

DEVELOPING AN AUTOMATED LAP TIME MEASUREMENT SOLUTION USING NI EQUIPMENT

Kristóf Barnabás DOMBI

University of Debrecen, Faculty of Engineering, Mechatronics Department, Debrecen, Hungary
kristof.barnabas1996@gmail.com

Abstract

Nowadays, we encounter automation all around us; whether it be at work, in industry, public places, our homes, or on our handheld devices. It is unavoidable. In this rapidly developing world machines are continuously replacing the human workforce, due to their superior workload capabilities, absence of loss of concentration, greater accuracy, speed and efficiency. In this way, machines surpass manpower in many fields. This project presents an automated lap time measurement system built using NI components which will save the voltages tied to the lap time, and measure human and machine lap times. The data will then be compared and evaluated.

Keywords: *myDAQ, LabVIEW, comparison, measurement, data processing*.

1. Short introduction of measuring system

The basis of this project is the system shown in [Figure 1](#). During the task a system was built that included a myDAQ (Data Acquisition Device), optical sensors, the electronics responsible for moving, the necessary power to control the various devices and the program written for the evaluation [\[1\]](#).

There were many ways to connect the two different pairs of elements to each other, a few of which are shown in [Figure 2](#). The aim was to compare the lap time of the man with that of the controller. To do this, however, it was necessary to build a track where both the man and the controller-controlled track section were of the same length.



Figure 1. *The built system*

For the above-mentioned condition to be met, the track section figure "8" must be implemented. The built "8" track section contains two tracks through which the racing cars were controlled. Data from a digital sensor gave me both the number of laps that the racing cars started to run in, and the lap times for each. In chapter 3.4. I will discuss a little bit more.

2. The hardware part of the system

The complete system including all current elements can be seen in [Figure 3](#).

The circuit shown in the [Figure 3](#) can be divided into 5 parts:

- object reflection sensor circuit for the internal track;
- object reflection sensor circuit for the external track;

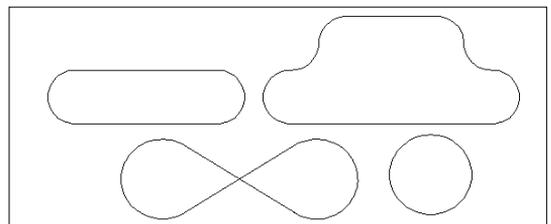


Figure 2. *Possible path creations*

- human-controlled track circuit;
- the circuit of the signal issued by the myDAQ Data Acquisition Device;
- controlling active state of track control signals.

2.1. Object reflection sensors attenuator circuit

The inner and outer track light gate sensors share the same circuit, so they are discussed together. These are the framed portions 1 and 2 in Figure 3. The essence of these circuits is voltage distribution. The sensors transmits 24 V voltage to the output, but myDAQ inputs operate with TTL logic.

$$\frac{R_1}{R_1 + R_2} * U_e = U_r \quad [V] \quad (1)$$

Equation (1) is used to calculate the voltage distribution. R_1 and R_2 are the two resistors used in the circuit, U_e is the voltage at the output of the sensor, while U_r is the resulting "reduced" voltage. The nominal value of the resistor R_1 is 1 kΩ and the resistance of R_2 is 4.7 kΩ.

2.2. Human-controlled track circuit

This circuit is shown in Figure 3. in the „box” numbered 3. The DMM (Digital Multimeter) input of the data acquisition device was used at the beginning to store the signal for several reasons. First, the port has a large measurement interval (60 V). The maximum value of the scanned voltage was 20 volts when it was unloaded. To read

the values for analogue inputs, a voltage divider should have been used, because the maximum voltage that can be used to load the analogue input is 10 V. Using the DMM input also means that the sampling frequency is very small. It follows from the data flow principle that the cycle will not (re)run until all the code lines in it are finished. The result is that the cycle time will be determined by the slowest code line. The DMM port received a sample every half a second, and this became the cycle time..

It can be seen in Figure 4. that in 30 seconds a few (numerically 26) samples were obtained from the myDAQ device. Thus, one of the analogue inputs of the data acquisition device was used to achieve better cycle time. However, this required the placement of a voltage divider between the transmitter and receiver. This could be calculated using equation (1). In this case a 1 kΩ and a 3.7 kΩ resistor was used.

