JIEE Časopis priemyselnej elektrotechniky
Journal of Industrial Electrical Engineering
ISSN 2454-0900
www.jiee.eu

An universal IO peripheral device based on Arduino

[1]*Marek MARCIŠ,* [2] *Radoslav BUČKO,* [3]*Jozef DZIAK*

[1,2,3] Department of Theoretical and Industrial Electrical Engineering, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic

[1]marek.marcis@student.tuke.sk, [2]radoslav.bucko@tuke.sk, [3]jozef.dziak@tuke.sk

***Abstract* — This paper aim is to highlight the versatility, utilization of resources and the possibilities for small automation, specifically we will deal with the Arduino platform, as a simple programmable module that is readily accessible and is not costly. This module provides creation of an electronic circuit, a prototype, which serves as the controller, communicating with a PC. The computer program is created that is then transferred into the Arduino board and it carried us now programmed commands. We will deal with universal input-output device for switching and control other external devices. The principle is based on the Arduino, which can be directly connected to the PC. An Arduino controls the entire device and control. The hardware part consists of several relays and transistors.**

***Keywords* — Arduino, general purpose, IO peripheral, relay**

I. INTRODUCTION

Nowadays, without electronics easily we moved. The foundation of any programmable electronic devices is microcomputers or microcontrollers used to control electric circuits and implementation of programmed commands us. At the moment, without these components cannot imagine "life" almost no electronic devices (cell phones, MP3 players, USB sticks, microwaves ...). This component forms the basis for the Arduino platform which is an essential element of the kit, which is a controller universal input-output (IO) device (hereinafter referred to as UIOD), which we will discuss in this article. The whole device is composed of three parts: Arduino, PCB (Printed Circuit Board) and software. Arduino provides a wide range of applicability in technology, whether in the domestic or industrial applications to control and process automation.

II. ARDUINO PLATFORM

Arduino is a universal platform based on a simple control hardware and software. Specific board Arduino MEGA 2560 series represents the greater Arduino boards. A powerful microcontroller ATmega 2560 containing 256kB Flash memory (8kB bootloader), 8 kB SRAM, and 4KB EEPROM. Also on this board we find the 54 I / O digital pins, where 15 is supported by the PWM output further 16 analog input pins. For these pins can be connected more other components and equipment, which will also be applied in this solution. Individual pins operating voltage of 5 V and can deliver a maximum current of 40 mA. Power is supplied by USB interface. Using this interface also Arduino programs where the recorded program copied to the Arduino bootloader. Of course if the current USB interface exceeds 500 mA protection interrupts the connection to the PC. The optimal operating voltage is 7-12 V. If the voltage drops below 7 may be the case where the pin 5 is to supply less than 5 V, and thus becomes unstable board. On the contrary, at a voltage higher than 12V may occur overheat board or to its destruction. It is managed by the crystal oscillator with a frequency of 16 MHz. In the picture we can see the actual design of Arduino boards.

III. UNIVERSAL IO DEVICE USER REQUIREMENTS

According to these requirements, the entire device is designed. The result should be a universal input-output devices communicating with the PC and enable the generation of control signals from a PC or their loading. The device should be based just on the Arduino platform with the control element of the whole device. The resulting IO device should meet user requirements such as:

- Ease of Use
- Connectivity
- Compatibility
- Speed Communications
- Compact and dimensions
- Usability skills of Arduino

Easy to use - the entire device or module to be designed so that the options can be used and used by anyone with even partial knowledge from the field of electrical engineering.

Connectivity - the entire module to be immediately active and able to perform correctly given us orders. This means that when connected to a PC via USB interface should expect our response and be able to respond accordingly to the maximum extent possible.

Compatibility - It is mainly the operating system and therefore the software part. It shall be possible to control this versatile devices from different operating systems such as Windows, Macintosh or Linux.

The speed of communication - this requirement essentially relies on the Arduino platform in order to speed communication between PC and adjustable hardware part. It should be available for use communications software settings, for example. It will set the basic speed of communication, but using an appropriate order to be able to change.

Compact dimensions - since it is a universal device, it should not exceed A5 (148x210mm). It should contain 3x relay, 4 x transistor to control the PWM modulation decorated pins - female (Eng. Header female strip pins) and male (Eng. Header male strip pins), 3 terminals for output relátok, 4 terminals for transistor output, 1x reset button and 1 custom button (UB-User button), whose function is to be put to define.

