

Design and realization of electronic device for sports result representation

Petar Škomrlj¹, Hrvoje Radovan²

¹ Aspira University College – Split, Mike Tripala 6, 21000 Split, Croatia, pskomr00@gmail.com

² Aspira University College – Split, Mike Tripala 6, 21000 Split, Croatia, hradovan1@gmail.com

Sažetak

Ovaj rad sadrži opise projektiranja i izrade elektroničkih uređaja za unos, izračunavanje i prikaz sportskih rezultata (semafor). Za izradu ovog uređaja glavne potrebne komponente bile su: Arduino Nano mikrokontroler, integrirani krug MCP23017 (koji se koristi za prevladavanje nedostatka digitalnih ulaza na mikrokontroleru), integrirani sklop MAX7221 (koji se koristi za upravljanje zaslonima), tipkovnice i 7-segmentni zaslone. Očekivani način rada opisan je u sljedećim rečenicama. Uređaj očekuje unos s jedne od dvije tipkovnice. Kada je jedna tipkovnica aktivirana, moguće je samo read-only unos s te tipkovnice (uključujući unos koji daje kontrolu drugoj tipkovnici koja zatim preuzima). Dva središnja zaslona pokazuju broj koji treba dosegnuti za pobjedu, dok četiri prikaza rezultata pokazuju trenutni rezultat svakog igrača. Osim zbrajanja i oduzimanja vrijednosti prikazanih na zaslonima, moguće je dodavati i brisati vrijednost središnjeg zaslona, brisati vrijednost prikaza rezultata i prepustiti kontrolu drugoj tipkovnici.

Ključne riječi

semafor; Arduino; mikrokontroler; elektronički uređaj; integrirani krug

Abstract

This paper contains descriptions of designing and making electronic devices for entering, calculating, and showing sports results (scoreboard). To make this device, the main required components were: Arduino Nano microcontroller, MCP23017 integrated circuit (used to overcome lack of digital inputs on the microcontroller), MAX7221 integrated circuit (used to control displays), keyboards, and 7-segment displays. The expected way of work is described in the following sentences. The device expects input from one of two keyboards. When one keyboard is activated, it is possible to read-only input from that keyboard (including input that gives control to the other keyboard which then takes over). Two central displays are showing the number to reach for the win, while four result displays are showing the current result of each player. Besides addition and subtraction of values shown on result displays, it is possible to: add and delete the value of the central display, clear the value of the result display and give control to the other keyboard.

Keywords

scoreboard; Arduino; microcontroller; electronic device; integrated circuit

Introduction

Motivated by the high prices of production electronic scoreboards for displaying sports results, this paper shows how to build the electronic device for that purpose.

In general, an electronic device is a system of interconnected electronic components which, depending on connections between them, make a functional device. The components of the device are determined based on the expected behavior of the device. Moreover, when choosing components, it is important to analyze available components and their technical specifications to use optimal ones.

The microcontroller is an integrated circuit (in this case Arduino Nano [[2]]) that contains the processor, memory, and input/output peripheral components. The behavior of the microcontroller is defined by the program code which is loaded in it.

Interaction between user and microcontroller is enabled through keyboards (input device) and displays (output device).

1. Electronic components

This chapter describes the main components used to make an electronic scoreboard.

1.1 Arduino Nano

Arduino Nano [[2]] is based on an Atmel Atmega328/P microcontroller. It has 32KB flash memory, 2KB SRAM (Static Random Access Memory) and 1KB EEPROM (Electrically Erasable Programmable Read-Only Memory). It is powered over mini-B USB (Universal Serial Bus), an external 6-20V power supply on pin 30, or a 5V power supply on pin 27. It consists of 14 digital pins which can be used as input or output, support a maximum current of 40mA and have internal pull-up resistors. Also, it has 8 analog pins with 10-bit resolution.

FIGURE 1 shows full pinout, including the ones not already mentioned.

1.2 Maxim Integrated – MAX7221

MAX7221 [[3]] is an integrated circuit driver for 7-segment displays with a common cathode. To control with six 7-segment displays directly from Arduino [[2]], 42 digital pins are required (Arduino [[2]] only has 14). To overcome this, MAX7221 [[3]] is used to control displays. It communicates with Arduino [[2]] over SPI [[4]] (Serial Peripheral Interface) using only 3 pins.

