Design of control program for filament extruder

[1] *Peter HRABOVSKÝ,* [2]*Ján MOLNÁR*

[1-2] Department of Theoretical and Industrial Electrical Engineering, FEI TU of Košice, Slovak Republic

[1] peter.hrabovsky@tuke.sk, [2]jan.molnar@tuke.sk

Abstract — **This article is dedicated to designing a control program for a plastic fiber manufacturing machine for a 3D printer using a PLC. At the beginning of this article is describing the current market state and a plastic filament manufacturing process. In the next part is described PLC system and software for its programming. Next a control program and his programming editor are described. Finally, the future concept of HMI interface is visualized in the form of an image.**

Keywords — **control program, filament, HMI, PLC**

I. Introduction

A machine for producing a plastic filament also called filament extruder is a device that converts crushed plastic (pellets) into a plastic fiber. Extruders are the reason you should stop throw up your old or failed prints to waste. They use shredded plastic or pellets, which are much cheaper compared to comparable amounts of the filament. Inventables CEO Zach Kaplan claims that the cost savings can be as much as 90%! That is why this machine is very welcome today. [6]

The principle of plastic fiber production is simple: Plastic pieces push through the heated area, where they are melted, and the liquid plastic is extruded through the nozzle as well as the strand of filament. The idea of having your own production line and being independent of commercially available materials is a great dream for every 3D printer user. On the market, commercial solutions are offered in the form of large production lines designed for bulk production. . Such a production line can be seen in Fig. 1. However, such a large production line is not required for our use. We do not plan serial production of strings and their subsequent sale. The equipment we need for our use will be serve for self-consuming.

Fig. 1 Commercial manufactured line for production of filament for 3D printers [5]

The main part of the extruder is a barrel containing the Archimedean screw. The screw is mounted in the cylinder-shaped housing. One end is connected to the electric motor and the other is at near the nozzle. The electric motor will transport pellets trough barrel towards the heater. The electric motor will

transport material through the barrel towards the heater. Pellets are gravity-fed continuously from a hopper. As the motor is continuously driving the Archimedean screw, the pellets are pushed into the heater. The thermoplastic pellets will soften and melt because of the heat and are then pushed mechanically through a die with a specific diameter and shape. Pushing the soft thermoplastic pellets through the die will cause it to form a continuous filament strand with the diameter of the die (1.75 mm or 3 mm). This process is called extrusion, hence the term "extruder". [3][4]
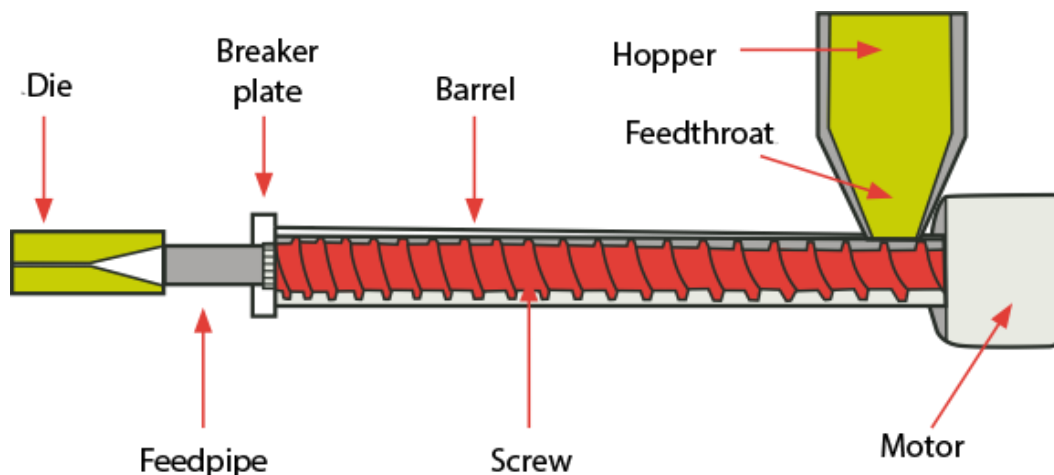

Fig. 2 The process production of plastic filament extrusion [4]

II. PROGRAMMABLE LOGIC CONTROLLER

A programmable logic controller (PLC) or programmable controller is an industrial digital computer which has been ruggedized and adapted to the control of manufacturing processes, such as assembly lines, or robotic devices, or any activity that requires high reliability control and ease of programming and process fault diagnosis. [1][2]