Usability capabilities Arduino - Arduino choice of the type to be the most versatile. It must include the maximum possible number of digital and analogue pins. The digital pins laid emphasis on setting the input or output but also a sufficient number of pins to support PWM modulation. Another requirement is also tolerance of 5V logic.

IV. UNIVERSAL IO DEVICE PARTS IDENTIFICATION

A. Arduino board selection

For the design solution was chosen specifically Arduino MEGA 2560 because of the choice of this type are several. The first is that this type has a large number of input-output digital pins (54 pins), which is important for the input-output device, based on the universality. Another is that it comprises a plurality of support pins digital PWM (pin 15). Comparing the most universal type used Arduino UNO and Arduino MEGA 2560 to get a better idea of why it was used just this type. We use 4 transistors to control PWM, thus we type in Arduino MEGA 2560 remains another 11 pins, which we can use. However, if we use Arduino UNO, which has only a 6-Timi pins supporting PWM, would remain available if only two pins, which reduces the degree of versatility. On the other hand, this view could also be used Arduino DUE, but this type is characterized by high performance, which will be

solutions were needed. The most striking difference, however, is using 3.3V logic at Due. What is not sufficient, because of user requirements imply support for 5V logic. This requirement with respect to the foregoing aspects only performs MEGA 2560. Finally, the board DUE features a significantly higher price than MEGA 2560.

Fig. 1 Arduino MEGA 2560

B. PCB

PCB itself, on which are led pin (male) which simply connect specifically Arduino MEGA 2560 which contains pins (females). The entire kit to fit each other with ease. Using USB interface connect the Arduino to the PC, where there is a serial communication via UART, and then use a third part of the whole program, the entire device rules (discussed in the next section).

The whole device is designed in the program Proteus. First, you need to design the circuit diagram, which will be located on the printed circuit board, and we are based on user requirements. DPS is: 3x relay, 3 NPN transistors for switching relays 4 MOSFET transistors (IRF540N) to control PWM, 1 reset button, 1 user key (user button), decorated GND pins for better grounding, and of course decorated pin - male to female connection Arduino Arduino pins to steer the PCB. The entire design is shown in Fig. 2.

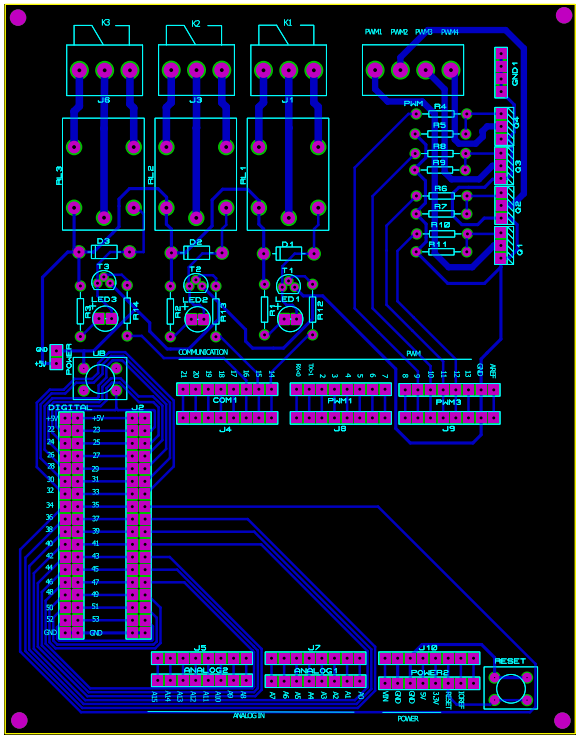

Fig. 2 PCB design background

V. SERIAL COMMANDS

With commands via the serial port, the individual components contained in the PCA (relays, transistors, and pins), switched and monitored. The term means the program software for Arduino. This whole program is designed in an environment Arduino IDE. The reason is the mutual compatibility Arduino with this environment.

A. Defined group of commands:

- Configure I / O pins
- Set the output - 0 (LOW) and 1 (HIGH)
- Reading input
- Set PWM
- Control commands

Configuration of input and output pins:

SET OUT PIN; - the first part of the SET OUT defines the command to set the specific PIN as digital output. PIN code is variable and specified an Arduino pin. It can take values 2-53.

SET IN PIN; - SET IN is a command to set the appropriate PIN as a digital input.

