

DIGITALIZATION AS A PROCESS OF ASSISTANCE IN THE TRANSPARENCY OF UNIVERSITY TEACHING

Ragmi Mustafa

University "Kadri Zeka" Gjilan, Kosovo

Kujtim Mustafa

Kosovo

Xhevdet Thaqi

University "Kadri Zeka" Gjilan, Kosovo

Basri Ahmedi

University "Kadri Zeka" Gjilan, Kosovo

Ekrem Halimi

University "Kadri Zeka" Gjilan, Kosovo

ABSTRACT

This is a project with objectives to study and to apply the surface of the hardware and software part of the panel with LED diode lamps. Much more is focused on the programming of LED matrix diodes located on 16×96 electronic panels, respectively 16 rows of 96 columns. Also, there is elaborated the study of the mode of operation, composition, coding and use of software for panel diodes. In the focus of all this is the introduction of diode panels in use at "Kadri Zeka" University exactly at the Faculty of Computer Science to provide digital information to students, professors or anyone passing through the premises of the university what is happening inside the classrooms of the Faculty of Computer Science. This happens in real time in accordance with the schedule of lectures and exercises in the respective classrooms of the FCS. In fact, this project explains in detail the concept of LED diodes and microcontrollers, their features, operation, power supply and work with 0 [VDC] and 5 [VDC] voltages, respectively 0 and 1 logic bits. Meanwhile, the project also explains how coding, software building that locks, and unlocks LED diodes on a 96×16 LED diode panel, thereby producing text on the panel (display, screen) in static, mobile, animation or up-down, left and right movements according to the time allocated to the timers of the microcontrollers. Such action achieves the goal of the digitalization project of the classrooms of the Faculty of Computer Science.

Keywords: LED diodes, light-emitting diode panels, digitization, microcontrollers, memory, microprocessors, programming codes, software.

Introduction

While having lessons in different classes at University “Kadri Zeka”, it happened so often to interrupt from the outside from students and professors who had opened the door in order to see if it is free for studying. Such a behavior even without intention of students and professors, irritated the rest of the people who were studying inside the classroom. Therefore, we as a group of professors of this university have analyzed the problem and came to the solution such this digitalization. With this solution has passed the problem of interruption and now all the students and professors are studying without interruption from outside, first in all the classrooms of Faculty of Computer Science in University “Kadri Zeka”.

Materials and methods

In order to get rid of those problems with interruptions during the studying process are held a long discussions, consulting, meetings between students, professors, administrative staff of Faculty of Computer Science and meetings with working groups of students and professors.

Within the working group of students and professors were also the authors of this paper and also the dean of the faculty. After the all discussions it was concluded that the problem can be solved with panels of diodes of size 96×16 above the entrance doors of each classroom, and the programming of them dynamically also at the real-time according to scheduling timetable for lessons, and the updates of the panels will eliminate the problems of disturbing while opening the doors of classrooms at the critical time while the lessons are held. In the continuation we will explore and describe the programming of LED matrix diodes located on 16×96 electronic panels, respectively 16 rows of 96 columns. Also, there is elaborated the study of the mode of operation, composition, coding and use of software for panel diodes. In the focus of all this is the introduction of diode panels in use at “Kadri Zeka” University to provide digital information to students, professors or anyone passing through the premises of the university.

Characteristics of LED Matrix Panel

The panel that is the object of study and review is not the most qualitative in the market, but performs the function for which it is explored, and any other model works according to the same principles.

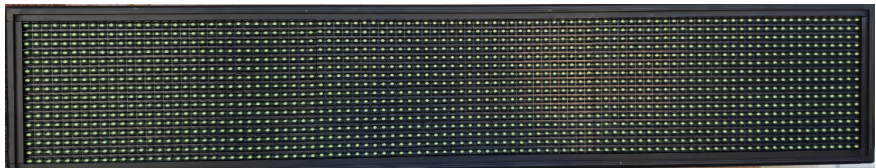


Figure 1. LED display 96 × 16 with unlit LED diodes (switch off)

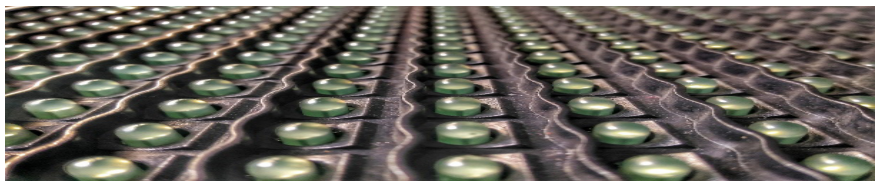


Figure 2. LED panel diodes 96 × 16

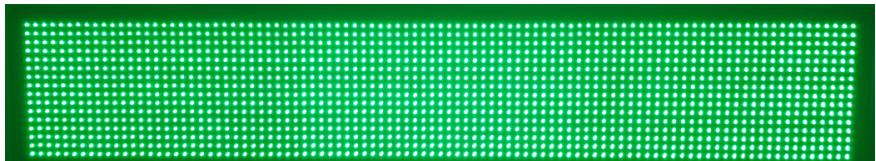


Figure 3. LED Matrix 96 × 16 Panel with LED Diodes (switched on)



Figure 4. Panel Microcontroller and electronic Card Model HD-E65 for LED Diodes of Panel 96 × 16

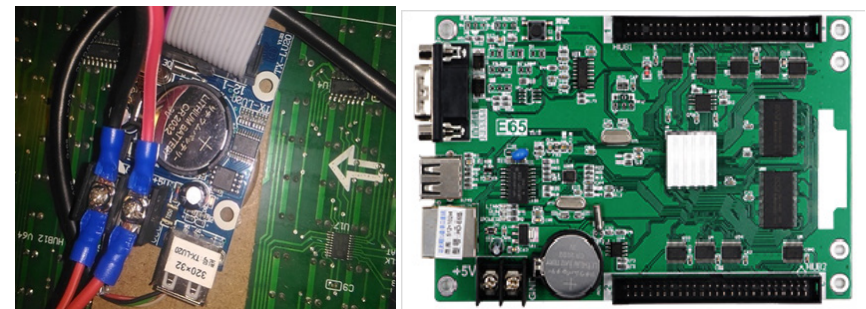


Figure 5. Led Panel 96 × 16 diodes and information about what is currently happening in the room B24 FSHK's



Figure 6. Two panels connected physically and present the inscription of the University

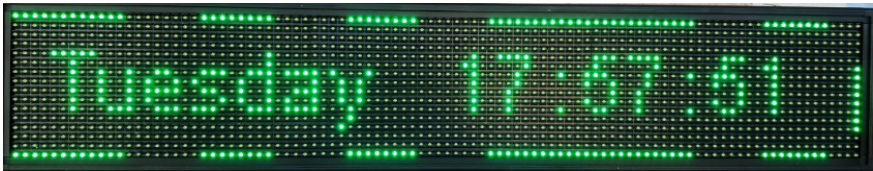


Figure 7. Moving text, digital day and time

Table 1. Characteristics of LED panel 96 × 16 diodes (from Pond, R. J. and Rankinen J. L, 2008)

Model	HD – E65
Chip type	Supports only green color
Distance between LEDs	10 [mm]
Supply voltage and electricity	5 [V] 40 [A]
Luminance	≥ 7500 cd/m ² (for outdoor)
Light module format	32 × 16 LED
LED number for module	512 LED
Panel Dimensions	110 × 10 × 22 [cm]
The amount of power consumption	20 [W] – 40 [W]
Number of modules	3
Angle of view	120° horizontal, 60° vertical
Working temperature	-20 [°C] ~ + 60 [°C]
Supply to the city’s electrical network	Alternative electricity 100 ~ 240 [VAC]
LED diodes for panel	96 × 16 LED
Total number of LEDs on the panel	1536 LED

ML 96 x 16 R (model: HD E-65) and Led Diodes

The digital display is able to emit static text, moving, displacement, animation, cyclic, scroll, slug, blinker, that text moves from left to right, from right to left, bottom-up, top-down or similar movements. Show the correct time, date, temperature and even one or more rows depending on the number of diodes according to the lines of the display.

