

MULTIPLEXER USED FOR MEASURING TEMPERATURES IN A TOF DETECTOR*

ADAM KUBIELA^{a,†}, KRYSTIAN ROSŁON^{b,c}, MARCIN BIELEWICZ^{c,d}
DANIEL DABROWSKI^{b,c}, MAREK PERYT^{b,c}

^aUniversity of Warsaw, Pasteura 5, Warszawa, Poland

^bWarsaw University of Technology, Pl. Politechniki 1, Warszawa, Poland

^cJoint Institute for Nuclear Research, Joliot-Curie 6, Dubna, Russia

^dNational Centre for Nuclear Research, A. Sołtana 7, Otwock-Świerk, Poland

(Received July 23, 2018)

This work presents the development of a multiplexer device used for multipoint temperature measurement system. The multiplexer consists of an 8-bit shift register driven by an AVR microcontroller. A 2-bit prototype was constructed and tested with a Slow Control dedicated temperature sensor, and a printed circuit board for the 8-bit version was designed as well. Furthermore, the multiplexer approach to multi-sensor measurements is discussed and compared to another solution, the parallel measurement approach.

DOI:10.5506/APhysPolBSupp.11.739

1. Introduction

Temperature measurements are an essential part of the Slow Control System (SCS) [1, 2], such as a control system for MPD-NICA collider [3–5]. With temperature sensors, one can monitor the operation of electronics within the system and detect possible faults resulting in overheating or, in worst case, fire. Multipoint measurements might also be used in monitoring heat generation and distribution, for example in a Quinta assembly [6, 7]. In order to work properly, multiple temperature sensors are scattered around the whole system, and data acquisition must be able to handle all of them. The simplest solution is to assign every single sensor to its own ADC and have them connected to data acquisition system at the same time (Fig. 1(a)). Unfortunately, this can become expensive when a lot of sensors are connected. Since at least 40 temperature sensors can work in

* Presented at the II NICA Days 2017 Conference associated with the II Slow Control Warsaw 2017, Warsaw, Poland, November 6–10, 2017.

† Corresponding author: adam.kubiela@fuw.edu.pl

SCS, a different approach is studied. In the second solution, only one ADC is assigned to all sensors, and a switching device is used to connect only one sensor at a time to the ADC (Fig. 1(b)).

The challenge is to demonstrate which solution is more applicable to SCS — the parallel, multi-ADC approach or single ADC, switching multiplexer approach. The aim of this work was to design and construct a multiplexer prototype and test it in a simple multipoint temperature measurement system. The results determined strong and weak sides of this solution.

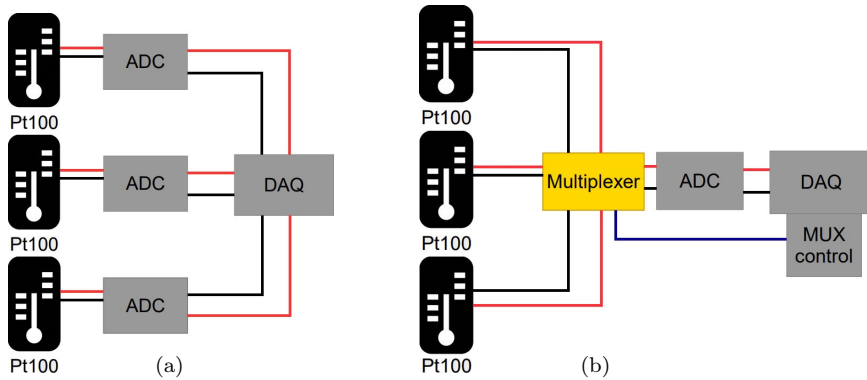


Fig. 1. Two approaches to control multiple sensors. This paper examines approach (b).

2. Multiplexer design

2.1. Hardware

The proposed design uses relays to switch between temperature sensors and an 8-bit shift register to drive the relays. The device was designed as a 216×126 mm board. Using a shift register instead of a multiplexer chip allows the device to be scalable — modules can be daisy chained to create more channels on the switching device.