SET OUT / IN BYTE A; - Setting a series of predefined digital pins as output / input. This is the setting of the digital pins 22-53 which are hard-defined as constants (for direct changeovers 8 pins simultaneously). Pins assignment to bytes:

- BYTE A – pins 22-29
- BYTE B – pins 30-37
- BYTE C – pins 38-45
- BYTE D – pins 46-53

B. Output settings - values 0 (LOW) a 1 (HIGH)

PIN PIN_NUMBER VALUE - PIN command is used to adjust the values of output pins. *PIN_NUMBER* as a variable can take values from digital pin 2 to 53. The variable value has only two values, L and H. This command can be controlled for example switching relays. For build-in relays are reserved to their control three digital pins 14, 15 and 16 pin. Specific command "*PIN 14 H;*" (Assigning the value 1 to pin 14) will switch relay 1 labeled RL1. After entering the command "*PIN 14 L;*" (Assigning the value 0 to the pin 14), the relays contacts disengage.

BYTE VALUE - this command, you can set the values 0 and 1 for each byte of pins A, B, C or D conversion to decimal values of the binary number. The entered variable value is in decimal in the range 0 - 255. Program this number is converted into a binary condition and adjusts it according to individual pins byte. For instance, a value 15, "00001111" in binary form, indicates that the first 4 bits of the byte are set to 0 and the remaining 4 to 1. The first bit of the byte (= 0) represents MSB (Most Significant Bit. The last bit (value 1) represents the LSB (Less Significant Bit).

C. PWM command

PWM PIN VALUE - PWM is part of the command. In the PIN is a particular pin, which must qualify as support for PWM modulation - these are digital pins 2-13 and 44-46. Other pins cannot be set as a PWM output. The last part of the order VALUE shall be specified in a specific value range of 0-255 to be set. This command can be used with optimal engagement in which one of the transistors

Q1-Q4 is located between the ground and the load. The load is connected to an external power supply. By adjusting the PWM transistors are used four specific pins associated with transistors and PWM output terminals:

- Terminal block PWM1 – transistor Q1 – PWM pin 12
- Terminal block PWM2 – transistor Q2 – PWM pin 11
- Terminal block PWM3 – transistor Q3 – PWM pin 10
- Terminal block PWM4 – transistor Q4 – PWM pin 9

D. GET command

GET UB - the count value of UB (User Button), is pressed or not.

GET PIN - the count value of the digital pins. Since this is a digital pin, the value is either L or H (1 or 0).

GET A15 – the count value of the analog inputs. Arduino MEGA 2560 features a 16 analog input pins A0 - A15. For analogue inputs can expect a value between 0 -1023.

GET BYTE A – the count value of a series of 8-Michel digital pins. Again, you can use bits A, B, C and D. This command returns the decimal value of the binary numbers i.e. returns a value from 0 to 255. For example. Get the answer "100" so we know that the individual pins are set according to the binary number "01100100". Again, the first bit of the binary number is the MSB and the last is LSB.

VI. UNIVERSAL IO DEVICE CONSTRUCTION

As the most reliable method for PCB this creation was choose the photoresist way. The principle of DPS in this way is to spread paint on the photosensitive PCB copper surface. It is necessary to let the paint to dry thoroughly (24 hours at room temperature). The next step is to print a master transparencies. After drying, paint the artwork printed on the sheet is pressed against the side coated with varnish. Following illumination by the UV lamp. In this particular case, 90 seconds uniform lighting. After lighting the rest of the paint removed with NaOH. The last step is etched in ferric chloride $FeCl_2$ for 15 minutes. Created PCB according to design from Fig. 2 is shown in Fig. 3.

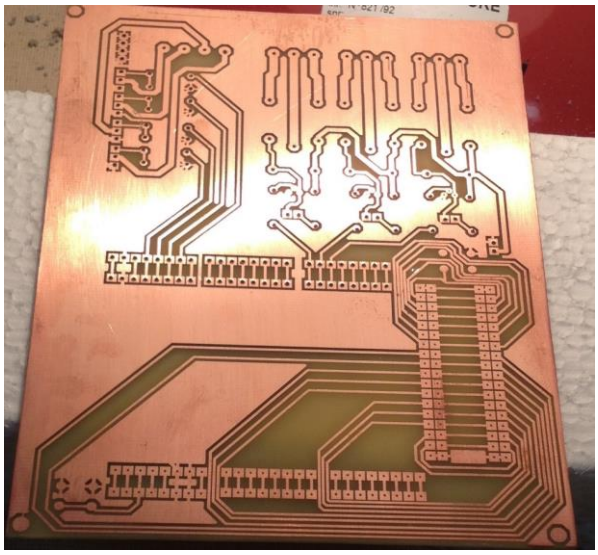

Fig. 3 Created PCB

The device contains all the components according to user requirements. The main components of an universal input-output devices are arduino, relays and transistors. Each contains 3 Relay contacts brought out in the form of a box:

- *NC* – Normally Close, it is used rarely.
- *COM* – this contact we bring stress that we want after switching relays to control other electronic components or circuits.