Relatively with high resolution (from Pond, R. J. and Rankinen, J. L., 2008). 24 types of animations can be selected for text you can adjust the speed of the text movement. The possibility of the timer display option that automatically activates or deactivates the device programs within certain time intervals. In the computer is formed a text, the module is programmed and it can be stored and then transferred to USB connected to the appropriate port of the diode panel and transferred as text on the display.

The text on the digital display is easily programmed through software that works in the Windows operating system. The system recognizes and can perform with different types of fonts that are used on the Windows operating system (from Pond, R. J. and Rankinen, J. L., 2008). The LED panel programming is performed through the respective software installed on the personal computer and transferring the program through USB port with USB memory, via RF or Wi-Fi technologies (from Pond, R. J. and Rankinen, J. L., 2008)!

Digit: **ML96 × 16R**; HD2016 LED Display Controller; Card Model: HD E-65



Figure 8. Incription display using LED green diodes

Illuminating Diode (LED)

LED is the abbreviation of English words “*light-emitting diode*”, that is, a light-emitting diode. The illuminating diode is built in the same way as the usual semiconductor diode. It will work if it is polarized on the right side. The intensity of the light is regulated by the current which flows through the diode also exponentially.

The diode will not emit light when it is polarized in the opposite direction (Milman-Halicias, 1972; SK Bhattachary, 2000).

The color of the light emitted depends on the type of semiconductor and the added impurities.

The diode produced by *gallium phosphate* (GaP) emits red light while that of *gallium arsenide phosphate* can emit light green or yellow (Milman-Halicias, 1972; SK Bhattachary, 2000).

The graphic symbol and characteristics of the light-emitting diode are shown in Figure 9 as follows:

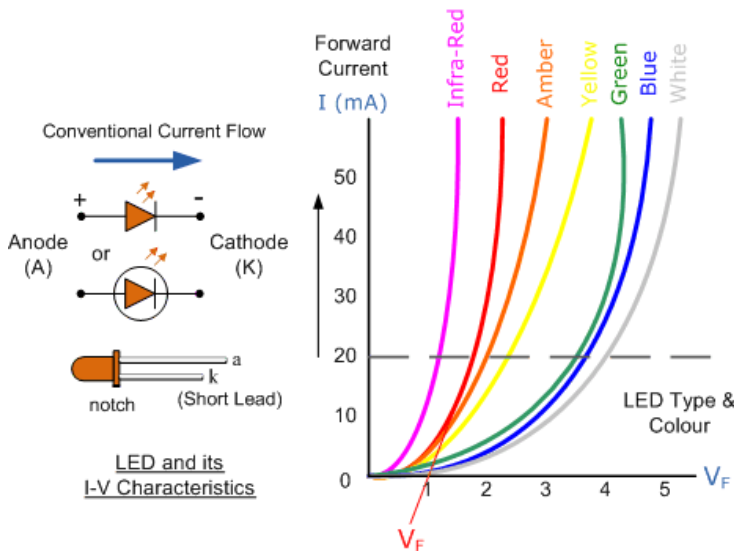


Figure 9. Characteristics of LED diodes (D. Hearn, M. Pauline Baker, 2004)

The advantages of light-emitting diodes are:

- small size
- safety at work
- long term service
- work with low tensions and currents
- small losses of electricity
- great work speed.

Usability

LED Light-emitting diode are used for signaling, through which the open or closed status of an electronic circuit is indicated. Especially they are implemented in bright light indicators (Philip Burgess, 2016; D. Hearn, M. Pauline Baker, 2004). LED diodes look like in figure10:

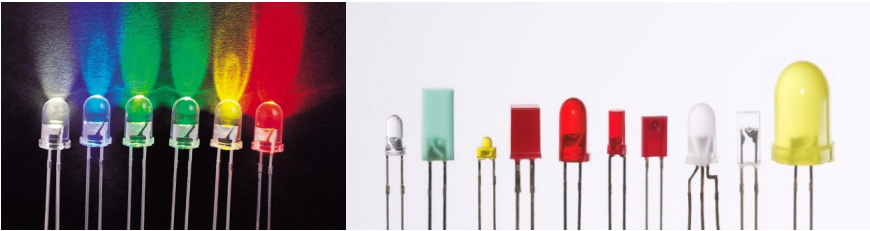


Figure 10. Types of LED light-emitting diodes by color enlightening

Blue light emits LED diodes in the electromagnetic waveband from 450 to 500 [nm] – (nanometers).

Green light emits LED diodes in the electromagnetic waveband from 500 to 600 [nm] – (nanometers).

Red light emits Led diode in the electromagnetic waveband from 600 to 690 [nm] – (nanometers).

At the moment of power supply of LED diodes with semiconductor material content of gallium arsenide phosphate and its positive polarization leads to electron bulging causing the production of light waves in the range of 550 to 640 [nm]. Breaking light on the conic lenses of the observer’s eye (human) creates a sense of perception of the green color according to the spectrum of the colors of the light waves (Philip Burgess, 2016; D. Hearn, M. Pauline Baker, 2004). In other cases, semiconductor material and semiconductor diodes will produce light beams at other intervals of the wavelength and creates a sense of perception of the respective color, always according to electromagnetic spectrum ranges of electromagnetic wavelengths as shown in Figure 11.

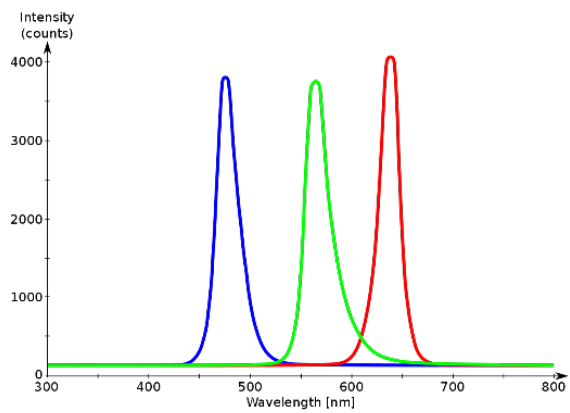


Figure 11. RGB base color spectrum acquired by the length range of the light beam (electromagnetic wave) (D. Hearn, M. Pauline Baker, 2004)

Lightining Diode (LED)

LEDs are special diodes that emit light when connected to an electrical circuit. They are often used as a pilot lamp in electronic applications to show when the circuit is running or not (Milman-Halicias, 1972). In a right polarized diode, free electrons pass the union of p-n contact and become conductive. If these electrons pass from a higher energy level to a lower power level, they generate energy.

