

Control System for a Tensile-Testing Device Using Low-Cost Hardware and Open-Source Software

Matej Kranjec* – Jernej Korinšek – Miha Ambrož – Robert Kunc
University of Ljubljana, Faculty of Mechanical Engineering, Slovenia

The aim of this study is to verify whether a Raspberry Pi 3 B+ can be utilized as a low-cost device for controlling a tensile-testing device used for material research purposes. A list of requirements based on already-available hardware was drawn up, which the new control system had to fulfil. To connect all the necessary equipment, a connection board was constructed, and some additional hardware was acquired for the system to be able to perform all the necessary tasks. The whole controlling system was also put in a small enclosure to make it portable. The control-system software was written in C++ using the Pigpio library. The developed system was then tested, and the results were compared to a commercially available Instron 8802 device. A comparison of the results shows that the upgraded equipment can produce comparable results to commercially available devices and is sufficiently accurate to be applied for research purposes for the characterization of soft tissues and other materials.

Keywords: Raspberry Pi, low-cost, tensile testing, measuring equipment

Highlights

- Existing test device's control system has proven unreliable and has been upgraded.
- A Raspberry Pi was chosen as the core of the new control and data-acquisition system.
- The system was tested and compares favorably to a certified machine.
- The developed system is affordable to most researchers.

0 INTRODUCTION

The everyday challenge of many researchers is to conduct experiments with a limited budget. We strive to spend as little as possible, but still have equipment that is reliable enough to provide accurate and fast measurements as the protocol requires. Because of the age and malfunctioning of our control and data-acquisition hardware, we wanted to upgrade that part of a specifically designed uniaxial tensile-testing machine used for the acquisition of soft-tissue material properties in physiological conditions. The proposed system had to control the motion of a stepper motor and acquire data from a magnetic linear encoder and a tensile load cell.

Many researchers have already searched for a low-cost system to replace or substitute expensive devices for their research purposes and some of them chose an open-source single-board computer (SBC) such as the Raspberry Pi [1] to [4]. The main reason for using the Raspberry Pi is its low cost, small size, versatility and a huge online community that can help solve problems while proposing improvements for new generations of the device. We decided to use the latest currently available version of the Raspberry Pi as the main processing unit for our tensile-testing device. Because of the limitations of the Raspberry Pi, a circuit board had to be constructed to connect the encoder and the motor controller. Also, an analog-to-

digital converter (ADC) had to be added to acquire the data from the load cell. The goal was for the new control system to perform at least as well as the previous system, or even better. The new control solution was expected to be at least 80 % cheaper than a commercially available solution on the market.

In the world of industry and research, there are many commercially available tensile-testing devices from firms such as Instron, Hegewald&Peschke, TestResources, Mecmesin, and Labthink, where the user can buy the whole device with a control system included. Because we already had a tensile-testing device that had been developed in our laboratory and control equipment that was independent of the testing part of the machine, we were able to replace only the control part of the device.

Using the newly constructed control and data-acquisition system we want to show that low-cost equipment and simple open-source software can be used to obtain the tensile properties of materials.

1 MATERIALS AND METHODS

1.1 Tensile-Testing Machine Hardware

The main purpose of the research is to upgrade the control and data-acquisition part of our custom-built mechanical tensile-testing device (Fig. 1).



Fig. 1. Custom-made tensile-testing machine for obtaining soft-tissue tensile properties in physiological conditions

The device is constructed using a standard aluminum framing system (Bosch Rexroth) [5], on which a linear ball screw drive compact module (CKK 20-145, Bosch Rexroth) [6] is mounted, driven by a stepper motor (VRDM 3913/50LWCEB, Berger Lahr) [7] through a 3:1 gear-reduction gearbox (GBX080003K, Schneider electric) [8]. The motor is controlled by a motion-control stepper-motor drive (SD 326RU68S2, Berger Lahr) [9]. For measuring tensile loads, an S-type load cell (CTS63200KC25, AEP transducers) [10] connected to an analogue transmitter (ETA4/2IXO11D24, AEP transducers) [11] is used. The displacement of the carriage, fixed on the linear module, is measured using a magnetic linear incremental encoder system (EMIX 23, ELGO electronic) [12]. The tensile-testing device in this configuration can produce vertical carriage speeds of up to 80 mm/s and generate forces up to 2100 N, while measuring the movement with an accuracy of 0.001 mm at any time.

