Design of low-cost robotic arm for education

Filip Tóth
Faculty of El. Engineering and Information Technology
Slovak University of Technology
Bratislava, Slovak Republic
ftxrobot@hotmail.com

Boris Rohaľ-Ilkiv, Pavol Krasňanský
Faculty of Mechanical Engineering
Slovak University of Technology
Bratislava, Slovak Republic
boris.rohal-ilkiv@stuba.sk

Abstract – **The goal of this project was to design and implement the electronics for a robotic device – namely a robotic manipulator. The manipulator has a total of six degrees of freedom, three of which are on the manipulator itself, and the rest are in the wrist. A gripper is used as the end stage of the device. It was necessary to design the electronics so that they could be interfaced with standard hardware and software as that which is available in the automation labs of STU. A similar robotic manipulator will be used as a part of a mobile robot in the future.**

Keywords: robotic manipulator; degrees of freedom; electronics

I. Introduction

Pavol Krasnansky, a student of the Faculty of Mechanical Engineering of STU is the author of the original idea. I was given an opportunity to create the electronics of this device, which I took. The development of the electronics for the device has been going on for a total of three years so far. The goal of the project is to be able to control and solve direct and inverse kinematic problems via MATLAB.

II. The mechanics

The whole arm (fig. 1.) is connected onto a base, in which the drive and the mechanics for the first degree of freedom is mounted in. The second, third and fourth degrees are implemented inside the arm itself. The fourth degree of freedom is the gripper's wrist. The fifth degree is the rotation of both the gripper and the held instrument. As for the gripper itself, it is made up of two opposing "fingers", which move against each other symmetrically – they are the sixth degree of freedom.

A classical serial kinematic structure requires a fair amount of power from the motors – this is why the joints of the manipulator are driven through a worm drive gear arrangement. This provides great strength with relatively small motors, a zero backlash and, thanks to the self locking properties of this arrangement, zero power consumption while not moving. The disadvantage of this solution is the relatively low speed of the system.

Each joint (fig. 2.) has its own servomotor, an absolute rotary encoder, two limit switches for the maximal and minimal angles and the worm drive arrangement.

The highest requirements for torque are on the second degree – it is a joint that has to move the greatest weight (effectively the whole weight of the manipulator as well as any objects held by the hand). Also, the arm acts like a lever force which increases the torque requirements in this joint. Therefore a motor with a higher torque than anywhere else was used here.

The gripper itself has two fingers which create a parallel kinematic structure – thanks to this; the fingers are always parallel to each other. There are pressure sensors in the area in which the gripper makes contact with the gripped object. The total length of the manipulator (when set to a straight alignment) is 580mm. It has a total weight of 5075g. The actuators used are two types of servomotors (see table 1). In the second degree, an industrial TONEGAVA SEIKO servomotor with steel gears is used. In the rest of the degrees, Hitec HS-5955TG servomotors (professional RC modeling servomotor with titanium gears and shaft) are used.

MAB25 absolute rotary encoders are used for the sensing of the angle between joints. These are connected directly onto the shafts of the joint gears.