In ordinary diodes like Silicium (Si) or Ge (Germanium), most of this energy goes in the form of warmth. In other materials such as gallium (Ga), Arsenic (As) and Phosphorus (P) or Gallium-Phosphorus (GaP), the number of photons of light is sufficient to create a visible light source. In light emitting diodes, this energy radiates as light.

The most important part of an LED is a semiconductor chip placed in the center of the lamp. The chip has two layers: the **p** layer has mainly positive electrical loads and the layer **n** with negative electrical loads. When a sufficient voltage is applied to the diode, the electrons can flow easily in one direction to the **p** and **n** layer contacts.

So it starts to flow the current, because layer electrons **n** has enough energy to move to the **p** layer (Milman-Halicias, 1972).

When an electron moves and falls on a positive load, both loads are recombined, so when an electron hits a hole, it falls to a lower level of energy and emits energy in the form of a photon.

Whenever an electron is recombined with a positive load, the potential electric energy is transformed into electromagnetic energy (Milman-Halicias, 1972).

Graphic symbol of enlightening diodes is given in Figure 12.

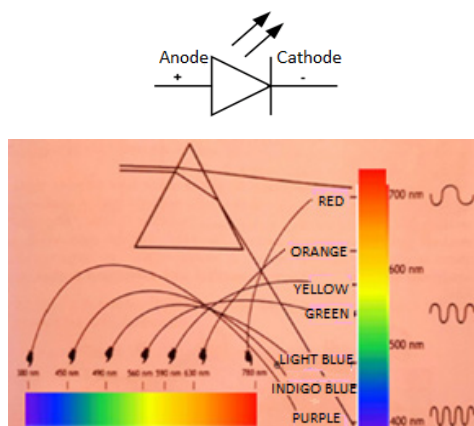


Figure 12. Graphic symbol of LED light-emitting diodes (Robert J. Pond, Jeffrey L. Rankinen, 2008)

Figure 13 shows how a LED diode connects to an electrical circuit.

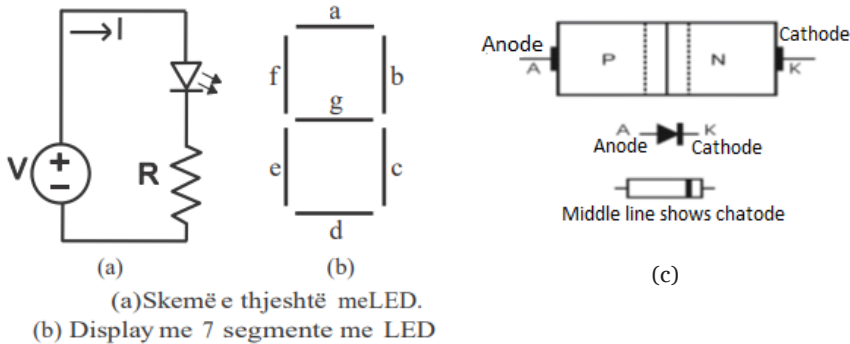


Figure 13. Electrical scheme of a LED diode and the p-n contact (Milman-Halicias, 1972)

- a) the electric polarization circuit of straight LED illuminating diodes
- b) 7 segment LED display.
- c) p-n contact with cathode and anode of the semiconductor diode.

A seven segment display contains seven LED diodes. Each LED diode is called a segment. In figure (a) there is shown a segment of a 7-segment display as well as the series-related resistors used to limit the current value of the diode and its protection. Given the supply voltage of LED diodes from 0 to 5 [VDC] and the permissible current from 0 to 20 [mA], the LED diodes are connected to serial-resistor whose resistance is calculated according to the Ohm Law:

$$I = \frac{U}{R} \text{ respectively: } R = \frac{U}{I}$$

Example: For supply of 5 [VDC] voltage to LED diodes and a 10.63 [mA] current flow permissible within the range of {0–20 [mA]}, according to the Ohm Law, calculate the required resistance of the resistor needed to connect to the diode in series:

$$R = \frac{U}{I} = \frac{5 \text{ [VDC]}}{10,63 \text{ [mA]}} = 470 \text{ } [\Omega]$$

Some of the LED uses in everyday life are:

- architectural lighting
- indicators (condition indicators) in many electronic devices
- traffic signals and road signs
- handheld flashlight
- remote controls thanks to the infrared LED
- optical fiber for traffic
- for Christmas tree lights
- LED anti-acne phototherapy has been effective in the disappearance of acne for a 3-month period,
- lighting panels and other textual descriptions.

Disadvantages of LED usage: LEDs are more expensive currently than other lighting technologies. LED performance generally depends on the ambient temperature in which it works. By operating the LED at high ambient temperatures, this may result in overheating of the diode and lead to damage (drilling effect) and to malfunction.

Microcontrollers

The microcontroller contains microprocessor, memory and a large number of peripheral devices such as timers (timer relays), serial ports, input / output pin terminals, numerators, analog inputs and so on (Warwick Smith, 2016). All of these are within a silicon circuit in the form of a built-in system.

Examples of built-in systems are: calculators, computers, and smart phones. The 8x8 LED diode panel and the programming code for their connection are defined in Appendix A.

The microcontroller system architecture has changed from time to time, but what is left behind is the programming language C (Warwick Smith, 2016).

The programming language consists of alphanumeric characters, syntax of commands, functions which constitute a program code comprehensible to man, respectively the programmer.

On the other hand, the execution of the programming code on the computer is performed when the program code is compiled in the machine language. This implies that each alphanumeric sign of the programming code is converted to the binary code (0 and 1 logic) according to the international agreement the ASCII table or the extended EBDIC table (Warwick Smith, 2016). This, in effect, represents the string of square time voltages of the lowest level 0 [VDC] and highest 5 [VDC] respectively.

Meanwhile, 5 [VDC] voltage supplies LED light-emitting diodes, which causes LED diode brightness, while 0 [VDC] voltage fails to arouse the illumination of LED diodes (Warwick Smith, 2016).

All input data are placed in the microcontroller memory. From the microcontroller memory, the input data is processed in the micro-processor for processing and the results obtained according to the respective software or programming code are stored in the memory and transmitted to the respective ports of peripheral devices and in this case the 96×16 diode panel where diodes are placed within an electronic plate (Warwick Smith, 2016).

If the microcontroller memory inserts the letter B and the voltmeter (voltage measuring instrument) it will be possible to measure the voltages of each bit in turn we will show the voltage measurements as in the following figure 14:

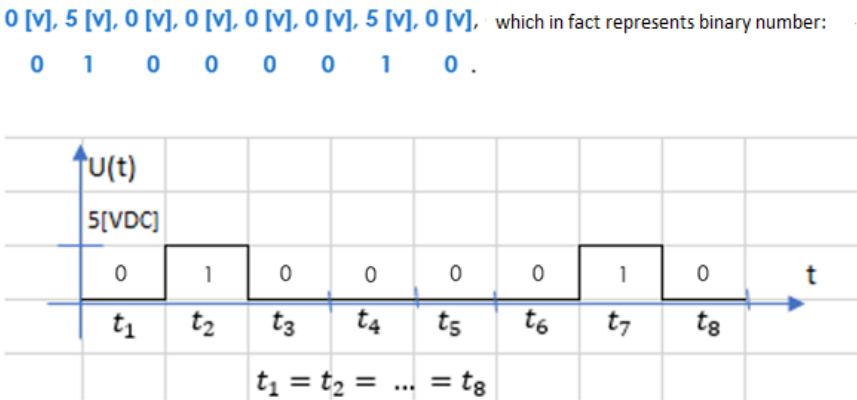


Figure 14. Chart tensions 0 [VDC] 5 [VDC], presentation by binary numbers 1 and 0 logic

Presentation of binary digit 1 with voltage of 5 [VDC] or any other value depends on the technical characteristics and electronic components of the microcontroller system. 8-bit array: belongs to the decade number because: $(01000010)_2 = 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 + 0 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 2 + 64 = 66 = (66)_{10}$ viewed at the ASCII – code table, this belongs to letter B.