1.2 Controlling Hardware before the Upgrade

In the previous configuration [13], National Instruments (NI) equipment was used for the stepper-motor motion control (NI PCI-7324) [14] and a NI USB module for the data acquisition (NI USB-6221)

[15]. NI-Max software was used for the configuration of the PCI and USB cards. The experimental protocol was developed using LabVIEW GUI software. The whole process of control and data acquisition was carried out on a personal computer (PC) running Windows XP. The sensor's sampling rate for the soft-tissue tensile testing was set to 1 kHz.

The described system is suitable for experimental work, but only if it is properly set up in both the NI Max and LabVIEW environments. The software for motion control and data acquisition must be properly coded by a trained person, otherwise damage can occur to the cards or controllers.

1.3 Upgrade Requirements

Like the old NI system, the upgraded control and acquisition system has to:

- ensure the correct speed and distance of the motor's movement in both directions,
- reliably monitor the encoder movement at all times while performing any protocol that needs the measurement of distance at any motor speed and direction,
- monitor the analogue transmitter voltage and convert it to a digital signal with at least 1000 divisions of the full load-cell measuring range,
- read measurement data from both the encoder and the load cell with a minimum frequency of 1 kHz,
- collect all the recorded data into one file or separate files that can be later combined into one,
- ensure the repeatability of the measurements,
- be simple to use (an error during component connection has to be impossible) and execute desired protocols (ease of use).

1.4 The Hardware Setup of the Proposed Upgrade

For a cost-effective upgrade, the Raspberry Pi 3 B+ (RPI) was chosen. The RPI is a miniature, low-cost SBC with a performance similar to a low-end desktop computer, but it has the advantage of being portable, upgradable and easily replaceable if any damage occurs. It is also possible to integrate the RPI into another system, as we did. The RPI can be powered via a micro USB connector using a 5 V / 2.5 A power supply or via its 40-pin general purpose input-output GPIO header using a 5 V input connector or even via Power over Ethernet (POE) with a POE HAT [16]. It uses a 1.4 GHz Broadcom's BCM2837B0, Cortex-A53 64-bit System on Chip (SoC) processor and 1 GB of LPDDR2 SDRAM. The connection to the RPI can be established using 2.4 GHz and 5 GHz

maximum operating processor load, incurring a voltage drop. For that purpose, the two 5 V power pins and at least two ground (GND) pins on the connection board had to be connected to improve the current delivery.



Fig. 3. Top view of the connection board

Some extra wire-to-board screw terminal blocks were also added to the connection board for 5 V and 24 V DC power delivery for the motor brake, inverted motor signals (as required by the installation manual of the stepper drive [9]), operational amplifier and other future devices that need a power source, which is provided using two power supplies: one for 24 V DC (IRM 60 24, Mean Well) [23] and the other for 5 V DC (IRM 30-5, Mean Well) [24]. The provided power of 60 W on the 24 V rail and 30 W on the 5 V rail is more than enough to fulfill the system's power

requirements. The detailed and complete connection scheme is available at our GitHub repository [25].

The controlling hardware enclosure was made using a short part of a wall-mounting cable raceway (Fig. 4) on which the holes were drilled for cable connectors. On one end the enclosure is covered with the front plate incorporating the power and error LEDs and the manual switch to override the limit switches. On the other end it is covered with the rear plate on which the RPi, the power switch, the power connector and a 24 V cooling fan are mounted. The plates are our own design and were manufactured in-house using a FDM 3D printer.

Table 1. The cost of the controlling system upgrade

Component	Quantity	Cost [€]
Raspberry Pi 3 B+	1	39.03
HDMI cable	1.5 m	4.99
micro SD card 32GB	1	16.90
multi-pin screw-in connectors	5	7.03
Custom connection PCB	1	10.60
MCP 6002 ADC IC	1	0.32
Baart board	1	20.95
IRM-30-5 PSU	1	13.62
IRM-60-24 PSU	1	15.61
Fan 24V	1	5.25
other small components	approx.	25.00
Total:		159.30

The cost of the control-system upgrade was approximately 160 € (Table 1). In comparison to other



Fig. 4. Custom-made enclosure for the control hardware

commercially available equipment (i.e., NI modules) which can cost €1000 or more each [26], we can easily claim that our proposed upgrade is at least 80 % less costly and affordable to most researchers in the field of small-scale, non-commercial tensile testing.