2.3. The circuit of the signal issued by the myDAQ

The role of this circuit is signal amplification and power matching. The maximum output of the myDAQ device is 10 V or 2 mA. This circuit is shown in Figure 3. in the „box” numbered 4. The signal amplification of the built circuit is as follows:

$$A_U = \frac{R_2}{R_1} + 1 \quad (2)$$

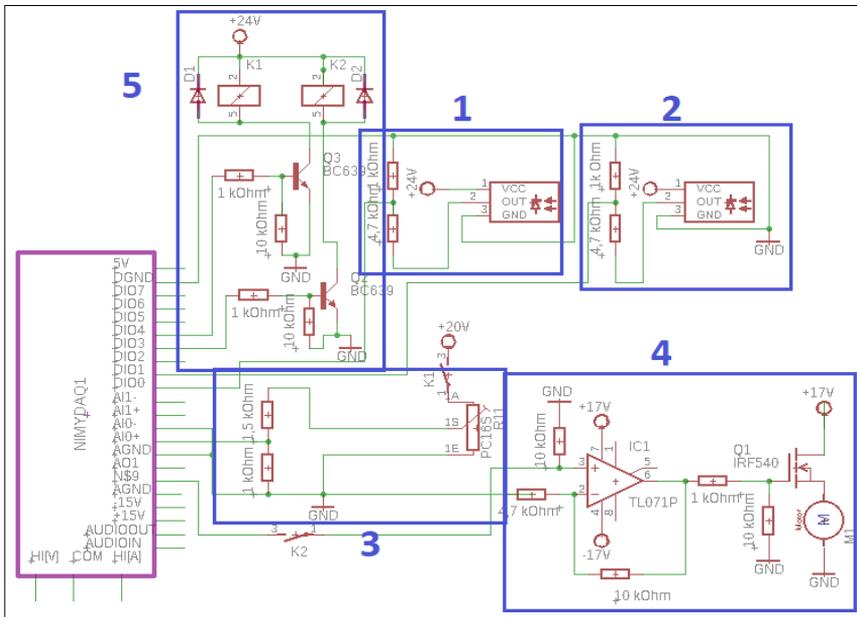


Figure 3. Schematic circuit diagram of the system

By replacing the corresponding values in equation (2), the gain factor, AU, will be 5.25. However, the signal is only 4.7 times higher on the track due to various losses. This value is approximately equal to the voltage divider. This is important to get straight lines with the same linearity, so the incoming data line can be reproduced on the output. Due to the opening characteristics of the MOSFET, a shift of 3.7 V had to be applied to the system. In Figure 5. the amplified output signal image can be seen with offset (g function) and without offset (f function). [2]

2.4. Controlling active state of track control signals

There is a possibility that when the program stops running there is still some voltages on the output. As a result the race car will keep moving until the program gives another low-level signal for the track. This is the reason why the output had to be disconnected directly from the track. Even though the launch of the race is software-initiated by LEDs (Light Emitting Diode) in order to ensure that no car will start sooner than the desired moment I connected the input to a digital signal. I solved this with a BC639 type transistor, which switches to the output LVTTTL signal. This circuit is in the framed section with number 5 in Figure 3.

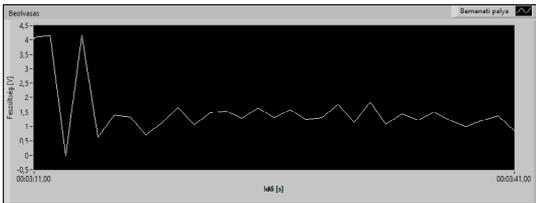


Figure 4. DMM port voltage-time diagram

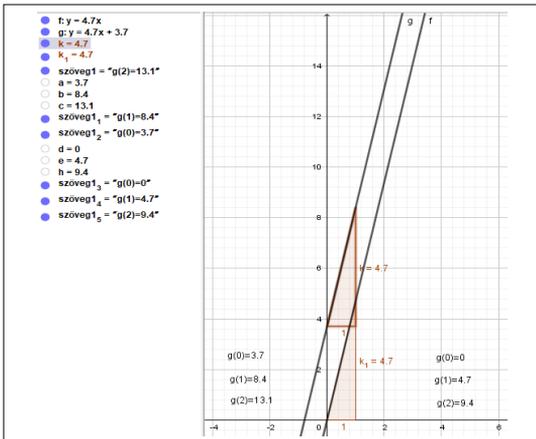


Figure 5. The output signal image with offset and without offset

3. The software part of the system

I used the NI LabVIEW 2015 version to build the software. The system communication is shown in Figure 6. [3]. The software contains a block diagram – including the structure of the program, the connection of VIs, the code – and the front panel, which performs a task like an HMI (Human Machine Interface). [4]

Tab control has been used to divide the front panel into 3 parts. The first is informative, which describes the course and rules of the competition. The second page – which is the main page also – gives feedback from the start of race, about the momentary voltage values, and about the lap times (Figure 7.), while the third is a competition evaluation page.

3.1. Start of a race

This section is shown in Figure 7. in brown and marked with number 1. The program can be started by pressing the "visszaszámlálás indítása" button. As a result of this signal, the upper black LEDs are lit in red and two LEDs turn on after every 800 ms. There is a counter above each of red LEDs, indicating the remaining time. After all red LEDs are lit, the bottom 10 LED's light up green, and a „START” subtitle will appear, which will start the race.