Why to work with binary numbers?

Programming of built-in electronics such as microcontrollers consists of communication with hardware devices that are related to the microcontroller, as well as the devices located in the microcontroller itself (memory, processor, timer and other stuff) (Warwick Smith, 2016).

Let us consider the microcontroller to the output are connected four light-beam diodes as shown in Figure 15.

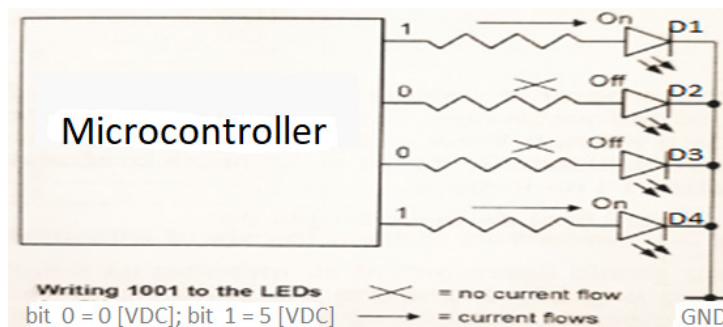


Figure 15. Logic circuit, how the diodes are turned on by the microcontroller (Warwick Smith, 2016)

To connect the LEDs D1 and D4 to the LED diode, these LEDs will be able to transfer binary digits 1 0 0 1. As binary numbers are archived in the form of voltage levels. Indeed, we connect the 5 [VDC] voltage to anode of the diode D1, 0 [VDC] to anode of the diode D2, 0 [VDC] to anode of the diode D3 and 5 [VDC] to anode of the D4 diode. In this case, the LEDs D1 and D4 will illuminate while the LEDs D2 and D3 will not illuminate.

Memories

In microcontrollers and built-in systems there are two types of memories, called ROM and RAM (Warwick Smith, 2016).

ROM memory – This type of memory is used to archive the program that is written and compiled. This is a durable memory and that even when power supply disconnects does not lose its content (Warwick Smith, 2016).

RAM memory – Memory RAM is used to store variables and data from the microcontroller work program. Data and variables can also be changed during the work of the program.

This type of memory is not stable which means that with the termination of the power supply its content are lost or deleted at all. SDRAM SRAM and EPROM memory are often used for memory expansion (Warwick Smith, 2016).

Archiving of data in memory

The data in the memory is placed in the form of a byte sequence. Each byte possesses its unique address and can be addressed as in Table 2. Data in the memory can be recorded and read.

Bit positions in a memory address from 0 to 7 of a byte!

Table 2. Presentation of data storage (bytes) in memory

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Address 0	1	0	1	0	1	0	1	0
Address 1	0	0	1	1	0	1	0	1
Address 2	1	1	1	1	1	0	1	1
Address 3	1	0	0	0	0	1	0	0
Address
Address n

1 Byte of data

Each cell can store bit of value 0 or 1

Appearance of the circuit (chip) memory

In addition to ROMs of RAM, many types of microcontrollers also have an additional data storage memory called SRAM (Warwick Smith, 2016). Figure 16 shows an SRAM with a capacity of 2 kB (kiloBytes). Each memory byte is called a cell and may contain logical levels respectively logic bits 0 and 1.

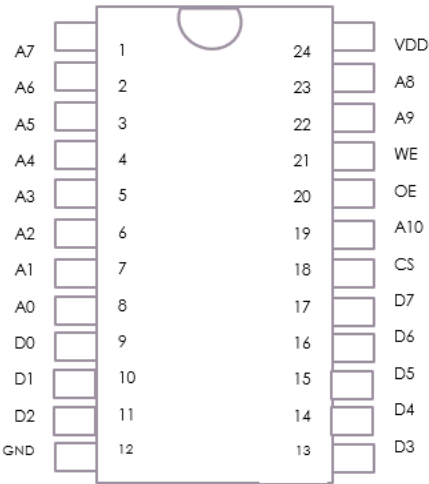


Figure 16. Memory circuit, 8-bit S-RAM

- VDD and GND:

A0 to A10:

D0 to D7:

CS:

OE:

WE:
- power supply to chip

Address bus – data byte address placed here

Data bus – 8 bit data byte read and written here

Chip select

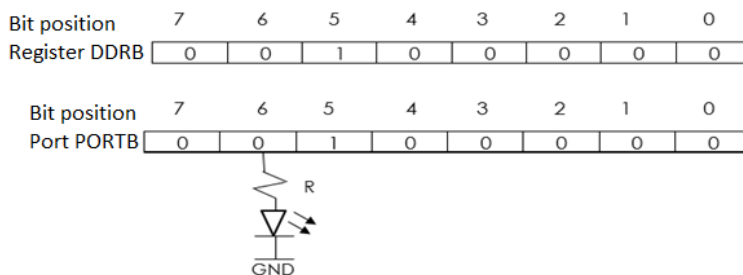
Output enable

Write enable
- }

Control bus

Microprocessor access to memory and peripheral devices

The microprocessor contains Address Bus, the Data Bus, and the Control Bus. The memory connection with the processor is performed through the respective buses of the two main microcontroller devices (Warwick Smith, 2016). When the microprocessor is supplied with power its counter is set to 0. If the permanent memory (RAM) that contains the program is connected through the microprocessor circuit, the microprocessor will receive the first instruction (instruction is a binary number) of the programming code and executes it. The microprocessor program counter increases for one and receives the next programming instruction at address 1 of the memory and executes it. The microprocessor counter continues to increase for one by enabling execution of the codebook instructions to the latest programming instruction as in Figure 17.



```
int main(void)
{
    DDRB = 0x20;    // set bit 5 of DDR register which makes PB5 an output
    while(1)
    {
        PORTB = 0x20;    // switch LED on
        Delay();
        PORTB = 0x0;     // switch LED off
        Delay();
    }
}

void Delay(void)
{
    volatile unsigned long count = 100000;
    while (count--);
}
```

Figure 17. Connecting microprocessor with memory and peripheral devices

Pins and ports of microcontroller

Microcontrollers possess a relatively large number of pins known as input / output pins (I / O – Input / Output) of general implementation.

Pins are defined as outputs serving to connect the microcontroller output connections as in the present case for switching and disconnection of LED lighting diodes as well as input pins for reading logic levels 0 or 1 located in the microcontroller pin (Warwick Smith, 2016). 8-bit microcontrollers possess 8-bits ports that represent pin I / O groups labeled as ports A (PORTA), B (PORTB), C (PORTC), and so on.