Data from the microcontroller is written into MAX7221 [[3]] registers and then multiplexed to connect electrical circuits to light up required LEDs (Light Emitting Diode). FIGURE 2 shows MAX7221 [[3]] pinout.

FIGURE 1: ARDUINO NANO PINOUT

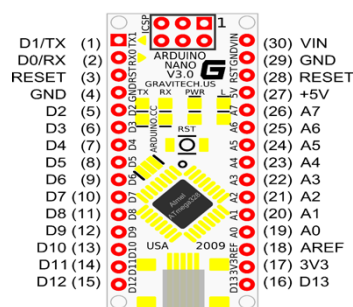
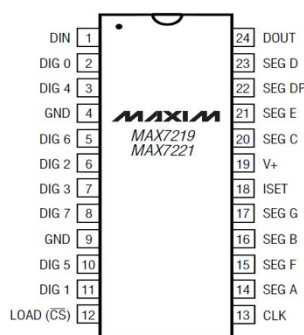


FIGURE 2: MAX7221 PINOUT

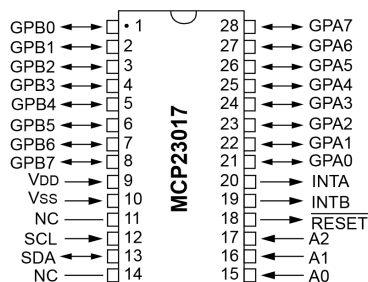


1.3 Microchip – MCP23017

MCP23017 [[5]] is an integrated circuit that is used to overcome the issue of lack of input digital pins. It communicates with Arduino [[2]] over I2C [[6]] (Inter-Integrated Circuit) serial communication. Depending on pressed keys on a keyboard, MCP23017 [[5]] writes one byte of data in the register and sends them to the microcontroller on request.

Based on values within the register, the microcontroller determines which key was pressed and acts accordingly. FIGURE 3 shows MCP23017 [[5]] pinout.

FIGURE 3: MCP23017 PINOUT



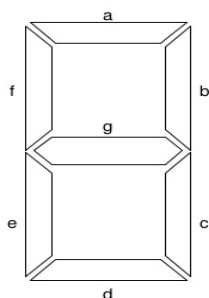
1.4 7-segment display FJ4201AH

A 7-segment display is based on the LED layout – depending on combinations of powered LEDs it represents a digit or character. It consists of seven segments (option with 8 segments have decimal point) which are marked with lowercase letters (a, b, c, d, e, f, g).

The type of display is common cathode, which is required to be controlled by MAX7221 [[3]].

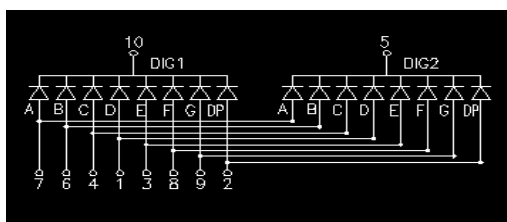
FIGURE 4 shows the display LED layout.

FIGURE 4: 7-SEGMENT DISPLAY LED LAYOUT



In this variant (FJ4201AH [[7]]), there are two displays in one housing, showing two individual digits. It is important to know which pins are used for which segment to connect them to MAX7221 [[3]] properly. FIGURE 5 shows a 7-segment display pinout.

FIGURE 5: 7-SEGMENT DISPLAY PINOUT



1.5x5 Matrix keyboard

Matrix keyboard consists of keys forming electrical circuits when pressed, assuming that columns alternately receive power supply with high frequency. Depending on which key is pressed, one of the rows receives a power supply. In the case of a 4x5 matrix, there are four columns and five rows, and a combination of logic 1 on one of the rows and columns unambiguously determines which circuit is connected and which key was pressed. The keyboard communicates indirectly with the microcontroller through MCP23017 [[5]].

As for the program part, the existing library is reused where most of the necessary functions are already defined and ready to use.