The central part of the multiplexer is an Atmel ATmega328P microcontroller [8]. It handles communication with external controlling device, drives shift registers and monitors sensor collisions and responds if such a collision occurs. The multiplexer uses serial communication via UART protocol module. Using popular external converters, one can also communicate with the device using RS232 or USB protocols. This allows the device to be easily controlled using a PC. ATmega328P can also handle SPI and I2C protocols, although they are not used in the design. It is still possible to enable those ports in future versions of the device.

Relays were driven by the 74HC595 chip, 8-bit serial-in, serial or parallel-out shift register [9]. It receives serial data from the microcontroller and sets its 8 outputs accordingly to input data. Outputs of the 74HC595 can be set

as HIGH-state, LOW-state or high-impedance OFF-state. In the design, switches are active-low, therefore, a pull-up resistors on each output of the shift register make sure all switches are disabled when 74HC595 is in high-impedance state.

By design, 74HC595 allows for daisy chaining multiple chips, effectively creating a larger shift register (Fig. 2). This is a crucial feature for scalability of the multiplexer. Daisy chaining of the multiplexers occurs in master/slave mode. Master device is the one accepting commands from the PC or another controller. It is also the only multiplexer in chain with the microcontroller present on board. In the design, microcontrollers are intentionally connected to the board via a socket rather than soldered directly. This way, presence of the microcontroller acts as a master/slave switch and no other switches are required. Slave devices are effectively extensions of both the shift register and sensor collision detector. Connecting multiple slave devices has not been tested yet, so maximum number of slaves is unknown. The main restriction here is the time of switching and measuring. The more outputs the multiplexer has, the more time it takes to measure with all sensors.

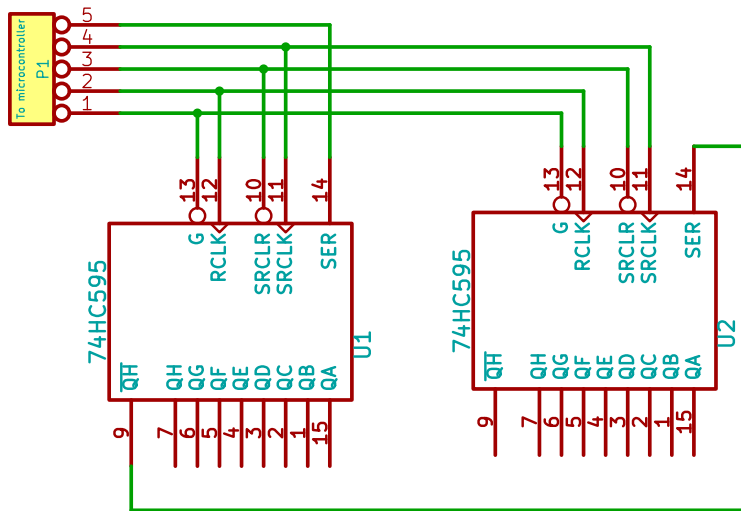


Fig. 2. Example of daisy chaining of two 74HC595 shift registers. G, RCLK, SRCLR and SRCLK inputs are respectively output enable, latch, master reset and clock pins and are common for all chained registers. SER pin is a serial input. In master, it is connected to the microcontroller, in slave, it is connected to the last output from the previous register.

Shift register outputs are connected to P-channel MOSFETs, acting as switches for relays (Fig. 3(a)). S14-2C-0505, double-pole, double-throw relays were used [10]. One of the switches inside the relay was connecting one

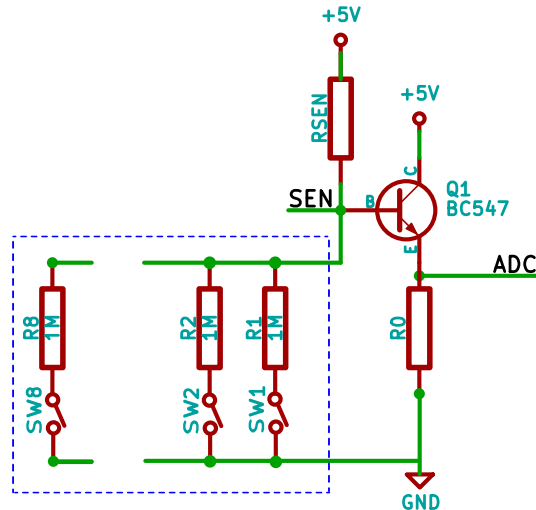


Fig. 4. Collision detection system of the multiplexer. Switches SW1–8 are driven by relays. Number of resistors R1–R8 determines the voltage at SEN terminal, which is measured by microcontroller’s internal ADC.