Switching on and switching off the LED light-emitting diodes in the electronic board

When the built-in microcontroller system is connected, all of its pins are initially defined as inputs. To switch on or off LED light-emitting diodes placed on an electronic board, we need to set or describe the values in the respective data entry port (DDR – Data Migration Data Logger). Therefore, for access to Port A, the DDRA register is used, Port B is used for DDRB and so on, where LED diodes are connected (Warwick Smith, 2016). Figure 18 shows the memory registers that are required for the control of LED lighting diodes on the Arduino Uno electronic board, which also applies to other microcontrollers wherever they are placed on the electronic board.

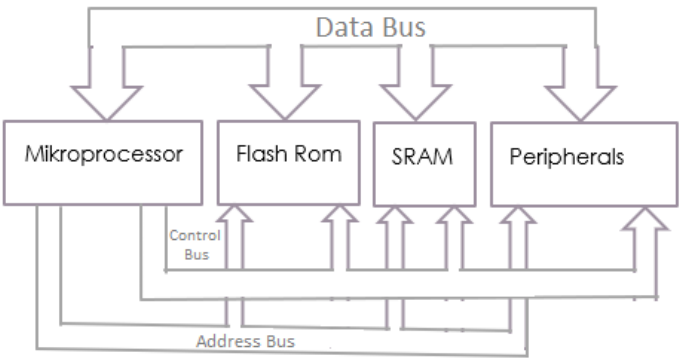


Figure 18. Memory registers in Arduino Uno (Warwick Smith, 2016)

In the pin with index number 5 (PB5) of the plate (Arduino Uno) is connected the LED Led Illumination. If the bit with position 5 of DDRB is described or transmitted logical energy level or be 1 then port B ordinal number 5 (PB5) is set as PIN DOWN.

The hexadecimal number to be described in this case in the register is: (20) 16 or 0×20 , in which case the LED will be illuminated. Therefore, if logic or beam 1 is described in the register, this is the fact that the LED diode is supplied with a voltage of 5 [VDC] which will switch off its illumination.

The binary number in the register (00100000)₂ is converted to the hexadecimal as follows:

$$\begin{aligned}(00100000)_2 &= (0010\ 0000)_2 \\ (0010)_2 &= 0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 = 2 \\ (0000)_2 &= 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 = 0\end{aligned}$$

Therefore, the hexadecimal number is: (20)₁₆! This happens with the best intention that data in programming language C are written in hexadecimal format rather than binary format! The programming code example C in the programming language C for diode input is defined as follows:

The result of this programming code in the programming language C is that in the electronic board where LED diode is connected, the same to switch it on or off the LED diode to pulsate by switching on and off at a high frequency!

The file: [io.h](#) located at the beginning of the program code enables the registry to be accessible for access. Setting logic 1 to the set bit position in the DDRB registry creates the corresponding pin declaration option as OUTPUT PIN on the PORTB port.

In the loop [while \(1\)](#) the PORTB register is set to enable the LED diode to light up, provided that the corresponding bit in the log has the logical value 1, the same bit is again set but in logical value 0 in order to switch off the LED diode light. Calling the [Delay \(\)](#) function between the LED diode switch on ensures that the LED diode remains in the turned on state and turned off enough at a high frequency so that it is visible to the human eye, creating the impression of the constant illumination.

With the [while \(count\)](#) command is reached that the program cycle is repeated whenever the counter [count = 100000](#) decreases for one to the 0 value when also the command [while \(count--\)](#) gets the false value. This actually determines the timing of the LED dialing switch disconnection. At the end of this time according to [volatile unsigned long count counter \(100000\)](#) the program comes out of the cycle and completes its work.

Array

Programming language C are successive memory locations that archive specific type of data. For example, the programming string C, consisting of 5 integers (Warwick Smith, 2016), is defined as follows:

```
int arr_num[5];    or    unsigned char seq[ ];
```

Through the definition of the arrays, it is easy to switch on and off LED light diodes placed on an electronic board (LED DIODE PANEL)! The following programming code initiates the array through a set of numbers that are described on the LED for their switch on and off. Such array is known as a reference table.

Connecting four LED diodes to the pin terminals of PC0, PC1, PC2 and PC3 microcontrollers is explained in the same way as before (Warwick Smith, 2016)! The values in the full-range string are the values that are described (placed) on the microcontroller output port for LED light-disconnection and disconnection.

The array of the given numbers is 1, 2, 4, 8, 6, 9, 15, 5, 10, 5, 10. One-dimensional array numbers should be converted to hexadecimal format so that they are read in the file. The programming code for this case of four LED diodes is defined as follows:

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
int main(void)
{
    unsigned char seq[ ] = {0x01, 0x02, 0x04, 0x08, 0x06,
                           0x09, 0x0F, 0x05, 0x0A, 0x05, 0x0A};

    int ind;
    DDRC = 0x0F; // lower 4 bits of port C are outputs
    while(1) {
        for (ind = 0; ind < sizeof(seq); ind++) {
            PORTC = seq[ind];
            _delay_ms(1000);
        }
    }
}
```

The result of this program code looks like in Figure 19.

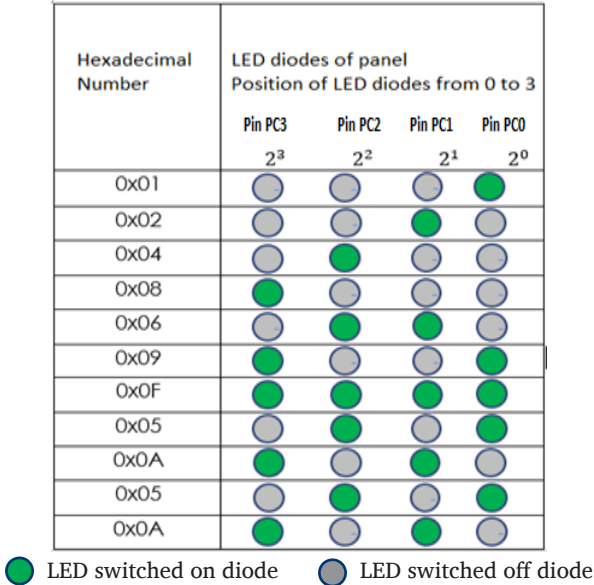


Figure 19. LED diode connection according to matrix programming code (Warwick Smith, 2016)

After defining the array and defining the variable that should be used as an index on the array elements, the C port will be tuned for connecting the four diodes to the corresponding pin PC0, PC1, PC2 and PC3 before it is put into operation `while(1)`.

If you want to present the letter F in a display panel with 12 rows and 4 LED diode columns as shown in Figure 20, write the program code in C as follows:

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
int main(void)
{
    unsigned char seq[ ] = {0x0F, 0x08, 0x08, 0x08, 0x08, 0x08,
                           0x0E, 0x08, 0x08, 0x08, 0x08, 0x08};

    int ind;
    DDRC = 0x0F;           // lower 4 bits of port C are outputs
    while(1) {
        for (ind = 0; ind < sizeof(seq); ind++) {
            PORTC = seq[ind];
            _delay_ms(1000);
        }
    }
}
```

















































Hexadecimal Number	LED diodes of panel Position of LED diodes from 0 to 3			
	Pin PC3	Pin PC2	Pin PC1	Pin PC0
	2 ³	2 ²	2 ¹	2 ⁰
0x0F				
0x08				
0x08				
0x08				
0x08				
0x08				
0x0E				
0x08				
0x08				
0x08				
0x08				
0x08				

Figure 20. The LED lighting diodes forming the letter F

The HD-E65 LED panel software, consisting of 96 LED diode arrays and 16 rows also LED diodes (96×16) known as: HD2016