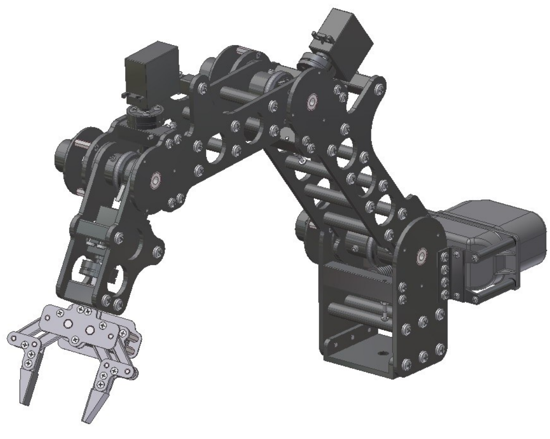

Figure 1. The whole arm

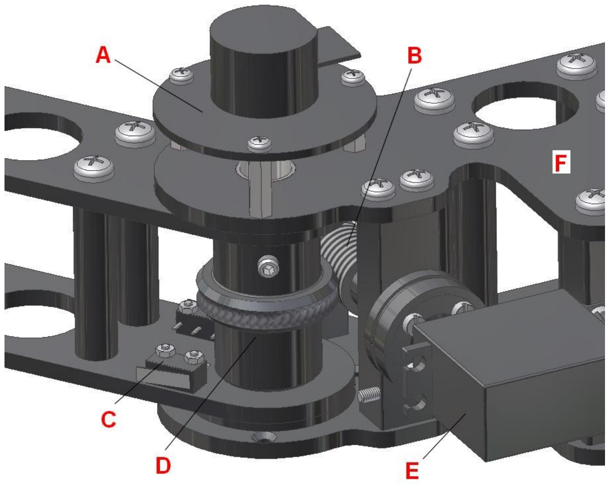

Figure 2. Joint assembly

A – absolute position sensor MAB-25, **B** – screw gear, **C** – safety switch, **D** – screw gear, **E** – servo – drive HS-5955TG, **F** – duraluminium

TABLE I. PARAMETERS OF THE ACTUATORS

	HS-5955TG	**PS-050**
Gears	titanium	steel
Torgue [kg.cm^{-1}]	31.2	91,5
speed[s / 60°]	0,15	0,29
Dimensions [mm]	40 x 20 x 37	100 x 44 x 79
Voltage [V]	4,8-6	5-12
Weight [g]	62	283,4

III. THE ELECTRONICS

Having electronics as a hobby I spent a great deal of time on its design, and tried to perfect it. I started designing the electronics (fig. 3.) once the mechanical part was almost ready, so there wouldn't be any changes in the electromechanical components of the manipulator (actuators or sensors).

Before the design phase it was necessary to analyze the requirements for this system, clearly define goals and priorities. These goals were set: Modularity, programmability, minimal cabling, robustness, software access to all parameters available, miniaturization, maximal power usage efficiency, connectable with a standard system, digital control of analog inputs/outputs, self-diagnostics capability, and safety.

I would like to point out that the device is completely original. Every part of it is a prototype.

A. The Hitec joint module

The Hitec joint module is a printed circuit board with control electronics, onto which the HS-5955TG servomotor, two limit switches, the absolute rotary encoder, the communication bus and the power for both the logic and power part connects. The board is fixed onto the segment between two degrees of freedom and is connected with the motor of that joint (relatively to the board, the motor is stationary). Since one of the goals of the design was modularity, each degree of freedom has its own module. These modules are connected onto a common bus and have a common power supply. The advantage of such a solution is that only six wires are used to connect all of the modules and enable the function of the whole system. These six signals are: Transmit data, Receive data, System ground, +5V, +7.4V, Power ground. In comparison, a centralized system would require a total of at least 64 cables going from the manipulator onto a central electronics board.

The other modules, such as the Tonegava joint module and the Joint module with grip force measurement are very similar to the basic module, with some minor differences (such as the extra electronics for force sensing).

The heart of the module is an ATMega8 [1] processor – an Atmel RISC processor. In this application I am using the ADC (analog to digital converter), UART interface (standard serial line), ISP (In system programming) interface, the PWM (Pulse width modulation) generator and the GPIO pins (general purpose input output).

A 14.745600MHz crystal is used as the clock source for the device, which is a frequency that allows both maximal computational power as well as precise, whole number settings of timers and the serial line.

To inform the user about the state of the device, three LED diodes are used (red, green and blue). Two of these (the red and green ones [2]) are connected directly to the GPIO pins of the processor, the third one is controlled by an N channel, IRF7341 MOSFET transistor [3].

The servomotor is connected in the classical manner of RC motors – 3 wires, one signal, the rest are for the power supply. The control signal is connected in series with a 1kΩ resistor onto a pin of the processor. The servomotor has been modified to allow continuous rotation of the shaft instead of the normal +/- 90° operation. The motor's speed and direction is controlled via a PWM signal – depending on the width of the signal, the motor will either rotate CW, CCW or halt.

To reduce the effect of the noise created by the power parts, the logic section and power section have their own power, which is not connected to the other.

