DOI: 10.35598/mcfpga.2021.012

Development of a Telegram Bot for Receiving Data from OPC Servers

Oleg Khmelik
dept. Design and Operation of Electronic Devices
Kharkiv National University of Radioelectronics
Kharkov, Ukraine
oleh.khmelyk@nure.ua

Pavlo Galkin
ORCID 0000-0002-0558-6448
dept. Design and Operation of Electronic Devices
Kharkiv National University of Radioelectronics
Kharkov, Ukraine
pavlo.halkin@nure.ua

***Abstract*—It is important to develop programming of the embedded control system which allows the real-time system to change and actually program the built-in control system based on the PLC. Chatbot is a computer program developed on the basis of neural networks and machine learning technologies, which can be used to communicate in audio or text format , but chatbot can also be used to program embedded systems without changing the PLC program itself. The purpose of the work is to develop a chatbot in the Telegram service, which can be used to obtain information from embedded systems based on the OPC server and send the necessary information at the request of the user.**

Keywords—OPC, PLC, Telegram bot, Chatbot, embedded control system, data, servers, IoT, IIoT, machine learning technologies.

I. INTRODUCTION

Embedded system - a specialized computer system or computing device designed to perform a limited number of functions, often with real-time constraints. The purpose of the work is to develop a chatbot in the Telegram service, which can be used to obtain information from embedded systems based on the OPC server [1] and send the necessary information [2] at the request of the user. An analysis of platforms and technologies was conducted [3-9]. Among other representatives of chatbots this development is unique, among the Internet in free access there was no similar implementation of the chatbots programmed for this purpose which transferred information from the PLC[3] through the OPC server[4].

Development of telegram-bot for OPC-technology of programming of the embedded control system on the basis of the PLC. Thus, it is advisable to develop a programming system for built-in control system based on PLC.

II. DEVELOPMENT OF THE SCHEME OF DATA EXCHANGE THROUGH THE OPC SERVER

Data exchange of embedded systems based on the OPC server is as follows (fig. 1).

- the configurator itself is configured;
- information from the controllers is sent to the OPC server;
- SCADA-system receives data from controllers by reading data via OPC.
- the telegram bot connects directly to the OPC server and on the command receives data, rewrites necessary variables, controls all process of work.
- he telegram bot sends data to the user, and sends the results of changes.

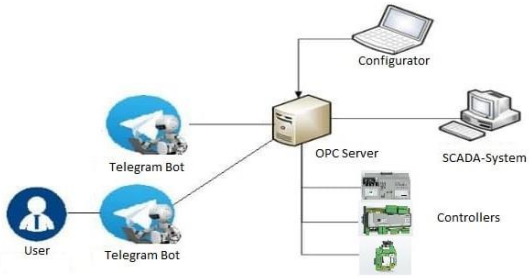

Fig. 1. Scheme of data exchange via OPC server.

The system must perform the task of programming the built-in control system in the following sequence: Bot> server> telegram network> OPCUA server> PLC network.

III. BOT DEVELOPMENT

The first step in programming a chatbot is to obtain its personal token. The personal token is a unique number that the chatbot works on, without it the program will not know through whom requests will be processed. The first step in programming a chatbot is to obtain its personal token. The personal token is a unique number that the chatbot works on, without it the program will not know through whom requests will be processed (Fig. 2).

The token is registered with the command "bot = telebot.TeleBot (" * ")", where "*" is your token. This command sets the token for our telegram bot, without it it will not work.

When we need something from the chatbot, we prescribe a command in the chat and the result is displayed to us. "@ Bot.message_handler (commands = " start '])" with this command we set the necessary condition to start working with the bot, namely, when the user types in the chat "/ start".

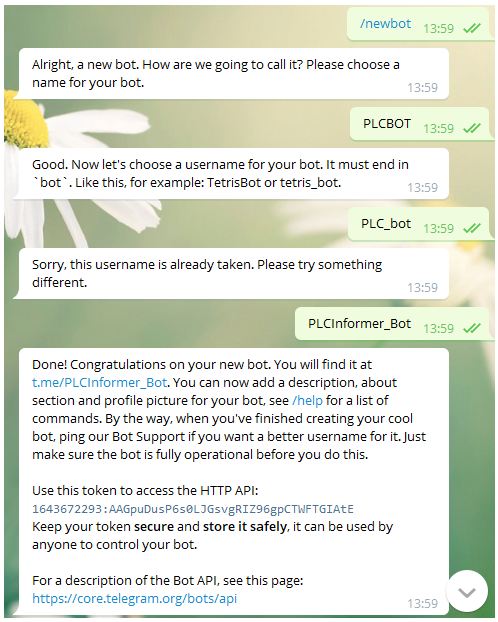

Fig. 2. Example of a conversation with "BotFather".

Next, we specify what will be executed when the user enters the required command (Fig 3).