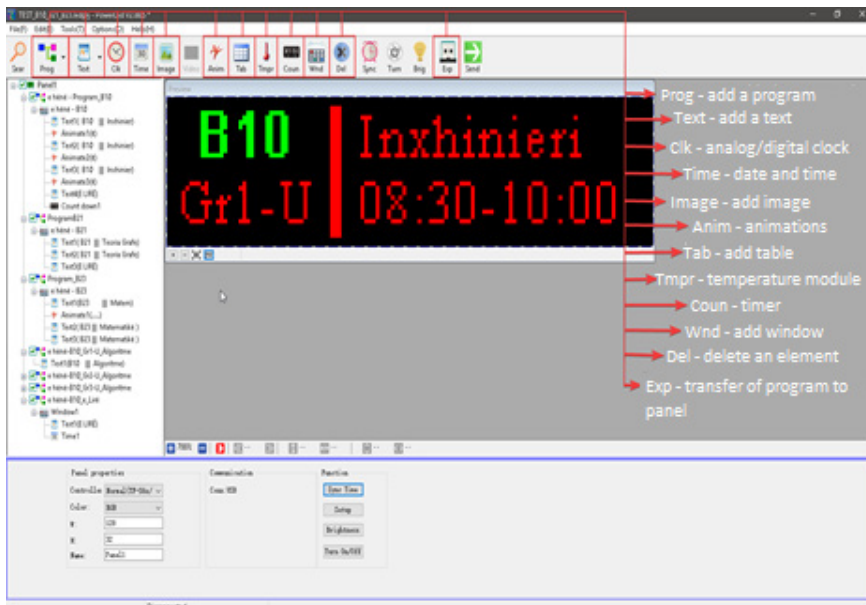


Figure 21. The look of the PowerLed 96 × 16 Test software

PowerLed is a software used for programming RGB matrix panels, it contains a collection of tools that can be used to program different panels (from manual HD2016_Operation_manual_V1.0).

PowerLed consists of:

- Menu bar
- Toolbar
- Window class
- Virtual Panel Window
- Feature Window

For the programming of the LED panel, in special cases the Software is used which is dedicated to 160 × 32 panels (160 columns with diodes and 32 rows) model:

- HD2016
- Module HD – E65
- Screen Size: 160 × 32
- Color: Single color
- Communication: Com port, U disc, Ethernet
- Memory: 2M

LED module panel diode 96 × 16 by the programming software is treated as a matrix of elements set in 16 rows of 96 columns, whose elements are

the addresses of LED diodes placed in rows and panel columns. Meanwhile, the positioning of the text appearing on the LED panel display is also treated as a plane coordinate system (x, y) oriented towards the x axis and y as in Figure 22.

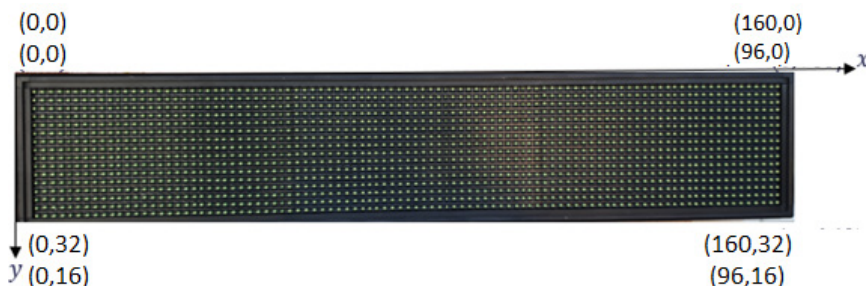


Figure 22. Coordinates of the LED panel 96×16 in the direction of the axis (x, y) respectively LED panel 160×32

Therefore, for panel programming, we need to consider which software is available for use.

When programming a LED panel with a 96×16 LED diode via a software module for 160×32 LED diode panels, it is necessary to determine the start of the software co-ordinate system so that we can set the text at the top of the panel from left to right (from Pond, R. J. and Rankinen, J. L., 2008).

Otherwise, we will not be able to see the desired text because the placement of the text will remain defined outside the positive (real) coordinates of the coordinate system (from Pond, R. J. and Rankinen, J. L., 2008).

For this case, if the software is designed for 160 columns of 32 rows, then in the extension of the x-axis, move right to: $160 - 96 = 64$ positions, which will be considered the point (0.0) for the 96-column panel and 16 rows! This is done in the window part as in Figure 23 as follows. Of course this is achieved by selecting the menus as follows: screen1 / program1 / text1 and window displacement for 64 points in the positive direction of the x axis, while in the direction of y axis only half of the field is transmitted because: $32 : 2 = 16$!

Figure 23 shows coordinating the LED array 96×16 coordinates via the 160×32 diode software

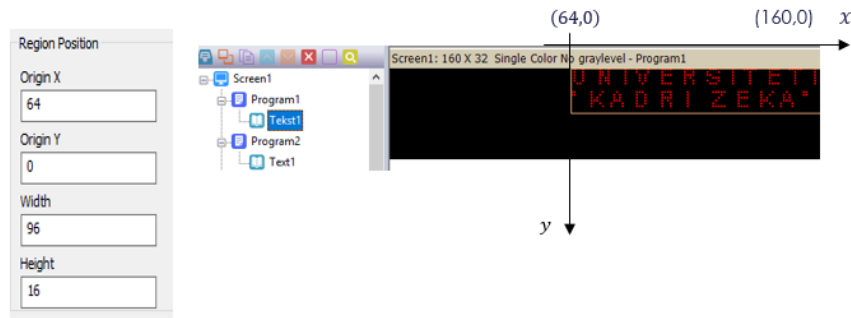


Figure 23. Allocation coordinates of LED panel diode 96×16 160×32 through software diodes

The rest of the use of the software whether it is designed for 96×16 and that for the 160×32 LED diodes is indistinguishable in essence. Through programming software for LED panels, it is possible to program up to 1000 programs in a screen so that in each program they can be divided up into 20 different areas (from Pond, R. J. and Rankinen J. L, 2008).

However, the programming software for LED panels can also be granted to whom will their module be as follows:

The File / New Screen menu is selected and in the open window the password or password 168 is set (according to the manufacturer's recommendations, implied) and in the next window are set the parameters that correspond to the panel LED available to us after setting parameters and confirming they do not need to be maneuvered in coordinating the coordinates. After this tuning, the plane coordinates (x, y) look like as in Figure 24.

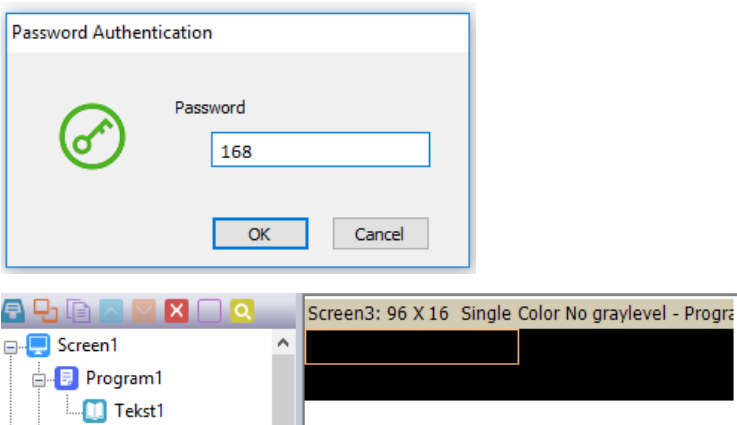


Figure 24. Synchronization windows – adjusting the programming software with the LED panel model that we have available

Origin X	<input type="text" value="0"/>
Origin Y	<input type="text" value="0"/>
Width	<input type="text" value="48"/>
Height	<input type="text" value="8"/>

If we look at the display window where the coordinate values of the axis x and y are set (Screen1 / program1 / text1) in this case, the width is 48 while the height 8 means that the software enables four fields to move the text. The mouse can easily capture and release the edges of the 48X8 frame until the 96X16 screen size is reached to write in a single full 96X16 diode screen area as shown in Figure 25.

