D. Sukhotskiy and A. Yudin “Startup robotics course for elementary school” in Research and Education in Robotics - EUROBOT 2010, ser. Communications in Computer and Information Science, vol. 156. International Conference, Rapperswil-Jona, Switzerland, May 27–30, 2010, Revised Selected Papers: Springer Berlin Heidelberg, 2011. pp. 141-148.

Startup robotics course for elementary school

Dmitry Sukhotskiy, Anton Yudin

Moscow Information Analytical Center (IAC), Robotics laboratory,
1-y Zborovkiy pereulok, 3, 107076, Moscow, Russia
Bauman Moscow State Technical University, IU4 Department,
2-nd Baumanskaya st., 5, 105005, Moscow, Russia
dsuhotsk@gmail.com, skycluster@gmail.com

Abstract. The article is an example of preparatory course for studying robotics by junior school children and aims at forming an educational framework for future expansion of the course. Main topics for studying were selected on the basis of author’s experience at elementary school age. Recommended curriculum and examples which are illustrative and easy to understand at the junior age are also discussed.

Keywords: Eurobot, robot, education, programming, elementary school.

1 Introduction

Author experience includes active participation in scientific and technical exhibitions and conferences and robot development for Eurobot [1] and Eurobot Junior competitions for more than 3 years. When grownups see the work of a kid hardly out of elementary school they usually ask one and the same question: “How did you start? What is the right way to start teaching robotics to younger children?”

At initial phase of studies with children it is important to form the right conception of how basic robot components (programs, electronic parts and actuating mechanisms) interact with each other. At the end of this phase a child will know the ultimate purpose of work, the concept of a robot. As a result wish and motivation to move on with robotic studies will arise.

Based upon author’s experience an educational programme is developed which includes logic elements explanation, visual programming and development of a simple robot, controlled with a personal computer (PC).

With assistance of Russian Eurobot NOC [2] a series of seminars were held with a group of pupils who afterwards participated in Eurobot Junior 2010 competitions. It made possible to validate and update study topics of the course and to form a list of necessary equipment and parts for it.

2 How does technical knowledge form in children?

Child's early awareness of electrical engineering elements and electronics is driven by modern life conditions. Children are facing electronic devices (lighting and heating devices, cooking machines and other consumer electronics) at extremely young age.

Toys with electronic parts usually trigger initial interest to look inside and to understand "how it works." By doing so, a child is getting basic knowledge of electric circuits and associated parts: current sources, lamps, motors, switches, etc. It is possible that from such toys the interest to robotics arises.

A major role in studying electronics play different sorts of construction kits. In Russia there is a kit named "Znatok" [3]. Easy-to-use electrical connections and a lot of circuit designs of different complexity make it unique. One may use it in learning process for as long as 2 or 3 years.

Amount of useful information in a toy is limited and fast to comprehend. But nowadays kids have a rather permanent "toy" – computer. There is even a term – computer kids. The answer to a question of usefulness of computers highly depends on the purposes the computer is used for. In robotics studies PC is an irreplaceable assistant. From early age children may draw, type text and understand that computers control such devices like printer or scanner. It is enough to give child and idea that same computer can control any simple device, e.g. a toy. And the aim of the proposed course is to teach them how.

3 Study topics

There are no exact seminar in-depth descriptions in this article. Instead, several most important recommended subjects and examples are presented. They are intuitive and easy-to-understand for kids.

Conscious interest in knowledge children show only while in game. That is why it is important to conduct only practical seminars in game style. First part of such seminar should be dedicated to circuit assembly and the second – to programming. Circuits are assembled on a bread board without any soldering.

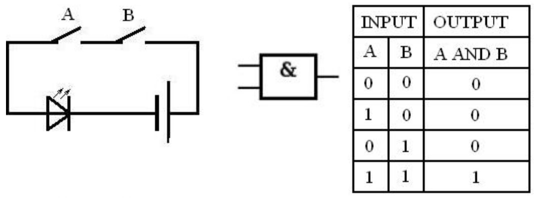
It is necessary to include these topics in curriculum:

- Logic elements;
- Visual programming environment Borland C++ Builder;
- Proteus Design Suite – for modeling electronic circuits;
- Motor driver L293D;
- Resulting robot assembly, able to be operated with computer.

3.1 Logic elements

Logic element is an electronic device in which any state of input signal is associated with a definite state of the output signal. It is not necessary to assemble all possible circuits of basic logic elements on a bread board. Any of the circuits gives a straight idea of how logic works. But truth tables should be thoroughly reviewed.

First seminar should be started with a simple AND element and its 2-switch representation. Circuit diagram, symbol and truth table for this element are presented below. Logical "one" is produced by the element only when both inputs are logical "ones". This explains the name of this element as "ones" should be present on one AND the other (both) inputs.



INPUT		OUTPUT
A	B	A AND B
0	0	0
1	0	0
0	1	0
1	1	1

Fig. 1. Circuit diagram, symbol and truth table for AND element.