The most used software for programming PLC is SIMATIC STEP 7 from SIEMENS. There are three forms of programming in standard software SIMATIC STEP 7: STL - (Statement List.), FBD - (Function Block Diagram) and LAD (Ladder Logic). These languages, you can switch from one language to another with virtually no restrictions, and choose the most suitable language for a particular block that is currently programmed. [1][2]

STL - Statement List

List command is the principal means for the creation of the program, i.e., this is the lowest level of PLC programming methods, this type of language is supported by all PLCs. Using the list of instructions can use all the instructions that are available in the CPU, the graphic displays that option may be limited. [1] [2]

FBD - Function Block Diagram

This language is used for programming flowcharts. The program consists of the blocks with the respective functions and recalls drawing a scheme of integrated circuits. [1] [2]

LAD - Ladder Logic

It enables programming using schematic diagrams, which show based on the principle of drawing electrical diagrams in liner design. The logic of writing from left to right, so the left side of the input signals, the output signals of the right. [1][2]
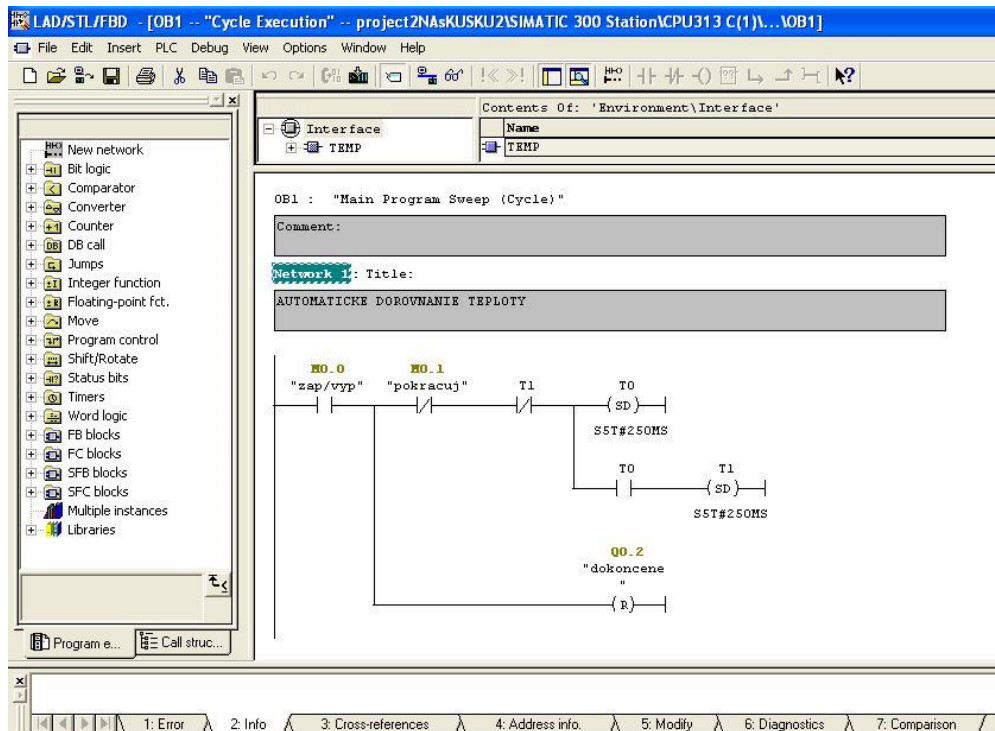
Fig. 3 LAD programming editor

III. PLC CONTROL PROGRAM

This chapter describes the functionality of individual processes running in the PLC system. The device control program is programmed in the SIMATIC STEP 7 - Micro / WIN development environment. The program is written for SIMATIC 300 Station, specifically for CPU313C hardware. Ladder Logic (LAD) diagram was used to create the program. The source code for this management program will not be published in the article. Fig. 4 shows the list of used inputs and outputs addresses in the PLC control program.