Figure 25. The window obtained with coordinates of origin according to axis boshtit (the width of the display field) and (the height of the display field)

Screen respectively LED display with 96×16 enlightening diodes and distribution of pixels according to the axis x and y appears as in Figure 26.



Figure 26. Screen respectively LED display with 96×16 enlightening diodes as well as pixel distribution by axis x and y

Menu bar

The software for programming LED Panels contains menus by which we work and manipulate with PowerLed. With this software you can choose text with a great opportunity of using different fonts, setting the analogue but also the digital clock, adjusting the time of text display on the screen, full programming according to a time period for one or more days, weeks, months and so for several years.

Meanwhile, a temperature thermometer (**Type: AM2301**) can be set for temperature measurement. The LED panel, according to the software, can also be used for numbering by increasing or decreasing number, as a device can be used for the numbering of vacancies in a smart auto parking.

Next, some programming methods will be displayed according to the menus defined by the programming software.



Figure 27. Menu bar

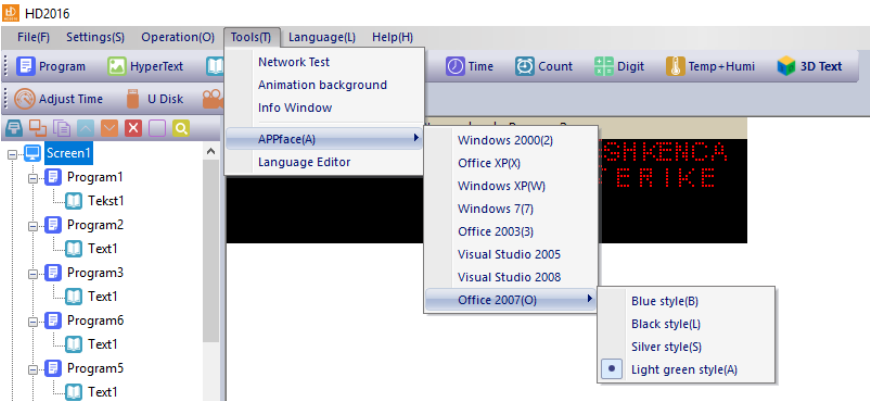


Figure 28. Tools Menu(T)

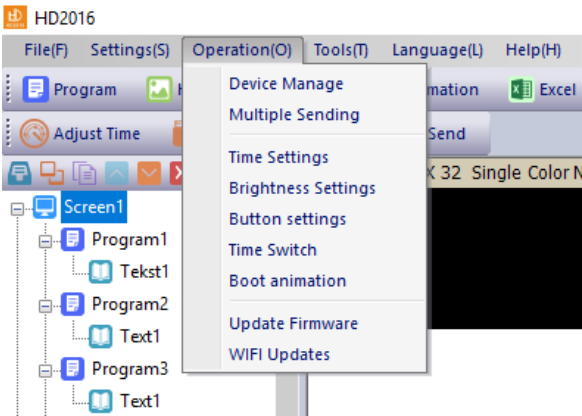


Figure 29. Operation Menu (O)

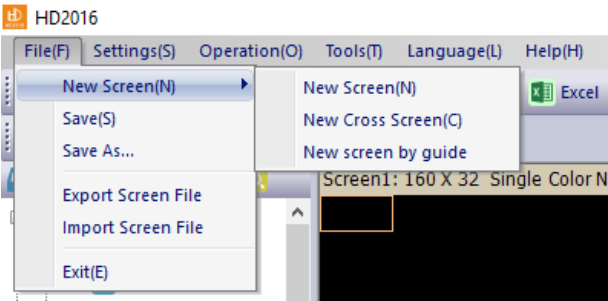


Figure 30. File Menu (F)

Toolbar



Figure 31. Toolbar

It consists of a collection of tools used for building various programs that contain text, pictures, animations. Each of these tools performs a specific function within PowerLED and are useful in building different programs, the use of these tools is limited only by creativity. They can be used as often as we need.

Program tool

It is used for the introduction of a new program within a screen (screen1)

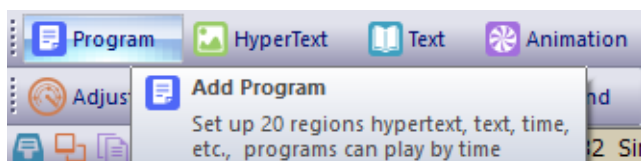


Figure 32. Program tool

Text tool – It is used for inserting text in certain areas of the screen

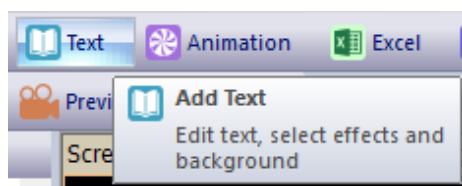


Figure 33. Text tool

Time tool – It is used to set the analog or digital clock in a certain area of the screen

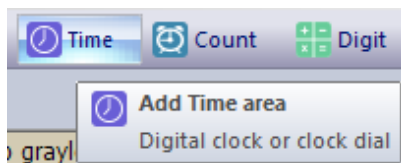


Figure 34. Time tool

Temp + Humi tool – It is used for temperature setting if the thermometer is connected to the LED panel:

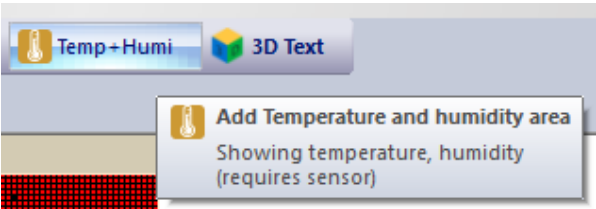


Figure 35. Temp + Humi tool

Preview tool – It is used for presenting the text of the program before it is stored in memory or transmitted via RF equipment or Wi-Fi technology in the LED panel 96 × 16.

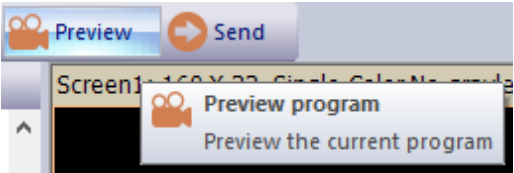


Figure 36. Preview tool



Figure 37. An example of the Preview window

U disk tool

It is used for exporting screen (Screen1) prepared with certain programs and texts in memory (USB) provided that the Memory Flush is placed in the relevant computer port. After export to Memory Card (USB), the same goes out of the computer port and is placed on the LED panel port. In this case the program is read and the OK message is displayed on the LED panel. At this moment, the flash memory is removed from the LED panel and the device works according to the program now installed on the 96 × 16 LED panel!