When explaining what a composed element is a simple NAND element can be used. Its work is similar to AND, but output signal is opposite to the input. When AND has logical “one” on output, NAND has logical “zero” and vice versa. This is easy to understand with the help of the figure below.

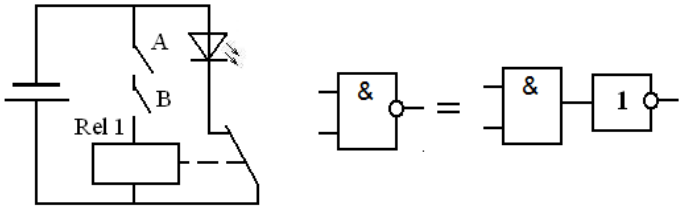


Fig. 2. Circuit diagram and equivalent symbol of NAND element.

Combination of logic elements is demonstrable with SR flip-flops based on two logic elements.

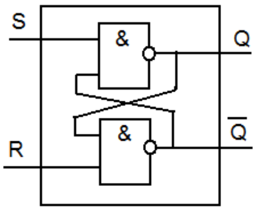
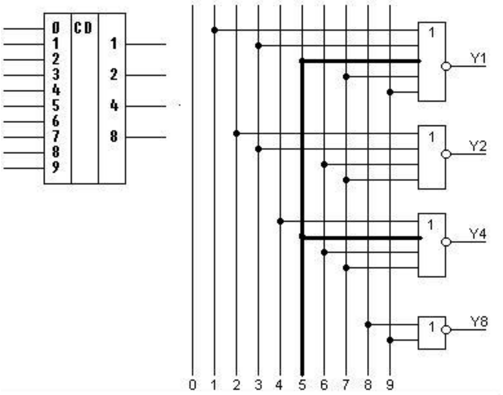


Fig. 3. SR flip-flop.

Flip-flop is a logic element that can be for a long time stay turned on and cycle its states according to external signals. Here it is important to revise truth tables of NAND elements once again and compare to that of a flip-flop.

Assembly of simple circuits of encoder and decoder can help to make the first step in learning binary arithmetic. Encoder translates single input signal to binary code on output.

In the next figure one can see an encoder based on 4 OR logic elements. Diagram shows an input signal for decimal “5” number. Diagram visualizes that when logic one is introduced to input dedicated to decimal "5" first and forth outputs will generate logic "ones", thus forming 0-1-0-1 binary number.

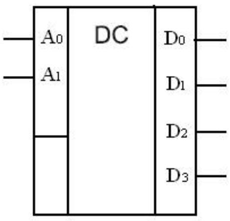


DECIMAL	Y8	Y4	Y2	Y1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Fig. 4. Encoder.

For back translation of binary to decimal numbers special decoders are used. When a binary number is on the input of the decoder a dedicated single output is activated.

In figure 5 one can see the symbol and truth table of the 2-to-4 decoder. During the seminars it is sufficient to look only into the truth table.



INPUT		OUTPUT			
A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

Fig. 5. Decoder.

It is important to note that decoder can be used to decrease the number of connections between electronic components; in our case it is possible to control 4 devices one by one. For real applications it is better to use a 4-to-16 decoder.

3.2 DLP-USB245M, UM245R modules

During the course seminars DLP-USB245M or UM245R module was used for interfacing computer to peripheral devices. These are converters of USB signal to simple parallel interface. One doesn't need to know USB protocol to operate these modules but it is required to download and install module drivers and libraries. Each module has a number of useful functions and modes. For educational purposes only one is sufficient – connect all 8 channels of the module as outputs. In such mode it can be compared to previously discussed encoder. It should be called repeater though as it repeats on module outputs the number submitted to computer program.

3.3 IDE Borland C++ Builder Programming

First programming lessons should be more visual, for example one could use Borland C++ Builder environment. Several lessons will be enough for children to learn a number of primitives and how to use them in a program. These could include buttons, text editor, progress bar, etc.

For startup projects while on initial stage of robotic study it is better to use previously prepared and checked program code, which is necessary to change according to current design needs.

There are a lot of literature and Internet sources with program examples for self-instruction of how to work in a programming environment. Below some code examples are presented to describe how control of periphery devices through DLP-USB245M module in Borland C++ Builder environment could be achieved.

Module start.

```
FT_HANDLE ftHandle;
FT_STATUS ftStatus;

ftStatus = FT_Open(0,&ftHandle); //Open port and set
identifier 0, which means one device in operation

if(ft_Status == FT_OK) {
   } else Caption = "Open Failed";
```

Next step is activating all channels to output data. When program is running function FT_SetBitMode() sets signal lines D0..D7 to output “1” or “0”.

```
FT_SetBitMode(ftHandle,0xFF,1);
```

Now all output lines are activated to output a hexadecimal number FF, which means 11111111 in binary code.

Set of “high” level on D1 output with button 1.

```
Void __fastcall TForm1::Button1Click(TObject *Sender) {
   State = 2; // set variable
   ftStatus = FT_Write(ftHandle, &State, 1,
&lpdwBytesWritten);}
```

Print current state of outputs on the screen.

```
FT_GetBitMode(ftHandle, &BitM);
Label1->Caption = BitM;
```

For convenience control buttons can be presented on screen as arrows indicating the motor turn direction or movement back and forth.