1.5 Software Setup of Proposed Upgrade

The operating system is based on a Raspbian Stretch Lite install (Version: April 2019, Release date: 2019-04-08) [27], using the latest testing branch repository and a Real Time kernel (4.14.y-RT) [28]. Every system service not needed for the operation has either been removed or disabled in the system software. The controlling software is run on the command line using different command arguments for every protocol implemented (Fig. 5) and the appropriate action commences.

The controlling software is written in C++ and the source code is open and available at GitHub [25]. For GPIO control the Pigiopio library [29] is used. The program consists of three main functions: encoder reading, ADC reading and motor control, each running in its own thread. The functions are then interconnected. Another important part of the program is the data saving and processing.

1.5.1 Encoder Reading

The resolution of the magnetic encoder ELGO 23 [12] is 1 μm when four-edge triggering is used. At the maximum motor speed of 95 mm/s, 95,000 pulses/s must be sampled and processed to measure the correct distance. For this purpose, one processor thread is dedicated to acquiring signals from the encoder at all times. A call-back function with four-edge triggering is implemented, so every time the state of a pin changes, the function is triggered and the position is either incremented or decremented. To keep the software code simple, we did not monitor the Z signal of the encoder, which is used as a 2 mm reference index and is not crucial for distance measurements.

1.5.2 Load-Cell Reading

The load-cell primary sensor is a Wheatstone bridge consisting of four resistive strain gauges [10], which requires its signal to be amplified by a load-cell amplifier [11]. The amplified signal range is between 0 and 10 V and is sampled by the ADC. An SPI reading function is implemented, which sends the appropriate bits for reading each channel and reads the result [19]. After all the required channels are read, it enters

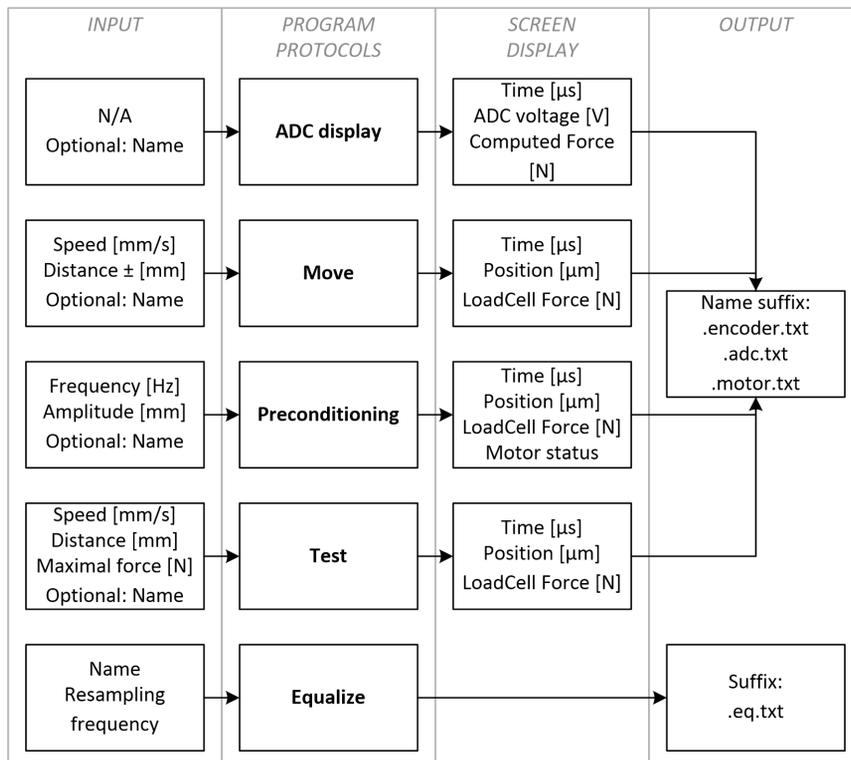


Fig. 5. Schematics of the implemented protocols

the sleep mode for a predetermined amount of time (~101 μs) repeating the cycle. Two variants of the function are implemented: one only for reading and another that signals the motor to stop if the maximum predetermined force is reached or exceeded. The latter includes an additional check of whether the force reaches or exceeds this value in any three non-consecutive samples, to avoid premature signaling due to signal noise.