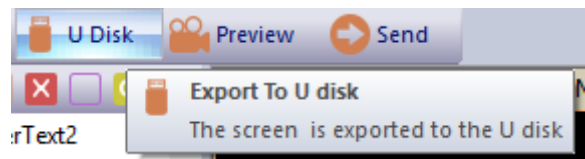


Figure 38. U disc tool

Property window

This window allows you to use all of the tools found in PowerLed. It is determined that a text that appears on the panel for a certain interval then comes next with the animation that I choose and many other manipulations such as the determination of the amount of diode illumination.

With this tool, the text is exported to the peripheral device and in our case on the flash memory (U Disk). Communication with the software is performed as in Figure 39.

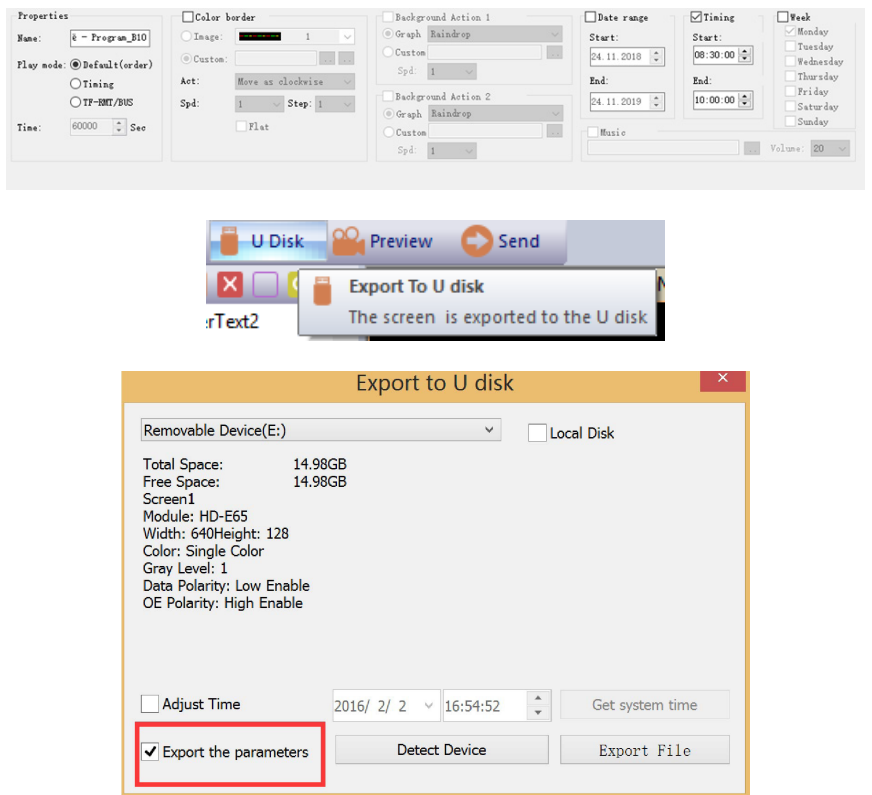


Figure 39. Exporting (transferring) the program in the flash memory (USB)

Results and discussion

During the development of this project, it was hard working with students in order to achieve the good results and to involve the students in the real projects and to teach more practically students with real projects. In this contest are applied the students knowledge from the lessons learned on electronics, programming and mathematics. Result of this project produced the success of eliminating the disturbing cases during the study process in the classrooms because the information what is happening inside the classroom now everyone from outside can take from the diodes panel and can get those information without opening the doors. With this project it is achieved that process of digitalization at faculties will be expended even more in order to have panels each door that need to show information what is going on inside.

What about the future

In the future it is planned to be presented and implemented in all the faculties of University “Kadri Zeka” and all the universities of Kosovo. At the same time it is made the request from the working group of professors and students to the Faculty of Computer Science, for such a new implementation. In the mean time will work with working group of students and professors in order to advance with new technology and new way of implementation, also integrating the Wi-Fi technology and also RFID to pass the code to the panels, and even more to add the other informations in the actual classrooms.

Conclusion

There are many ways of digitalization that could happen on the university areas, one of them we have used to digitalize our campus in the university using LED display panels. In our project this feature of panels who do have all the possibilities to write the information and the data could be transfer using USB disk it was very helpful on communicating with the LED panels in our University campus. The software that uses codes makes it easy to write the text, make animations, and set timers to the LED display panel. We have tested and fully implemented on our university campus where we have digitalized all possible information for students, professors, and others who do use our campus.

This was very helpful with U disk exporting the parameters to the LED diode panel display and was quite easy for the places where the panels are reached by people but for the places where the panels are placed on the high height it could not be easy to put the program each time you

change. In order to make easier to put the parameters to panels for the future work we do propose there could be done also communication using RFID and wireless technology.

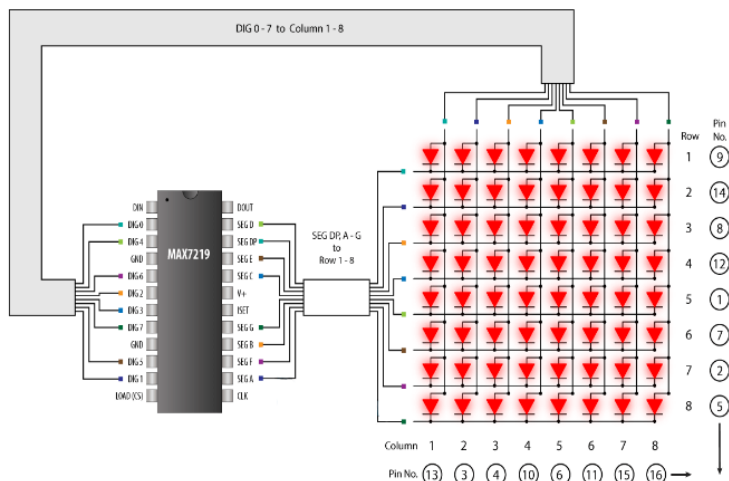


Figure 39. Electrical circuit of LED diode connection in the form 8×8 (Philip Burgess, Arduino and compatible boards)

References

- Bhattachary, S. K. (2000). *Electrical and Electronic Engineering Materials*. New Delhi, India: Khana Publishers.
- Burgess, Ph. (2016). *32x16 and 32x32 RGB LED Matrix*. New York, NY: Adafruit learning system.
- Burgess, Ph. *Arduino and compatible boards*. Retrived from <http://learn.adafruit.com/32x16-32x32-rgb-led-matrix>.
- Hearn, D.; Baker M. (2004). *Computer Graphics with OpenGL*. Third Edition, New York, NY: Pearson Prentice Hall.
- Milman, J.; Halicias, Ch. (1972). *Integrated electronics analog and digital circuits and systems*, Tokio, Japan: International Student Edition.
- Pond, R. J. (2019, March) *Interfacing 8x8 LED Matrix with Arduino*. Retrived from <http://www.circuitstoday.com/interfacing-8x8-led-matrix-with-arduino>.
- Pond, R. J.; Rankinen, J. L. (2008). *Intraduction Engineering Technology*, 8 Edition, Global Edition: PEARSON.
- Smith, W. A. (2016). *C programing with Arduino*. New York, NY: AVR Microcontrollers and ATMEL studio.
- Vision LED, (2019, January 15) retrived from <https://www.youtube.com/watch?v=7gKcGkaIUxM>.