It is important to properly connect the bus from the keyboard (FIGURE 6) to have correct pin definitions for rows and columns within the code. Even though there are 5 rows in the used keyboard, only 4 were used since it is enough to cover all required functions.

FIGURE 6: 4X5 MATRIX KEYBOARD



2. Control

In the previous chapter, two types of serial communication were mentioned to overcome the lack of I/O digital pins, SPI [[4]] and I2C [[6]]. Also, the program logic is important as interconnection between all electronic components listed in the previous chapter as well as IDE (Integrated Development Environment) used to write and upload code to the microcontroller.

This chapter describes the program flow [[1]] required to accomplish all necessary functionalities. Besides integrated libraries, there are two external libraries used to utilize functionalities provided by MAX7221 [[3]] (LedControl library [[8]]) and MCP23017 [[5]] (Keypad_MC17 library [[9]]).

2.1. Arduino IDE (Integrated Development Environment)

Arduino IDE is open-source software developed in Java, and the language used for programming microcontrollers is based on C. It is available for Microsoft Windows, Apple Mac OS, and Linux operating systems. The user interface provides a text editor to write code and upload code to the microcontroller in a simple way. All existing Arduino devices are supported with their respective libraries.

2.2. Control code

This device enables the user to update current results by considering current results, showing the current result on each display, and showing points

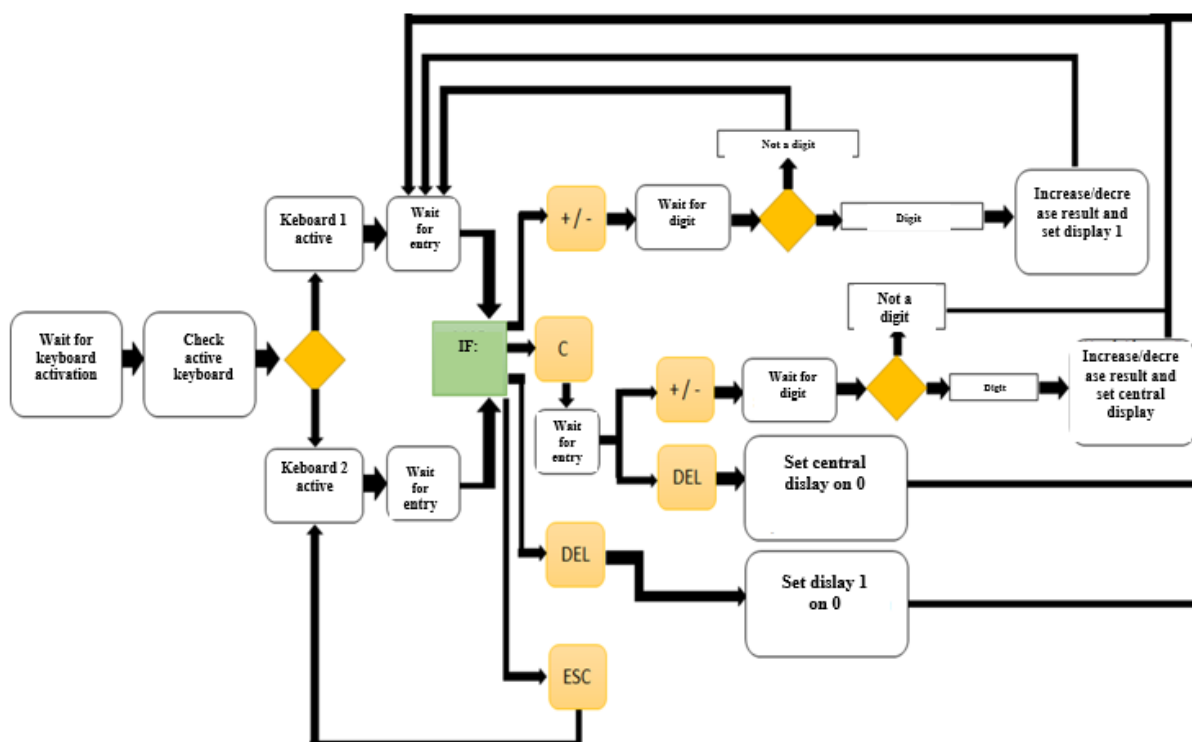
required to win on the central display. The program needs to expect an entry from one of two keyboards, register the value entered, and depending on the entry act accordingly.

