. Xu, W., Zhang, T., Bi, Y., Liu, W., & Xin, L. (2014). Analysis the compressibility impact of gas on vortex flowmeter measurement performance. *Journal of Electronic Measurement and Instrument*, 27(9), 797–802.