	Status	Symbol	Address		Data type	Comment		Status	Symbol	Address		Data type	Comment
1		chyba nizka teplota	M	20.2	BOOL		19		plna cievka	Q	1.2	BOOL	
2		chyba prehriatie	M	20.4	BOOL		20		pocet cievok/vyrobkov	MW	4	WORD	
3		chyba vyprazdnitZas...	M	20.1	BOOL		21		pocet navynutych cie...	QW	7	WORD	
4		chyba vyska teplota	M	20.3	BOOL		22		pokracuj	M	0.1	BOOL	
5		Cycle Execution	OB	1	OB 1		23		prazdny zasobnik	Q	1.3	BOOL	
6		dokoncene	Q	0.2	BOOL		24		priehriata tryska	Q	0.7	BOOL	
7		doplnZasobnik	M	0.6	BOOL		25		reset_hmNavVyrobku	M	8.0	BOOL	
8		emergency stop	M	0.2	BOOL		26		sprava	FC	1	FC 1	nizkaTeplota
9		emergency stop output	Q	0.1	BOOL		27		teplota trysky	QW	3	WORD	
10		hmotnostNavVyrobku	MW	6	WORD		28		uvolnenie vyrobku	Q	9.0	BOOL	
11		mnozstvo plastu v zas.	QW	5	WORD		29		v prevadzke/vypnute	Q	0.0	BOOL	
12		navyjanie	Q	0.3	BOOL		30		vaha cievky/vyrobku	MW	2	WORD	
13		neprechodna tryska	Q	0.6	BOOL		31		vyroba struny/odlieva...	M	0.3	BOOL	
14		nizka teplota	Q	1.1	BOOL		32		vysoka teplota	Q	1.0	BOOL	
15		odlievanie	Q	9.1	BOOL		33		vytlacanie	Q	0.4	BOOL	
16		ohrev	Q	0.5	BOOL		34		zap/vyp	M	0.0	BOOL	
17		pb emergency	M	20.7	BOOL		35		zaseknuty odliatok	Q	9.2	BOOL	
18		pla/abs	M	0.4	BOOL								

Fig. 4 Table of used input, output and memory cell addresses used in the PLC control program

A. Processes in the control program

Automated canned Cycle:

Before setting the bit "*zap/vyp (M0.0 = 1)*", the number of coils or products "*počet cievok/výrobkov (MW4)* " and the coil or product weight "*váha cievky/výrobku (MW2)*" must be selected. The settings you make determine the numbers of process cycles and the length of each process. The weight of the coil or product "*cievky/výrobku (MW2)*" is checked in comparator that compares the "*counter C2*" value and the desired weight. When it is reached, the number of products produced increases and the cycle is repeated again. The number of coils or products "*počet cievok/výrobkov (MW4)* " sets "*counter C1*" to the desired value. After each coil produced, the value stored in "*counter C1*" is decremented. When the "*counter C1*" reaches zero value, the cycle is terminated, production is complete and bit "*dokoncene Q0.2*" is set to 1.

Nozzle heating:

The heating and thermostat function are simulated by a flip-flop circuit which increments or decrements the temperature value *"counter C0"*, the nozzle temperature setpoint is set by selecting PLA (230°C) or ABS (250°C) plastic.

Extruding plastic out of the nozzle:

Extrusion of the plastic from the heated nozzle will be triggered when the correct nozzle temperature, which is determined by the plastic type, is reached. We can see the nozzle temperatures for material PLA and ABS in Table 1.

Turning on winding:

Winding is activated two seconds after the extrusion starts.

Number of Coils Wound:

It is obtained by subtracting the value of the *counter C1* from our set number of coils or products *"počet cievok/výrobkov (MW4)"*. *"Counter C1"* shows how many coils still need to be made to finish the process.

Heating Indication:

Heating is indicated when the nozzle temperature is rising up.

Alarm temperature indication:

Three temperature states are displayed in the form of alarms. The indication is realized by comparators that compare the nozzle temperature *"teplota trysky (QW3)"* with the temperatures defined by us for each state. Individual temperatures are set as shown in Table 1.

Table 1 Indication of nozzle temperatures for materials

Material	Nozzle temperature [°C]			
	Low	**Working**	**High**	**Overheated nozzle**
PLA	< 224	225 – 235	236 – 239	> 240
ABS	< 244	245 – 255	256 – 259	> 260

The temperature is low:

In this state, the extrusion of the plastic is stopped, the heating is switched on to set the desired nozzle temperature. After reaching the desired temperature, the duty cycle is restored.

The temperature is high:

In this state, plastic extrusion is suspended, the extrusion is suspended until it drops to the working temperature for the plastic used. After reaching the desired temperature, the work cycle is resumed.

Overheated nozzle:

This condition threatens to damage the device, so the Emergency Stop mode is triggered.

Working temperature:

The correct working temperature of the nozzle is considered to be the correct operation of the device and is therefore not classified as an alarm.

Plastic hopper:

The amount of plastic in the hopper *"Množstvo plastu v zásobníku (QW5)"* can be set to a maximum of 500 grams. Using the "*counter C3*", we count out the value from the hopper. After reaching $Q1.3 = 0$ *"prázdny zásobník"*, the process will stop. By setting the bit *"doplň Zásobník (M0.6)"* to log1, add value of 100 to the *QW5* value, representing 100 grams of plastic in the hopper. Subsequently, you can reset the paused duty cycle by setting the bit to continue *(M0.1)* to log0.