Besides digits, the keyboard contains the following functions:

- C – central display, enables control of central display
- V – activates keyboard in case no keyboard is currently active
- DEL – resets values to zero
- ESC – gives control to another keyboard
- -/+ – After these, the digit is expected to increase or decrease the current result by that value

Taking into consideration mentioned functionalities, FIGURE 7 shows the required program flow [[1]].

FIGURE 7: PROGRAM FLOW DIAGRAM



Making electronic devices consists of designing the electrical scheme, drawing an assembly diagram, making the printed circuit board, and soldering electronic components on the circuit board.

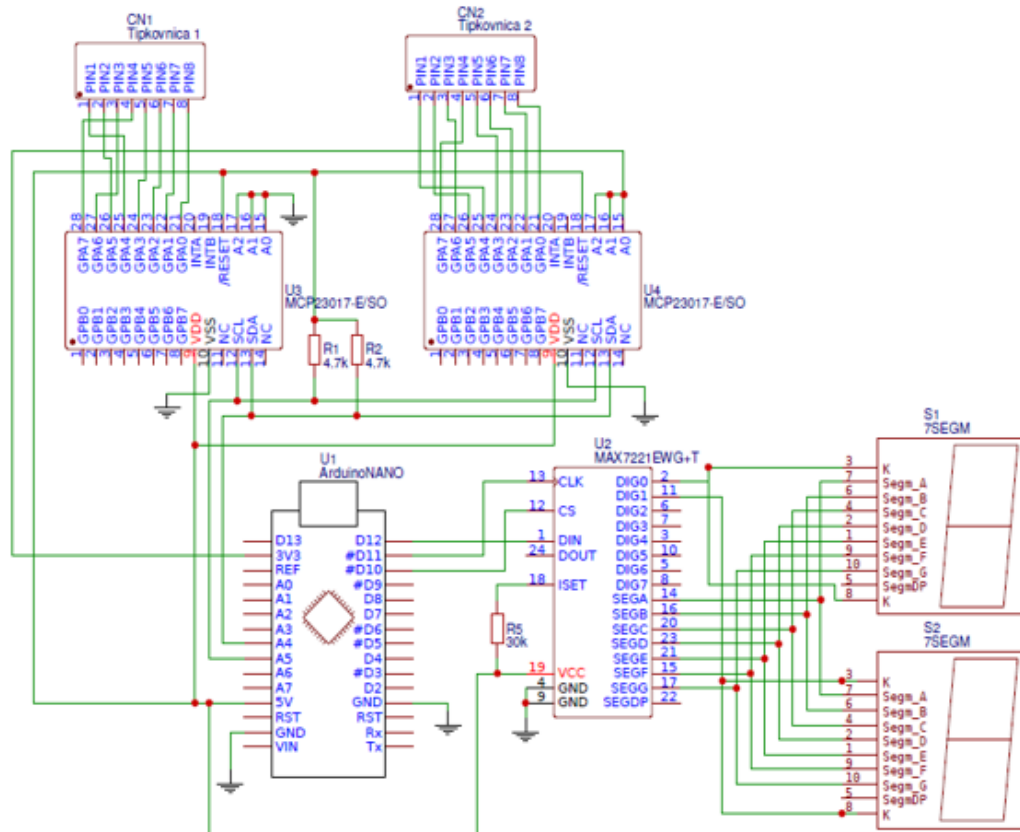
3. Electronic scoreboard design and development

3.1. Electrical scheme

Based on selected components and their technical specifications, the electrical scheme [[1]] is designed (Figure 8). To make it more transparent due to its size,

only two displays are shown here for example, instead of a complete scheme that contains six displays.