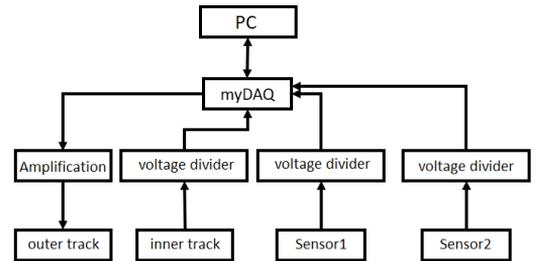
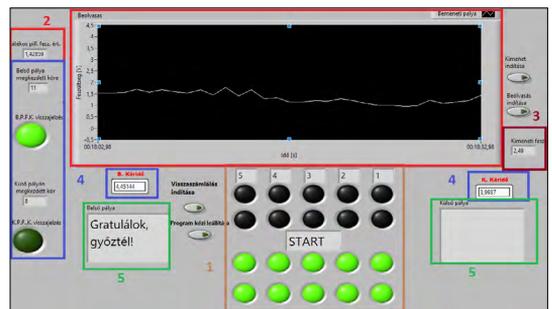


Figure 6. Communication of the built system



7. ábra. Front panel main page

3.2. Data Acquisition

At the moment when the green LEDs light up, the input data recording begins. At the initialisation part of the data read, the folder was created, backup location determined, and file name („beolvasott_fesz_ertek”_év_hónap_nap), extension („.lvm”), then the information to saved was determined, the time and voltage.(„idő”, „beolvasott_feszültség_ertek”) Once the folder has been created, it will store the data at the moment the race is started, according to the settings. The program assigns a voltage value to the current time.

3.3. Control signal read from data series

The output signal is based on a pre-read data line or a pre-written program. The pre-written program divides the track into 2 main parts and changes the voltage values based on them. A mathematical conversion on the scanned data line has been created to improve the already good time. The MOSFET offset error was also corrected here by software. However, since a gain was built up to MOSFET, the gain had to be considered for the correction. The data block required to be printed is first broken down into elements, mathematical corrections made, and then converted into a block for printout and evaluation. [5]

3.4. Time lapse measuring and counting programs

When the signal on the optical sensor output appears, the program adds one to the counter. Timer is placed at the same time as counting because the sampling frequency of the program is fast compared to the slow passage of the car, so it adds more than one unit to the counter. In order to avoid this, a 300 ms timer was built. Each sensor has a separate counter, and the program stops when one of the 11th rounds reaches it. The counters show the started laps, so the number of laps made is 10. A race contains 10 laps.

The program monitors the rising edges of the sensor, that is how it measures the lap times. When the sensor signal first appears on the digital input, a counter starts counting until the sensor detects again. The moment the signal appears again on the input, the time elapsed is stored in an array and then restarted. This process repeats until one of the sensor counters reaches the 11th lap.

4. Evaluation of competition

The last part is the evaluation. Depending on whether a read-back series or a signal generated

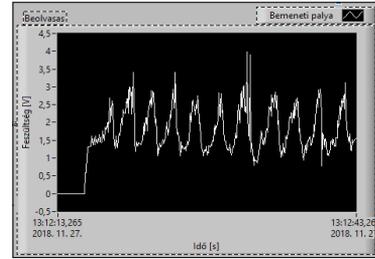


Figure 8. Voltage-time diagram

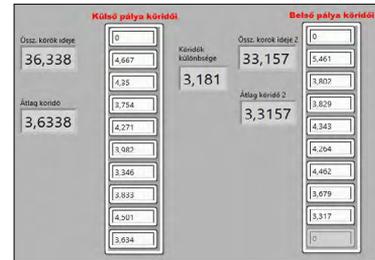


Figure 9. Time lapse evaluation

by the sensors is the one against which the race is taking place, we get feedback in a diagram that shows the voltage as a function of time. [6]

In addition, detailed description of lap times is included in the evaluation. The lap times are collected by a program into a block, and then the elements added to it. This gives the time taken to complete the 10 laps. The average lap time was obtained by dividing the time by the number of laps taken. Finally, the program calculates how much it missed the loser compared to the winner.

References

- [1] Tóth J., Tóth E., Mezei G.: *Automatic car park control with HMI and PLC*. Annals of the University of Oradea Fascicle of Management and Technological Engineering 11/21. (2012) 2.115–2.121.
- [2] <http://www.alldatasheet.com/datasheet-pdf/pdf/52968/FAIRCHILD/IRF740.html> (accessed on: 2018.10.20.)
- [3] Ács V., Tóth J.: *Test station for Reed Sensors and Inductive Proximity Switches*. Annals of the University of Oradea Fascicle of Management and Technological Engineering 16/26. (2017). 99-104.
- [4] Sipos K., Tóth J.: *Elektropneumatikus szelepek áramlási veszteségének mérése ipari eszközökkel*. Műszaki és Menedzsment Tudományi Közlemények 2/1. (2017) 68-72. <http://ijems.lib.unideb.hu/file/9/58edf7d02f0e9/szerzo/2017113.PDF>
- [5] Tóth J.: *Automatika*. 1. kiadás, TERC KFT., 2013. 17-27.
- [6] Tóth J., Kocsis I.: *Mérési, irányítási és diagnosztikai módszerek a sporteszközök alkalmazása során*. Debrecen, Magyarország: Debreceni Egyetem Műszaki Kar (2015) ISBN: 9789634738954