Emergency stop status:

It is executed with a bit *"emergency stop (M0.2=1)"*. After its setting all processes are finished and the device is switched off.

Early termination:

Set the bit "*zap/vyp (M0.0)*" to log0 to complete the production process.

Production stop:

By setting the bit *"pokračuj (M0.1)"* to log1 to suspend the production process. After resetting the bit to log0, production continues from the state it was stopped.

In Fig. 5 we can see the environment of the S7 PLCSIM simulator. The figure shows the individual windows showing the selected PLC inputs and outputs. We can use our check box windows to manage our device. The individual windows are related to Fig. 4, where the used addresses of inputs, outputs and memory cells are described.
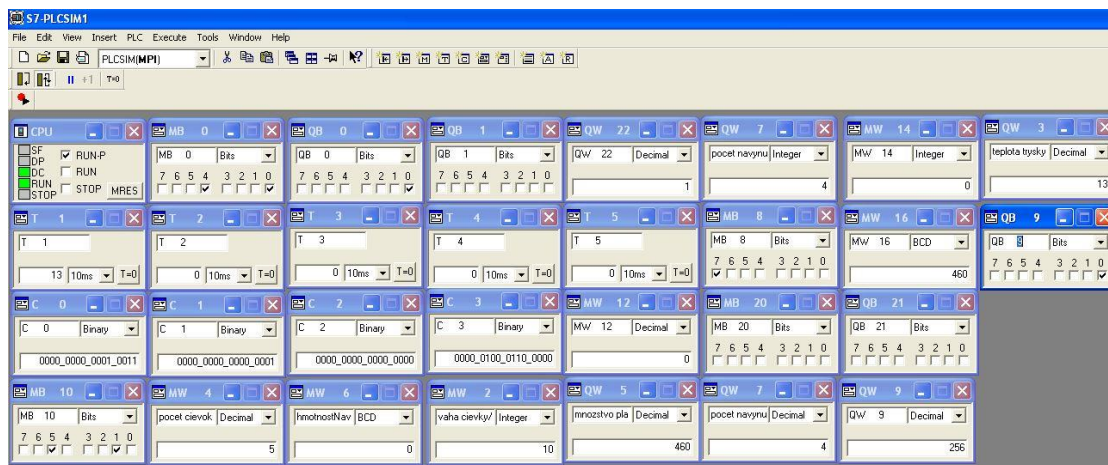

Fig. 5 Environment of S7 PLCSIM simulator

IV. CONCEPT OF HMI

HMI (Human Machine Interface) is the interface between man and machine. HMI panels allow access and control of industrial automation devices that are associated with a control or host computer and HMI software. The HMI, together with the PLC, is an OPLC (Operator Panel and Programmable Logic Controller). In Fig. 6, we can see the assumed concept of HMI interface design for our plastic filament manufacturing equipment. A detailed description of HMI environment creation will be described in the next paper. The HMI environment screen is divided into three main parts.
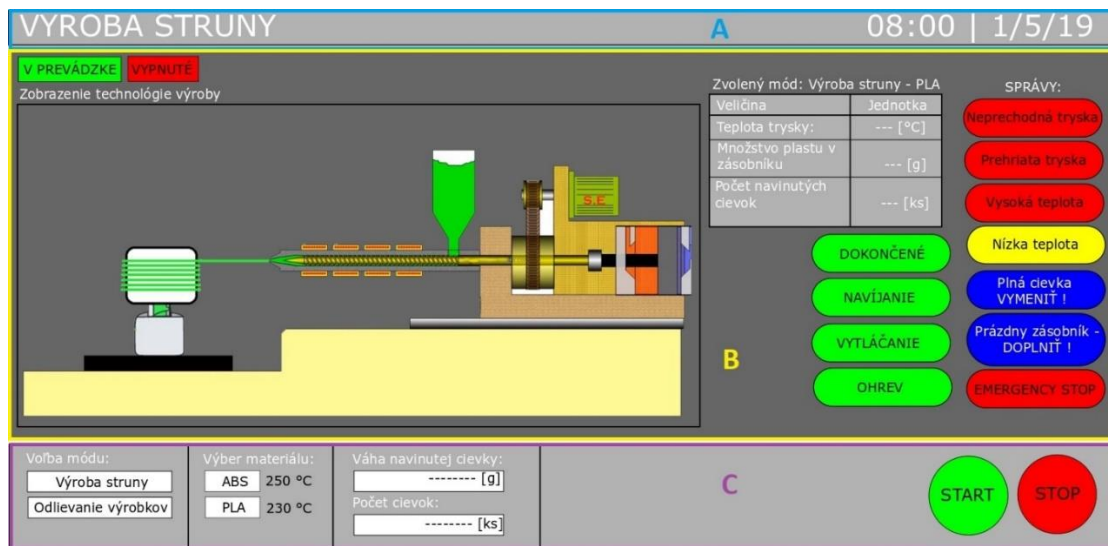

Fig. 6 Concept of HMI screen

A. Overview Area

In this section will be placed process messages, current date and time or logging into the system.