1.5.3 Motor Control

A pulse/direction protocol is used to control the motion of the motor. For this purpose, a square wave must be sent to the motor controller for every step of the motor. The speed of the motor is controlled by the frequency of the pulses. The shortest pulse the stepper driver can receive is 5 μs (2.5 μs up, 2.5 μs down) defining the maximum pulse frequency of 200 kHz.

Toggleing the RPi GPIO outputs at those frequencies, while maintaining an equally spaced square wave, has proven difficult to achieve consistently. To solve this problem, the built-in hardware pulse width modulation (PWM) was used because it can achieve much higher signal frequencies and is very accurate, as far as the frequency and duty cycle are concerned. This is very important for correct control of the motor speed.

Two operating modes for the motor control were developed: PWM mode and Pulse mode. The PWM mode for motor control is used when an accurate frequency and duty cycle is required. The Pulse mode for the motor control is intended for accurate positioning.

In PWM mode, the PWM hardware on the RPi is used and the duration is calculated. For the duration, the sleep function is used, and the CPU is free to perform other tasks. It also takes some time to switch the PWM ON and OFF. Because of this, the system can overshoot by a few pulses (<10 pulses equaling <0.01 mm on full scale or <0.01 %). In Pulse mode, we generate the square wave signal in software using busy loops in between. However, the frequency and duty-cycle stability of this mode are not very high as it can occasionally miss the frequency by up to 10 %.

1.5.4 Data Saving and Processing

Because the data-acquisition system runs on the RPi's Operating System (OS), the sample times are not equally spaced. In this configuration, the samples are approximately 250 μs apart, but the tests show that the actual interval can range from 200 μs to 300 μs . The

samples from the ADC, encoder and motor are also not sampled at the exact same time. To remedy this design limitation, a data-processing protocol called "Equalize" was implemented.

The "Equalize" protocol takes the data from the measurement files and produces a new file suffixed by ".eq.txt" containing equally spaced time-series data from all three files. This works by taking the latest available sample before each time step while discarding all the other samples in between [25]. A time step of 1000 μs proved to be a reliable choice, producing stable results.

1.6 Validation of Upgraded System

To validate the proposed system upgrade, tensile-testing measurements were conducted on our device and on the reference device, the Instron 8802 system [30]. The protocols (Table 2) consisted of 5 loading and unloading cycles, triangularly shaped (Figs. 6, 7 and 8), using a metal tension spring after pre-tensioning it to a load of 50 N.

Table 2. Validation protocol parameters

Speed [mm/s]	Distance [mm]	Frequency [Hz]
5	10	0.2500
	50	0.0500
	75	0.0333
10	10	0.5000
	50	0.1000
	75	0.0666
25	10	1.2500
	50	0.2500
	75	0.1666
50	10	2.5000
	50	0.5000
	75	0.3333

The results of all the tensile tests were then compared and evaluated. Measurement of the maximum speed and displacement of our device were also conducted to determine the limits of its capabilities. The load cell used on our device had a range of 200 kg and the one on the Instron 8802 had a range of 1 kN.

2 RESULTS

The results of the comparison between our device and the Instron 8802 are shown in Figs. 6 to 9. The left Y axis shows the tensile force from the load cell and the right axis shows the displacement of the tensile

testing head measured by the linear encoder. For representation purposes, all the charts (Figs. 6, 7 and 8) have the same scale on all three axes.

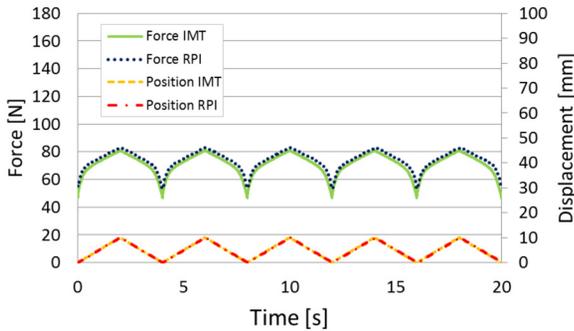


Fig. 6. Comparison of Instron 8802 (IMT) and our device (RPI) at a distance of 10 mm and a frequency of 0.25 Hz