2.2. Firmware

Firmware for ATmega328P microcontroller was developed using the Arduino library [11]. During its operation, the ATmega reads collision sensor voltage and waits for data on serial port. If a number is received, ATmega enables a relay corresponding to that number while disabling other relays. If collision of sensors is detected, all relays are disabled and the device halts. At launch, all outputs of the shift register must be set, because switches are active-low. The microcontroller cannot determine the number of slaves connected, so in current version, it assumes 10 slaves connected, thus 80-bit multiplexer. This assumption does not have any effect on the device operation after setup. Firmware can be easily reprogrammed using an AVR programmer or an Arduino Uno board. A different code, not involving Arduino library at all, can also be implemented.

3. Prototype and testing

The prototype was a 2-bit version of the designed device. The only other difference was using an Arduino Uno board instead of a stand-alone microcontroller in order to connect it easily to the PC and shift register. The temperature sensors used in testing were Pt100 resistance thermometers. The signal converter was a LUMEL SM1 analog input module [12], whereas read-out was handled by a LUMEL SM61 data logger [13]. So far, sensors were proven to give stable readouts at a rate of 1 measurement every 10 seconds.

Occasional glitches appeared at a rate of 1 measurement every 5 seconds, although it is not certain whether it is the multiplexer's fault. Results might be also better with the device under test being an 8-bit prototype assembled on a designed printed circuit board rather than temporarily assembled 2-bit one. Further tests should be performed to determine the device's behaviour at higher rates.

4. Conclusions

This work has proven that the multiplexer approach to controlling multiple sensors can be a useful solution both in SCS and in Quinta experiment. Its main advantage is the necessity to use only one ADC to read all sensors, which, for large number of sensors, makes this solution cheaper than approach from Fig. 1(a). The designed switch is compatible with a few communication protocols making it universal. It can also be easily reprogrammed to fit user's needs and can work with a variety of sensors. The main disadvantage of this design is the delay between switching that makes simultaneous measurement impossible. The problem is more severe when a lot of sensors are connected. It is also more complicated and more expensive solution for a small number of sensors (less than 4). Furthermore, a collision detection system is programmed in the device's software what makes it less reliable than the hardware solution. Finally, the device itself cannot determine which sensor is currently active. These issues will be addressed in future versions of the multiplexer.

REFERENCES

- [1] D. Dabrowski *et al.*, *Acta Phys. Pol. B Proc. Suppl.* **9**, 203 (2016).
- [2] K. Roslon *et al.*, *Acta Phys. Pol. B Proc. Suppl.* **9**, 299 (2016).
- [3] V. Golovatyuk *et al.*, *Eur. Phys. J. A* **52**, 212 (2016).
- [4] NICA — Nuclotron Ion Collider Facility, available: <http://nica.jinr.ru/complex.php>, [Accessed: Apr. 2018].
- [5] M.J. Peryt, T. Traczyk, *Acta Phys. Pol. B Proc. Suppl.* **9**, 293 (2016).
- [6] W. Furman *et al.*, *PoS (Baldin ISHEPP XXI)*, 086 (2013).
- [7] E. Strugalska-Gola *et al.*, *EPJ Web Conf.* **138**, 10007 (2017).
- [8] Atmel, “8-bit AVR Microcontrollers”, ATmega328/P datasheet, Nov. 2016.
- [9] Texas Instruments, “8-bit Shift Registers with 3-state Output Registers”, 74HC595 datasheet, Dec. 1982 [Revised Sep. 2015].
- [10] K-SHOI, “General Purpose Relays”, S14-2C-0505 catalogue note.
- [11] Arduino Library, available: <https://www.arduino.org> [Accessed: Aug. 2017].
- [12] LUMEL, “SM1 Module of Analog Inputs”, datasheet, revised Feb. 2017.
- [13] LUMEL, “SM61 Data Logger with www Server”, datasheet, revised Jan. 2016.