- *NO* – Normally Open, output contact. After switching relays will be linked contacts COM and NO, and therefore we can use voltage to the contacts COM NO auxiliary contacts and thus this voltage can control other circuits.

PWM output is also used in connector. Each one of the four transistors Q1-Q4 is output from just one contact in the form clamp. Terminals named PWM1-PWM4 serve PWM modulation and are connected with discrete transistors Q1-Q4. The final result is shown in Fig. 4.

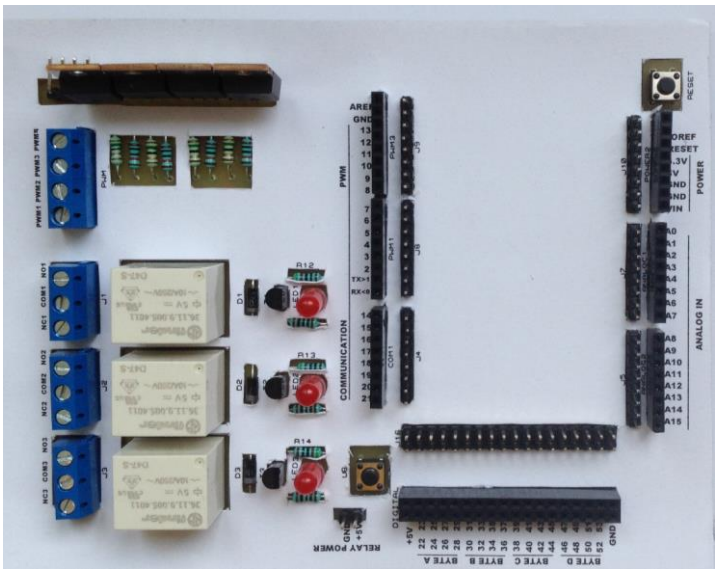

Fig. 4 Final universal IO device device without Arduino

VII. CONCLUSION

The advantage of the entire device is its simplicity and versatility. Using commands using Arduino can control all the equipment specific components on PCB, mainly transistors, used in important PWM or relays to switch on or off other electrical circuits.

Arduino offers great versatility and availability to solve small automation projects since it was created as a freely-accessible project. It is a popular and sources used for the production of prototypes for its extensive potential uses. Along with other devices or components using Arduino can create a variety of special electronic circuits.

REFERENCES

[1] Banzi, Massimo. Getting started with Arduino. ISBN 978-0-596-15551-3.
[2] BOŽÍK, M. Ako naprogramovať Arduino bez predchádzajúcich znalostí [online]. Ebook verzia 2013. [cit. 2016-4-17]. Available: http://mirobozik.sk/Media/Default/ebooks/mirobozik-ako-napr-arduino.pdf
[3] Arduino platform web page:: https://www.arduino.cc/
[4] Fotocesta: Domácí výroba desek plošných spoju. [online]. [cit. 2016-5-11]. Available on: http://www.elweb.cz/clanky.php?clanek=101
[5] USB relé a USB I/O moduly. [online]. 2013 [cit. 2015-1-10]. Available on: http://pandatron.cz/?3292&usb_rele_a_usb_- _i/o_moduly
[6] Využitie systému Arduino pri riadení procesov. [online]. 2012. [cit. 2015-1-10]. Available on: http://www.atpjournal.sk/rubriky/prehladove-clanky/vyuzitie-systemu-arduino-pririadeniprocesov.html?page_id=16348
[7] Jirásek, J. Vstupy a výstupy [online] Available on: http://ics.upjs.sk/~jirasek/arp/Vstupyavystupy.pdf
[8] Paralelný port LPT(IEE1284). [online]. 2005. [cit. 2016-1-5]. Available on: http://vyvoj.hw.cz/navrh-obvodu/rozhrani/lpt/hw-server-predstavuje-paralelni-port-lpt-ieee-1284.html