To enable the measurement of the current the motor draws, a 0.1Ω shunt resistor is used to monitor the current on the motor. By measuring the voltage on this resistor through the ADC, the current being used by the motor can be accurately measured. Also, the ADC is used to monitor the voltage for the control part and the power part of the system. When there is no need for the motor to be turned on, the microprocessor disconnects power from the whole power section to conserve energy. To do this, it uses a P channel MOSFET transistor, namely a Si4435BDY [4]. Because the logic voltage is lower than the power voltage, a P channel transistor cannot operate directly from a GPIO pin of the processor – these transistors are controlled by the voltage between their Gate and their Source pins. The lower the voltage, the higher the resistance. In this case, +5V on the Gate would still result in an open transistor. Therefore an extra transistor is used to control the Gate voltage – an N channel transistor (namely IRF7341 [3]), with a pull-up resistor from the +7.4V. This one is controlled by a GPIO of the processor. This setup controls the main transistor. Using this setup, I have achieved a R_{DS} value of less than 0.02 Ω, which, in the case of a maximum current of 5.2Amps creates a 0.5408W of waste heat, which is within the normal operational parameters of the Si4435. To improve heat dissipation, a heat sink is formed on the PCB, to enable the transfer of heat via the pins of the SO-8 package. When all six degrees are at their maximal power consumption, the total consumption can be as high as 31Amps. Since all the modules are connected into a series via a bus, 31Amps would flow through the first module. This is the reason why there are T connectors used in the latest version – they have a really big contact surface, and are designed for a continuous current of 40A. The other connectors are standard, low current, low voltage, Wire to board connectors.

The absolute position encoder is connected to the module via a 6 signal interface. It is a standard SPI interface, through which the 10 bit information about the current position is transmitted [5]. The encoder is powered by +5V. It is a electromagnetic rotational encoder, type MAB25, made by the MEGATRON company.

The communication interface can be configured to work either as a multiplexed serial line, or it can be used as an I^2C bus (or similar). In the normal configuration the modules communicate with an external device that acts as a master on the bus and gives commands to the modules. A special bus driver for the Transmit signal is used, so the bus acts as an open collector system – this allows a module to transmit and receive without first notifying the other modules. It also allows for a non destructive collision. Each module has its own pull up resistor. The Receive signal is common for all modules – they all receive the commands from the master. The idea of this bus is that the modules are slaves and behave on a "speak when spoken to" basis. The bus driver used is a 74HC125 [6]. There is also the option of using I^2C as the main bus, where SDA (bus data) and SCL (bus clock) take the place of the Receive and Transmit. The advantage of such a configuration would be that the modules might communicate between each other (the I^2C standard allows for multi master communication [7]), or simpler connection to an existing system.

The module also has a RESET button, a jumper usable as a user control (for long term mode changes), a connector for the two limit switches (used as a backup in case of a problem with the absolute encoder or as a simple testing/calibration tool). On each module there's an ISP connector, using which it is possible to program the microcontroller with the appropriate firmware. For programming a proper programmer must be used – various open implementations exist on the internet, as well as commercial products.

This is the third version of the module – since the start of the project, the total board area has decreased by an impressive 57%.

There are mounting holes for M3 screws on the PCB. They are used to fix the module onto the manipulator. Also spacers are used to fix a transparent protective shield above the board.

I designed the PCB in OrCAD. It is a two layer PCB with a soldering mask on both sides.

B. The Tonegava joint module

The Tonegeva joint module is almost identical to the Hitec module. The difference lies in the type of servomotor used – which is a PS-050 TONEGAVA SEIKO servomotor. This servo has a separate control part and a separate power part – the control part has its own power as does the power part. The control part connects to the board like a standard servomotor – three pins – +5V, PWM signal and GND. The power section only connects via its power pins - +Vs and GND. Any voltage, ranging from +6V to +12V can be used to power it. All of the other parameters of this module are the same as those of the Hitec module.

C. The gripper module

The gripper module is similar to the Hitec module, the main difference being that there is no absolute position encoder but force sensors. These sensors are mounted directly onto the gripper's fingers. The sensors used are FSR-150AS from the German company FSR-Sensoren. The sensor is a resistor, whose resistance changes when force is applied to it. Its small size (12x12x0.5mm) makes it ideal for mounting onto the grippers fingers. Depending on the strength applied to it (from 0.1 to 100N) the resistance changes (from several MΩ to less than 1kΩ). This wide range of output values makes it somewhat problematic to work with. Fortunately, the maximal force of the fingers is 30N, and I've limited the minimal force to 1N. With this I've gained a sensor with an input 1 to 30N range and a 10k to 1kΩ output range. The maximal current through the sensor is 1mA, its temperature drift is 100ppm/˚C. The sensor is connected into a simple voltage divider with another resistor. The output from this setup is fed into the ADC of the processor.