Figure 8 Electrical scheme



3.2. Making of the printed circuit board

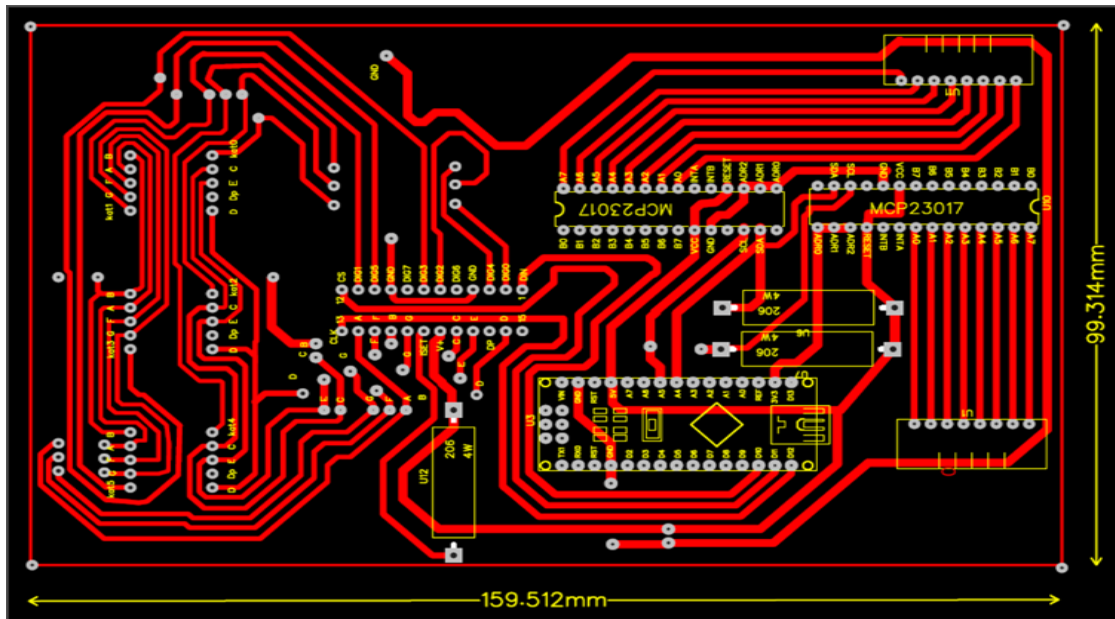
To verify the electrical scheme in the pilot environment, an assembly diagram can be drawn showing all components in their real dimensions with their real pinout. All components are put together with inter-connections between them to match the electrical scheme. If the software in which we draw the assembly diagram, by any chance, does not contain some of the components in their library custom component can be drawn based on their technical specifications. For example, in this diagram components for displays [[7]] and MAX7221 [[3]] were manually drawn. The target size of the board was 160mm x 100mm due to the standard dimension of boards available in stores.

Figure 9 shows a scheme [[1]] with all layers, including names of components, pins, etc., that are usually not printed on the circuit board.

Copper lines representing electrical circuits and connections between components are kept on board. The rest of the copper is removed with hydrochloric acid and hydrogen solution (3:1 ratio).

To keep lines on board, it is necessary to protect them before immersion. Besides the layer of copper, the board also has a layer of photo-sensitive film protecting it. To expose copper, a layer of the film must be removed in those exact spots. The film is removed in sodium hydroxide and that process is called developing the printed circuit board. Lines will be kept in case they were not exposed to ultra-violet light before developing.

FIGURE 9 ASSEMBLY DIAGRAM (WITH COMPONENT NAMES)