B. Working area

In this section, the technology of production, news, trends, and texts will be displayed on the left. There will be a message box and events on the right. Different pop-up windows located in this area must not overlap important elements. It will be a display of normal and alarm states during the production process.

C. Input Area

The last screen area will include menus, panels, keyboard, and other necessary controls to control the device. Choosing a color for an object or part of an object is governed by generally accepted ISO 9241 - Ergonomic requirements for working with display terminals in the office environment.

V. CONCLUSION

This paper describes the functionality of a control program for a machine for the production of a plastic filament for a 3D printer or also called filament extruder. The control program was written in the SIMATIC STEP 7 development environment from SIEMENS. The HMI concept of this device was presented. In the next article, the design of the HMI environment for this control program will be described in more detail. The hardware part of this device has not yet been realized but will be implemented in the future.

REFERENCES

[1] BOLTON, William. Programmable logic controllers. Newnes, 2015.
[2] SIEMENS.COM.: *SIMATIC Programming with STEP 7 Manual.* Available online at https://cache.industry.siemens.com/dl/files/056/18652056/att_70829/v1/S7prv54_e.pdf
[3] 3DPRINTINGFORBEGINNERS.COM.: *How to make DIY Filament for your 3D Printer*. Available online at <http://3dprintingforbeginners.com/how-to-make-diy-filament-for-your-3d-printer>
[4] WORKSAFE.GOVT.NZ.: *Plastics extrusion presses*. Available online at <https://worksafe.govt.nz/topic-and-industry/machinery/working-safely-with-plastic-production-machinery/plastics-extrusion-presses>
[5] PLASTICPELLETEXTRUDER.COM.: *ABS PLA PETG PA PEEK 3D Printer Filament Extruder*. Available online at <ttp://www.plasticpelletextruder.com/sale-10027039-abs-pla-petg-pa-peek-3d-printer-filament-extruder-0-03mm-tolerance-high-perfomance.html>
[6] ALL3DP.COM.: *6 Best Filament Extruders to Build or Buy*. Available online at https://all3dp.com/2/6-best-filament-extruders-to-build-or-buy
[7] Guzan M., Špaldonová D., Hodulíková A., Tomčíková I., Gladyr A.: Boundary Surface and Load Plane of the Ternary Memory, In: Electromechanical and energy saving systems. Vol. 15, no. 3 (2011), p. 163-167. - ISSN 2072–2052
[8] Bereš M., Perduľak J., Kováč D.: Autonomous mobile robot with obstacles prediction In: SSIEE 2014 : proceeding of scientific and student's works in the field of Industrial Electrical Engineering : volume 3. - Košice : TU, 2014 S. 166-169. - ISBN 978-80-553-1711-4
[9] Kováčová I., Kováč D., Vince T.: Elektromagnetická kompatibilita - 1. vyd - Košice : TU, - 2009. - 137 s. - ISBN 978-80-553-0150-1.
[10] Dziak,J. : Linear circuit simulation using MATLAB and modeling of nonlinear elements, In: SCYR 2014 Proceeding from Comference: 20.5.2014: Herľany, S. 70 - 71, Košice : Technická univerzita v Košiciach, 2014 /978-80-553-1714-4/.
[11] Jacko P., Kováč D.,: Converters and time conversion measurement of STM32F446RE microcontroller , In: Electromechanical and energy systems. Modeling and optimatization methods. - Kremenchuk : Kremenchuk Mykhailo Ostrohradskyi National University, 2017 P. 154-155. - ISSN 2079-5106.
[12] Fecko B., Vince T.: Simulink simulation of DC motor. In: *JIEE : Časopis priemyselnej elektrotechniky : Journal of Industrial Electrical Engineering*. vol. 3, no. 1 (2019), p. 48-51. - ISSN 2454-0900. Available online at <http://www.jiee.eu/wp-content/uploads/papers/vol3i1/v3i1_Fecko_ Vince.pdf>