D. The ATMega128 module

The ATMega128 module is a breakout module for the processor (an ATMega128 [8]) and a few discrete supporting components required for its operation.

The reason for this board is that the processor used is an SMD component and it would be difficult to replace in case of it being damaged. Also, different modules can hold different programs, so software development can be helped by this. All of the pins of the processor are connected to pins. The board can be connected into an appropriate socket.

Aside from the processor and the pins, the only components on the board are a LED and a RESET button. The module is the heart of the communication and HMI module.

E. The communication and HMI module

The communication module (fig. 3.) is a PCB that is not mechanically connected to the manipulator. Its purpose is to provide a “middle man” between the manipulators electronics and the measurement card on the PC. Another task of this board is providing analog inputs outputs for various degrees of freedom using a standard voltage signal (0-10V). The card has 8 such inputs and 8 outputs. Aside from this it has two RS232 serial ports and a special D-Sub connector for connecting the manipulator itself.
There is also a HMI (human machine interface) on this device. The user can control and monitor the robotic arm using buttons, LEDs, a piezoelectric transducer, potentiometers, two encoders and a 4x20 character LCD display.
The main processor used here is the ATMega128, which is mounted on the above mentioned module. All of its pins are used.
The board provides a power supply for both the power part of the manipulator and the logic part. The power supply is heavily filtrated using relatively large capacitors. The power supply is also protected by diodes and fuses. Only the logic power supply is monitored, the power part is monitored and switched by the modules themselves.
The angles of the joints of the manipulator can be set via analog the analog inputs. There are 8 of these inputs. The internal ADC of the ATMega128 processor is used for their measurement. A precise voltage divider, manually adjustable by a precise multi turn trimmer, is used to convert the standard signal (0-10V) to the range of the ADC (0-2.56V). The ADC is a classic 10bit successive approximation ADC with a maximal sample rate of 15kSps.
Unfortunately, the processor has no DAC (digital to analog converter), therefore it cannot create the analog voltage outputs by itself. Therefore, and external DAC was used. The requirements for this DAC were: voltage output, internal reference, at least 10bit resolution, a simple power supply (not symmetrical, preferably from +5V) and a parallel communication bus. In the end I chose the MAX530 DAC – a low power 12 bit DAC, with multiple power and output options. It also has an output buffer and an internal configurable reference - +2.048V, +4.096V, +/-2.048V, or an external reference can be used. It’s quite fast – a settling time of only 25µs. It also has several options for connecting it to a data bus and controlling it – either the 4 bit interface, when the three parts of the 12bit word are written in sequence, or, the 8 bit interface, where 8 and for bits are written in sequence. Aside from the main control bits, there are also control bits, which take care of the conversion and the settings of the DAC. A total of 8 pieces of these DACs are used here. If each one was connected individually to the processor, a total of 112 pins would have to be used. Instead, all of the DACs share a common data bus. Some of the pins are connected onto a 74HCT154 multiplexer, which is addressed by the processor and picks the appropriate DAC to work with. Thanks to this, only 16 GPIO pins of the processor are required to operate all of the 8 DACs.
Since the required output of the device is of the range 0-10V, amplification of the signal was required – all of the DACs outputs had to be put through an operational amplifier. The advantage of an opamp is that its output signal will be stronger than that of the DAC alone, also it will have a higher amplitude. It is simple to set the required voltage gain. The disadvantage is, that it is another block in the analog path of the signal – it can add nonlinear transfers, noise, offsets and such to the signal path and degrade the signal. To avoid additional problems with this I didn’t choose normal opamps, but rather precision opamps, which enable me to set various parameters and fine tune the amplifier stage. Also, it must not need symmetrical power, and have a maximal output voltage of at least 10V. In the end I chose the TLC271CP opamp – a low noise, precision opamp, with a typical noise of 25nV. It also allows the users to choose from three Bias modes – these define key parameters of the opamp and enabled me to choose the best mode for my application. The modes are HIGH, MEDIUM and LOW. Each of them is a compromise between power consumption and suppression of the negative parameters. I chose the HIGH bias mode, which almost nullifies the offset. The tradeoff is a higher power consumption. The power supply for the opamp is +12V. The output of the opamp is properly loaded using a resistor to the ground, and through a protection resistor it is connected onto the 0-10V output of the module.
There are holes for various spacers on the board – these spacers hold both hold the Perspex casing of the board and hold the board above ground.
The PCB is a four layer PCB, with a solder mask, and a description layer (with part description and helpful texts) as well, designed in OrCAD.