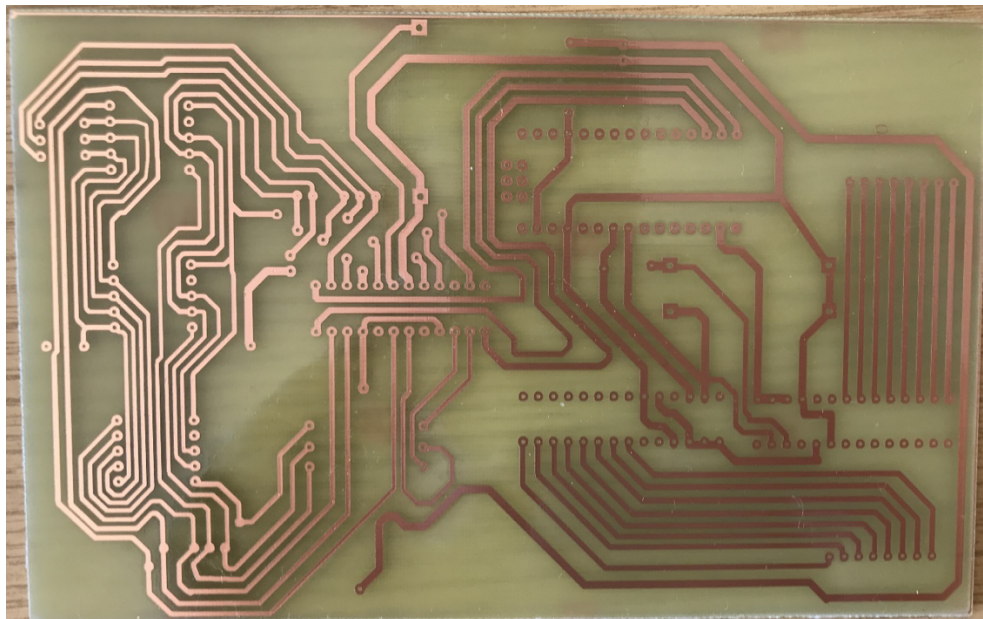


For this, we use an assembly diagram printed on transparent paper as protection from ultra-violet light to keep lines.

The developed board [[1]] (

FIGURE 10) is put in hydrochloric acid and hydrogen solution which slowly removes a layer of unprotected copper. When copper is removed, acetone is used to remove the remains of the photo film. The final steps are drilling holes, putting, and soldering components.

FIGURE 10: PRINTED CIRCUIT BOARD



Result is finished product in prototype housing shown in FIGURE 11.

FIGURE 11: PRODUCT IN PROTOTYPE HOUSING



4. Related work

Product that was used as idea is Play 8 Scoreboard [14]. Main advantage of solution described in this paper is lower price with almost same functionalities at the expense of less elegant appearance. This could be also improved by investing more in proper housing.

5. Conclusions

The result of this paper is a functional electronic device that fulfills the requirements for entering and showing sports results. Detailed planning and research of available electronic components, whose technical specifications fit design, are pre-requisites to any electronic project. Usage of current knowledge, combined with new relevant information, is necessary to make an electronic device.

To accomplish the compact design, Arduino Nano is used as the microcontroller. The main problem was the lack of input and output digital pins for functions of entering and display of the result. Resolution is in using additional driver integrated circuits which are used to control peripheral devices. They interact with the microcontroller using serial communication, which does not require many pins from the microcontroller.

It is necessary to test the design of the electrical scheme and verify its correctness. Program code needs to be loaded into the microcontroller and try all required functions, and if needed, make corrections. After tests prove the device works as in the design idea and no significant improvements can be done, a printed circuit board can be made.

References

- [1] P.Škomrlj, graduate thesis, unpublished
- [2] Arduino Nano, <https://docs.arduino.cc/hardware/nano> (30.5.2022.)
- [3] Maxim Integrated MAX7221, <https://www.maximintegrated.com/en/products/power/display-power-control/MAX7221.html> (30.5.2022.)
- [4] Serial Peripheral Interface, <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all> (30.5.2022.)
- [5] Microchip MCP23017, <https://www.microchip.com/en-us/product/mcp23017> (30.5.2022.)
- [6] I2C Communication, <https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/> (30.5.2022.)
- [7] FJ4201AH 7-segment LED display, https://datasheet.lcsc.com/lcsc/2103261832_Shenzhen-Zhihao-Elec-FJ4201AH_C10695.pdf (30.5.2022.)
- [8] LEDControl library, <https://www.arduino.cc/reference/en/libraries/ledcontrol/> (30.5.2022.)
- [9] Keypad_MC17 library, <https://www.scribd.com/document/400749565/Using-Keypad-MC17-1> (30.5.2022.)
- [10] Play 8 Scoreboard <https://sorensogaard.com/shop/scoreboard-play-8-381p.html?CookieConsentChanged=1> (30.5.2022)