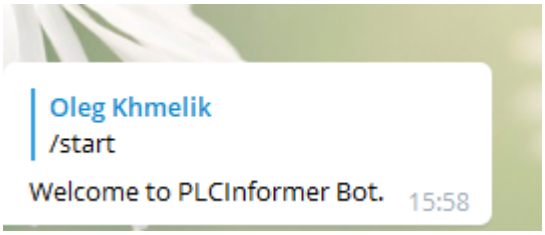

Fig. 3. The reaction of the bot to "/ start".

To read our variables from an embedded system based on OPC, there is a function "opc.properties ('*')", where "*" is the name of our variable, which we learn thanks to the OPC Client utility. To output data from the OPC, we use the binding of the variable to the command. After receiving the data at the end of the function it is necessary to prescribe "opc.close ()" to optimize the use of the connection to the OPC server, without this command after performing the function of connection to the embedded system based on the OPC server does not stop, and re-execution of the command will add more 1 connection without disabling the past. Therefore, it is better to close the communication channel (Fig 4).

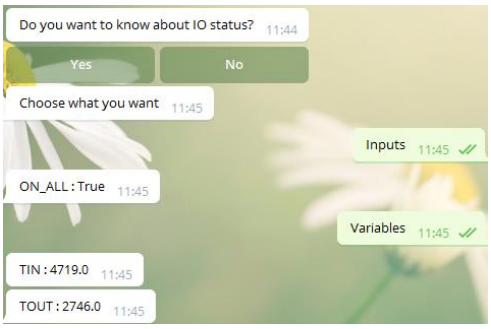

Fig. 4. The result of command processing and information output.

The basis for testing our development of a chatbot for embedded systems based on OPC will be a project in PC WORX with emulation thanks to PC WORX SRT software (Fig. 5).

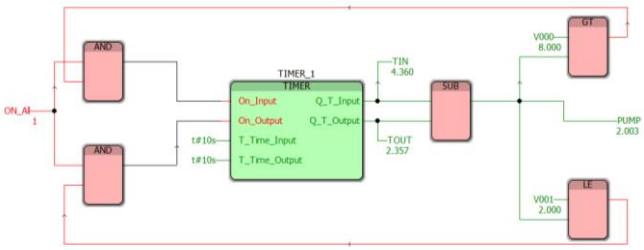

Fig. 5. Emulation thanks to PC WORX SRT in PC WORX

IV. Conclusion

This development Telegram bot is a unique way of user interaction on programming by embedded control systems and exchange of information between them. In the future, the development of this technology can provide remote work for both users and developers of embedded systems for their programming, monitoring and debugging.

References

[1] Lehnhoff S. et al. OPC unified architecture: A service-oriented architecture for smart grids //2012 First International Workshop on Software Engineering Challenges for the Smart Grid (SE-SmartGrids). – IEEE, 2012. – C. 1-7.

[2] Pavlo Galkin, Lydmila Golovkina, Igor Klyuchnyk Analysis of single-board computers for IoT and IIoT solutions in embedded control systems // 2018/10/9 IEEE –C. 297-302.

[3] R. Langmann et al., "Workshop: The TATU Lab & smart education", 2016 13th International Conference on Remote Engineering and Virtual Instrumentation (REV), pp. 400-402, 2016.

[4] P.V. Galkin, V.V. Gavrilenko and A.I. Mon'ko, "Issledovanie dalnosti i skorosti peredachi dannyih po vitoy pare v promyishlennyih setyah RS-485 i PROFIBUS [Investigation of the range and speed of data transmission over twisted pair in industrial networks RS-485 and PROFIBUS]", Problems of Telecommunications, vol. 2, pp. 94-110, 2016.

[5] M. Holikov and P. Galkin, "Analysis of possibilities to use neural network for remote control of electronic devices", Technology audit and production reserves, vol. 2, no. 6, pp. 42-49, 2018.

[6] S.I. Gorb, V.V. Nikolskyi, V.F. Shapo and S.H. Khniunin, "Programming controllers in the integrated development environment: training manual", Practice. – Odessa: National University Odessa Maritime Academy, pp. 164, 2017.

[7] Y. Chuanying, L. He and L. Zhihong, "Implementation of migrations from Class OPC to OPC UA for data acquisition system," 2012 International Conference on System Science and Engineering (ICSSE), 2012, pp. 588-592, doi: 10.1109/ICSSE.2012.6257255.

[8] I. V. Latypov, E. V. Ehlakov, N. Ivanov, E. F. Smirnov and I. Y. Khramov, "News Aggregator from Telegram Channels Using Thematic Text Analysis," 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), 2021, pp. 2150-2153, doi: 10.1109/ElConRus51938.2021.9396536.

[9] P. Galkin, "Analysis models of collection data in wireless sensor networks," 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), 2016, pp. 233-236, doi: 10.1109/INFOCOMMST.2016.7905392.