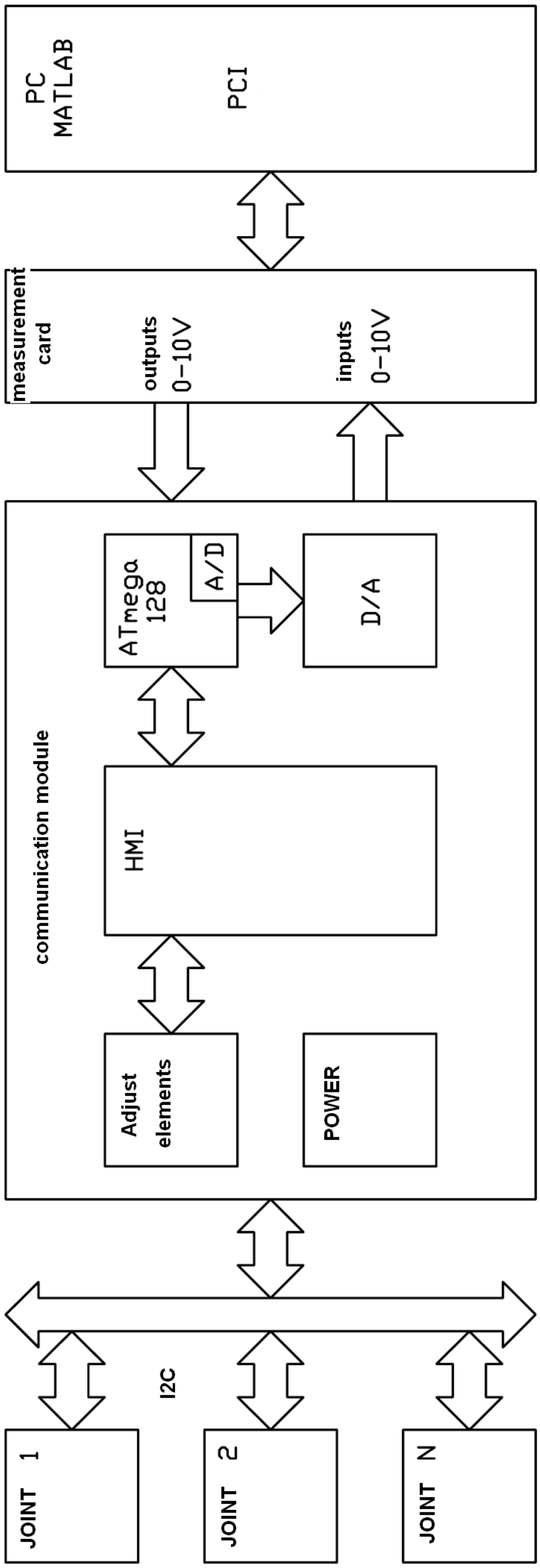

Figure 3. Block diagram of the electronics

IV. Module interconnection

All joint modules are connected parallel – they have a common communication bus and power bus. The connectors used are "Wire to board" for the data bus. It is a 4 pin connector, with the signals Transmit Data, Receive Data, System Ground and +5V. A common problem in robotic systems is how and where to attach the wires to avoid too much mechanical stress. It was necessary to choose the proper type of cable for this application. The best wires turned out to be flexible 0.25mm^2 silicon cables for measurement devices – their core is made out of 128 0.05mm threads.

T type connectors were used for the main power distribution. When dealing with high current, all additional resistance can prove itself to be a problem. The last module in the series will have to deal with a voltage that is down by all of power dissipated on the previous stages. Standard connectors used have a relatively low maximal current rating and relatively high resistance. Professional connectors which are meant for this kind of current have a too massive construction. In the end I chose RC hobbyist connectors – they have the best size-to-resistance ratio. Their resistance is within a few mΩ. Similarly, appropriate cables must be used. For the main power distribution I chose highly flexible PVC cables with an area of 2.5mm^2. Their core is composed of 651 fine wires with a diameter of 0.07mm. The most stressed cables are those connecting the force sensors. The gripper head can be rotated by up to 360°. The sensors are mounted at the end of the gripper head. Since the current won't rise above 1mA, it was possible to minimize their diameter. I used a highly flexible "LIFY" cable with an area of 0.05mm2. Its core is composed of 26 threads with a diameter of 0.05mm.