Pressing a mouse button generates sending a decimal number to port, “4” for example. Module output lines generate a binary number “00000100”. As a result “high” voltage level is introduced to third line and it can be used to control motor driver input or some other device.

3.4 Motor Driver L293D

There are a number of different designs to control electric motors. They differ in power and hardware components. Motor driver L293D [4] – is a unique design used by each and every beginning robot technician. It works similarly to a logic element.

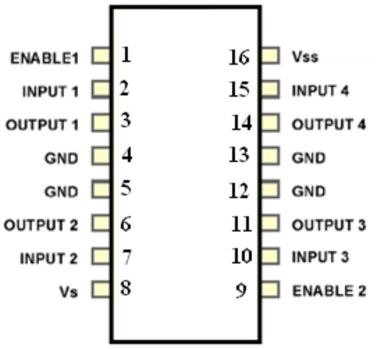


Fig. 6. Motor driver L293D.

During seminars it is important to highlight that motors are controlled bidirectional. Second important point is ENABLE inputs of the driver. For better understanding table below is presented.

Table 1.

ENABLE1	INPUT1	OUTPUT1
1	0	0
1	1	1
0	0	0
0	1	0

3.5 ISIS Proteus

For modeling electronic circuits, there are a lot of popular programs. During this course, ISIS Proteus Design Suite [7] was used. To master it within a short period of time is not possible, but it is important to give a general idea. Children quickly memorized how to draw circuits and how to find familiar components in a library.

With great relish they drew diagrams, putting virtual LEDs and motors into action. The figure below presents motor connection diagram to L293D driver.

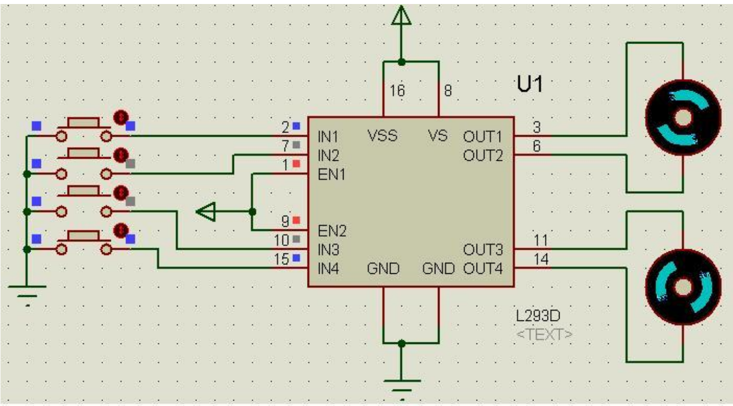


Fig. 7. Motor connection diagram.

4 Results of conducting a startup robotic course for pupils

Author was not limited in choosing topics for seminars or methods of teaching. But wishes of kids in a group and their parents were always taken into consideration. This way was conducted one of the seminars when a number of different relays and motors were disassembled. Soldering lesson was also conducted. Using bread board and thin conductors is rather diligent work and quickly annoys children. As a consequence the most number of circuit designs was realized on computer.

On the last seminar a group was divided into two teams. Each team assembled its robot during the seminar and in the end a competition was held. After the course was over, all participants decided to make a robot for Eurobot Junior. Some of them involved their parents in robot construction and made their own robots at home.

As a result of this first and experimental robotics course among junior school children valuable teaching experience was gained. A portion of this experience was presented in this article. Authors believe to contribute more to the course in future and form an educational framework for teaching basics of robotics according to gained experience. Such framework is to be described in future articles.

During the seminars work of students inspired an idea of a simple robotic construction kit. Several beta-versions of parts of such a kit were realized and used throughout the course. It was decided to develop a simple robotic construction kit for future startup robotic courses, which would allow assembling circuits without soldering, including a bread board, electronic components, motors and control software. The kit will be designed bearing in mind that many parents have little experience in electronics and programming but still wish to practice with their children at home.

D. Sukhotskiy and A. Yudin “Startup robotics course for elementary school” in Research and Education in Robotics - EUROBOT 2010, ser. Communications in Computer and Information Science, vol. 156. International Conference, Rapperswil-Jona, Switzerland, May 27–30, 2010, Revised Selected Papers: Springer Berlin Heidelberg, 2011. pp. 141-148.

References

1. Eurobot, international robotics contest, http://www.eurobot.org
2. Eurobot National Organizing Committee (NOC) of Russia, http://www.eurobot-russia.ru
3. Znatok electronic construction kit, http://www.znatokonline.ru
4. Datasheet for L293D Motor Driver, http://www.st.com/stonline/products/literature/ds/1330.pdf
5. Logic elements at Wikipedia, http://en.wikipedia.org/wiki/Logic_gate
6. UM245R Drivers, program examples, datasheets, http://www.ftdichip.com
7. Proteus Design Software official website, http://www.labcenter.co.uk/products/pcb_overview.cfm