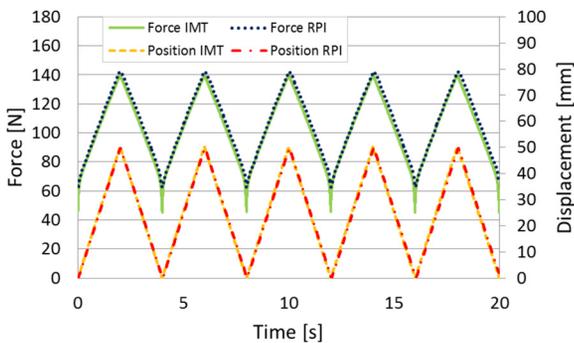


Fig. 7. Comparison of Instron 8802 (IMT) and our device (RPI) at a distance of 50 mm and a frequency of 0.25 Hz

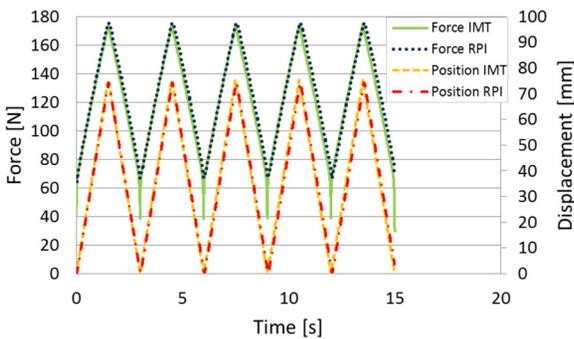


Fig. 8. Comparison of Instron 8802 (IMT) and our device (RPI) at a distance of 75 mm and a frequency of 0.333 Hz

The measured force from the Instron 8802 (IMT) and our device (RPI) shows a small difference of around 6 N, which is constant throughout the whole protocol. The difference is a result of the inaccurate pre-tensioning of the spring before the protocol on each device.

After the data acquisition the measured force from our device was filtered using a 10 Hz low-

pass filter to exclude the spikes that occurred due to sampling the load cell with a greater measuring range than the one on the Instron 8802 [30].

The force and displacement-response differences in time is minimal, but occurs due to the “soft-start” option on motor controller, which limits the starting acceleration of the stepper motor to reduce the wear on the motor and other hardware of the tensile device. The “Soft-start” option could be turned off, resulting in a much slower maximum speed of the tensile device and higher wear on the stepper motor.

All the measurements also showed similar characteristics of the metal spring (Fig. 9), which also reflects in a very high Pearson correlation coefficient (Table 3).

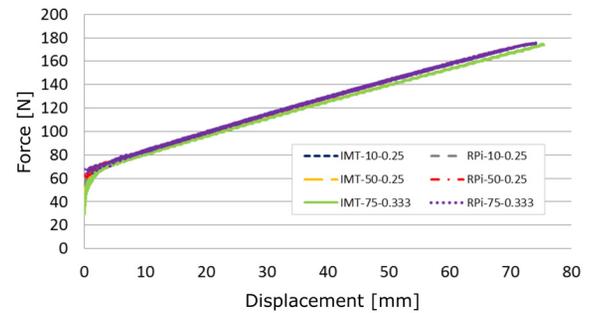


Fig. 9. Characteristics of the metal spring at three different protocols

Table 3. Pearson coefficient for measurements shown in previous graphs

	10 mm, 0.25 Hz	50 mm, 0.25 Hz	75 mm, 0.333 Hz
Pearson correlation	0.99337	0.99531	0.99018

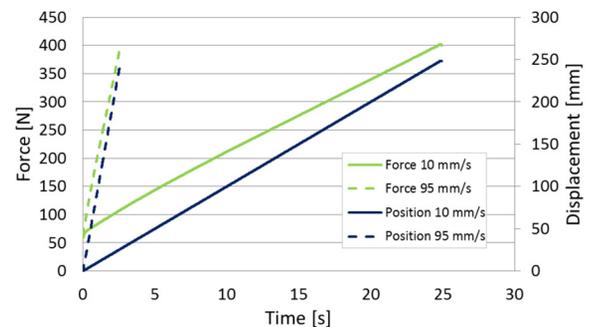


Fig. 10. Metal spring response at 10 mm/s and 95 mm/s loading speed for a distance of 250 mm

Using the same spring, a full-range tensile test was also conducted (Fig. 10) at slow speed (10 mm/s) and at the maximum device speed (95 mm/s). Both

measurements show the same force-displacement response of the spring (Fig. 11).