A special hybrid D-Sub connector is used to connect the manipulator itself and the communications module. This connector has 17 signal pins and 7 high current pins. The high current pins are designed to withstand a continuous current of 40A. They are gold plated with a 0.8 micron layer. The size of the connector is identical to a classical 50pin D-Sub connector.

On the communications module there are sockets for classical, 4mm banana plugs. This type of connection is common in labs and is very practical.

V. Design, manufacture and testing

I did this design based on the experience I got from other designing other robotic systems and on the application suggestions from the manufacturers. I tested out most of design on a bread board first. When the design did not meet my requirements I looked for a better solution.

The main design tool I used was Cadence OrCAD – a software package for the design of printed circuit boards and electronic schematics.

Most of the PCBs used have two layers. The communications module was made with 4 layers. All of the parts were soldered manually. After assembly, each module was tested individually and loaded with 100% of the designed load capacity for a few hours.

The Perspex protection covers were designed in AutoCAD – a popular CAD system for mechanical drawing. The output was processed into a program for a CNC machine which cut out all of the components.

VI. Education

The robotic manipulator is currently being used by students to solve basic robotic kinematic problems, direct and inverse kinematic problems as well as problems involving the calibration of the kinematic structure of the manipulator. Students solve these problems in the form of individual assignments, using AVRStudio, MATLAB/Simulink and utilize the connection of the manipulator to a computer. After completing these basic tasks, groups of students are faced with more challenging tasks – such having the end point of the manipulator follow a predetermined path.

Assessment

Currently, the mechanical part of the manipulator is undergoing reconstruction. So far, all of the components have been only tested by themselves. When the mechanics are completed, the electronics will be mounted onto it, which will animate the so far static manipulator. After this, it will be necessary to fine-tune all of the components together. After that, the programming of the controlling microprocessors may commence and the creation of the base algorithms for controlling the manipulator. Then, finally, it will be possible to use the manipulator to solve and simulate kinematic problems and try and solve other similar control problems. MATLAB doesn't need to be the tool used, since many other software packages can use the DAQ card.

Thanks to the experience gained on this project, it is probable that a new version of the manipulator will be developed, with better dynamics, lower weight. This manipulator will be mounted onto a mobile platform (with special mecanum wheels and a powerful computer), which can be used to autonomously explore an area or do similar tasks.

References

[1] Atmel Corporation: 8-bit AVR with 8K Bites In-System Programmable Flash , [online] Publication in May 2008, http://www.atmel.com/dyn/resources/prod_documents/2486s.pdf

[2] Kingbright Corporation: Surface mount LED lamps, [online] Publication in 2008, http://www.sos.sk/pdf/king/king-smdled.pdf

[3] IR WORLD HEADQUARTERS: IRF7341 HEXFET Power MOSFET, [online] Publication April .2005, http://www.irf.com/product-info/datasheet/data/irf7341.pdf

[4] Vishay Corporation: P-Channel 30-V (D-S) MOSFET, [online] Publication 03.03.2003, http://www.datasheetcatalog.org/datasheet/vishay/72123.pdf

[5] Megatron Elektronik AG & Co.: Halleffekt absolutwertgeber serie MAB25 SER, [online] Publication 28.02.2008, http://megatron.de/Impuls1/DB_MAB25.pdf

[6] SGS-THOMSON Microelectronics: QUARD BUS BUFFERS (3-STALE), [online] Publication in 1994, http://www.datasheetcatalog.org/datasheet/SGSThomsonMicroelectronics/mXrytxt.pdf

[7] TwiNgMan: Zbernica I2C, [online] Publication 17.04.2007, http://new.twingy.sk/rozne/zbernica-i2c

[8] Atmel Corporation: 8-bit AVR Microcontroller with 128K Bites In-System Programmable Flash , [online] Publication December .2003, http://www.datasheetcatalog.org/datasheet/atmel/2467S.pdf