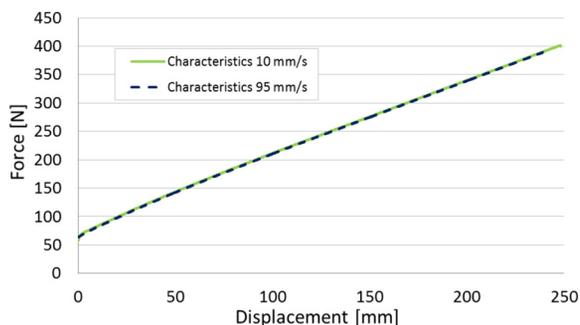


Fig. 11. Metal spring “force-displacement” response at two different loading speeds

3 DISCUSSION AND CONCLUSIONS

The principal purpose of the presented research and development was to test whether the principles of designing low-cost data-acquisition systems can be applied to a somewhat more complex system for tensile testing. The design of the system was adopted from the existing tensile-testing machine, which has proven problematic from the usability and durability point of view. The reasoning behind the component selection was to integrate a system that will provide the same level of measurement accuracy for an affordable price. This was the reason for selecting the low-cost off-the-shelf hardware and the development of own software using previously acquired experience.

The resulting system is a well-balanced compromise between invested resources and the quality of the resulting product. Using our product, we managed to control the whole tensile testing system with enough speed and accuracy to use it for the experimental testing of soft tissues. We successfully replaced the expensive and unreliable existing control equipment with low-cost hardware running open-source software developed in-house.

The results from the comparison of our custom-built tensile-testing device with the upgraded controlling system and a commercially available certified Instron 8802 tensile-testing device show that our device can produce comparable results and is sufficiently accurate to be used for research purposes. The typical dynamic tests of a steel tensile spring, conducted on both systems, yielded results with a Pearson correlation over 99 %.

The upgraded system is currently being used for acquiring the material properties of samples of post-mortem human ligament tissues. Although the

application is currently limited to a single purpose, the controlling and data-acquisition system can be easily utilized for other types of measurements or be transferred to another mechanical system.

Compared to the available turn-key solutions, our system did take more time to design and develop, yet it is more versatile, less costly and easily adaptable, while maintaining a comparable level of accuracy. The development of the presented system also enriched our expertise in this field, which has encouraged us to consider upgrading or developing future measurement systems in a similar manner.

4 ACKNOWLEDGMENTS

This research was funded by the Slovenian Research Agency (ARRS) as part of the Young Researcher Programme for the P2-0109 Research Programme.

5 REFERENCES

- [1] Foster, S.W., Alirangues, M.J., Naese, J.A., Constans, E., Grinias, J.P. (2019). A low-cost, open-source digital stripchart recorder for chromatographic detectors using a Raspberry Pi. *Journal of Chromatography A*, vol. 1603, p. 396-400, DOI:10.1016/j.chroma.2019.03.070.
- [2] Kuziek, J.W.P., Shienh, A., Mathewson, K.E. (2017). Transitioning EEG experiments away from the laboratory using a Raspberry Pi 2. *Journal of Neuroscience Methods*, vol. 277, p. 75-82, DOI:10.1016/j.jneumeth.2016.11.013.
- [3] Kirby, J., Chapman, L., Chapman, V. (2018). Assessing the Raspberry Pi as a low-cost alternative for acquisition of near infrared hemispherical digital imagery. *Agricultural and Forest Meteorology*, vol. 259, p. 232-239, DOI:10.1016/j.agrformet.2018.05.004.
- [4] Ambrož, M. (2017). Raspberry Pi as a low-cost data acquisition system for human powered vehicles. *Measurement*, vol. 100, p. 7-18, DOI:10.1016/j.measurement.2016.12.037.
- [5] Bosch Rexroth standard aluminum framing system, from <https://www.boschrexroth.com/en/xc/products/product-groups/assembly-technology/topics/aluminum-profiles-solutions-components/aluminum-profiles-products/index>, accessed on 2019-07-11.
- [6] Berger Lahr stepper motor, from <https://www.schneider-electric.com/en/download/document/0098441113309/>, accessed on 2019-07-11.
- [7] Bosch Rexroth linear compact module, from <https://www.boschrexroth.com/en/us/products/product-groups/goto-products/goto-linear-motion/ckk-and-ckr-compact-modules/index>, accessed on 2019-07-11.
- [8] Schneider electric gearbox, from <https://www.schneider-electric.com/en/product/GBX080003K/gearbox-straight-teeth-gbx---%C3%B8-80-mm--reduction-3%3A1-%3C-7-arc.min--85-n.m/>, accessed on 2019-07-11.
- [9] Berger Lahr motion control stepper motor drive, from <https://www.schneider-electric.com/en/product/SD326RU68S2/>

- motion-control-stepper-motor-drive--sd326--pulse-direction--%3C%3D-6.8-a/*, accessed on 2019-07-11.
- [10] AEP transducers tensile-compression load cell, from <http://www.aeptransducers.com/load-cells/82-ts.html>, accessed on 2019-07-11.
- [11] AEP transducers analogue transmitter, from <http://www.aeptransducers.com/instruments/164-ta4-2.html>, accessed on 2019-07-11.
- [12] Elgo magnetic linear encoder, from <https://www.elgo.de/en/products/measuring-systems/linear-measurement-incremental/emix23/>, accessed on 2019-07-11.
- [13] Stojanović, A., Omerović, S., Krašna, S., Prebil, I. (2012). Mechanical properties of human cervical spine ligaments: Age related changes. *Journal of Biomechanics*, vol. 45, no. 1, p. S611, DOI:10.1016/S0021-9290(12)70612-7.
- [14] National instruments motion controller NI PCI-7324, from <http://www.ni.com/pdf/manuals/322503b.pdf>, accessed on 2019-07-11.
- [15] National instruments USB multifunction I/O device NI USB-6221, from <http://www.ni.com/sl-si/support/model.usb-6221.html>, accessed on 2019-07-11.
- [16] Power over net HAT, from <https://www.raspberrypi.org/products/poe-hat/>, accessed on 2019-07-11.
- [17] Raspberry Pi 3 B+ specifications, from <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>, accessed on 2019-07-11.
- [18] Efftek ADC Baart board, from <https://www.raspberrypi.org/forums/viewtopic.php?f=93&t=132508&p=883282&hilit=adc+board#p883282>, accessed on 2019-07-11.
- [19] Microchip Technology MCP 3208, from <https://www.microchip.com/wwwproducts/en/MCP3208>, accessed on 2019-07-11.
- [20] Raspberry Pi GPIO overview, from <https://www.raspberrypi.org/documentation/hardware/raspberrypi/gpio/README.md>, accessed on 2019-07-11.
- [21] Electrical specifications of GPIO pins on Raspberry Pi, from <https://raspberrypi.stackexchange.com/questions/60218/what-are-the-electrical-specifications-of-gpio-pins/60219#60219>, accessed on 2019-07-11.
- [22] Microchip technology MCP6002, from <https://www.microchip.com/wwwproducts/en/MCP6002>, accessed on 2019-07-11.
- [23] Mean Well 24V AC/DC power supply, from <http://www.meanwellusa.com/webapp/product/search.aspx?prod=IRM-60>, accessed on 2019-07-11.
- [24] Mean Well 5V AC/DC power supply, from <http://www.meanwellusa.com/webapp/product/search.aspx?prod=IRM-30>, accessed on 2019-07-11.
- [25] Controlling equipment source code "trgalnik", FSKMTM GitHub repository, from <https://github.com/FSKMTM/trgalnik>, accessed on 2019-10-28.
- [26] National Instruments Multifunction modules - shop, from <http://www.ni.com/en-us/shop/select/multifunction-io-category#facet:&productBeginIndex:0&orderBy:&pageView:grid&pageSize:&>, accessed on 2019-07-11.
- [27] Raspbian Stretch Lite: Version: April 2019 Release date: 2019-04-08), from <https://www.raspberrypi.org/downloads/raspbian/>, accessed on 2019-07-11.
- [28] Real Time kernel 4.14.y-RT, from <https://github.com/raspberrypi/linux/tree/rpi-4.14.y-rt>, accessed on 2019-07-11.
- [29] Pigiopio library, from <http://abyz.me.uk/rpi/pigpio/>, accessed on 2019-07-11.
- [30] Instron 8802 Fatigue testing system, from <https://www.instron.us/en-us/products/testing-systems/dynamic-and-fatigue-systems/servo-hydraulic-fatigue/8802>, accessed on 2019-07-11.

6 APPENDIX

Supplementary material consisting of measurement results, protocol, data saving and processing descriptions, source code and the connection scheme of the whole system is available at our GitHub repository: <https://github.com/FSKMTM/trgalnik>