

ELT011 - Sistemas Distribuídos para Automação – Trabalho – Valor: 30 pts

Professor: Marco Aurélio de Souza Birchall

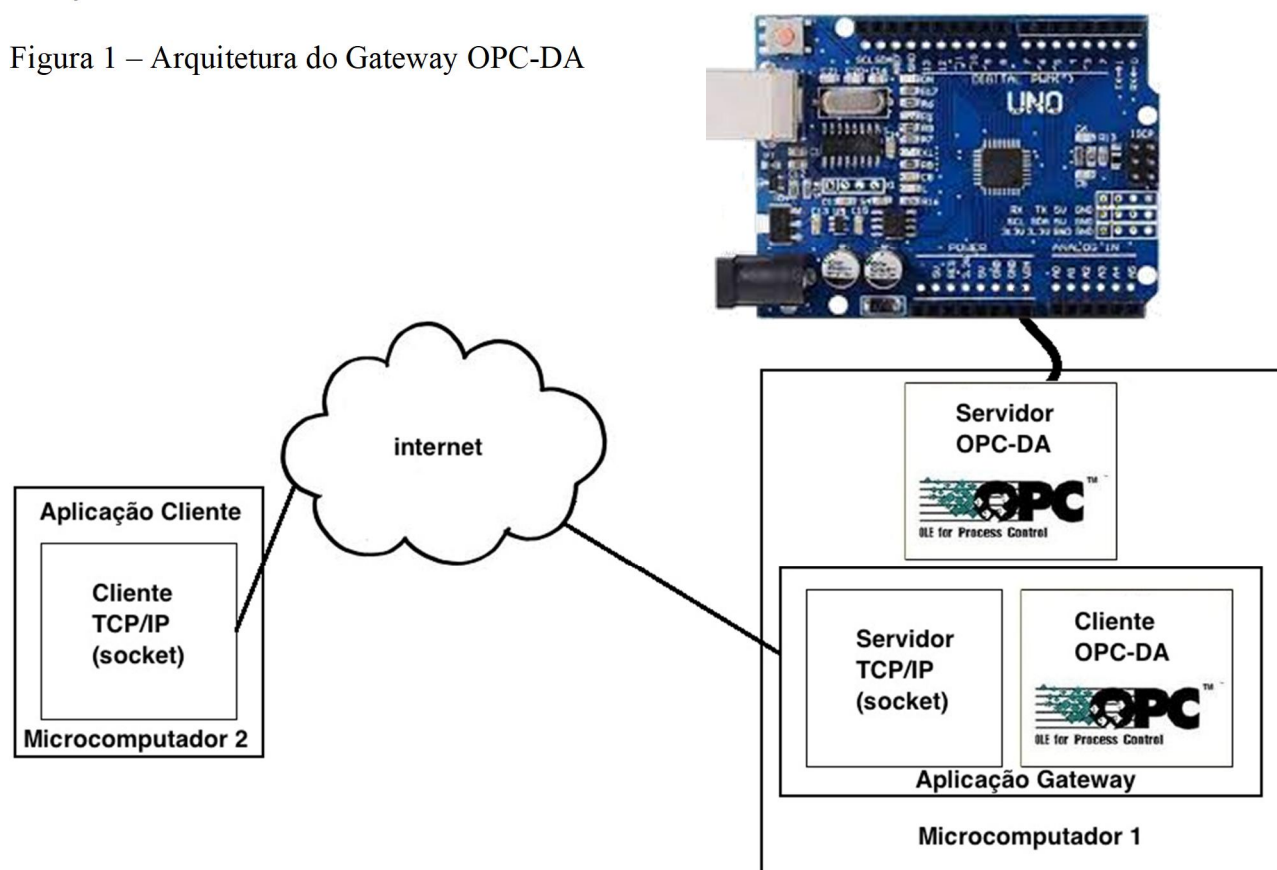
DESCRIÇÃO

Desenvolver uma aplicação que possa controlar, remotamente e via TCP/IP, um servidor Arduino OPC-DA.

Deseja-se que haja um aplicativo Gateway OPC-DA, que também execute no mesmo computador e possa se conectar ao servidor OPC-DA de forma a ser capaz de enviar e receber dados do arduino (ser, portanto, um cliente OPC). Este mesmo aplicativo deve ainda ser capaz de receber conexões TCP/IP através de um *socket*, de forma a que, comunicando-se com um cliente TCP/IP através da internet, receba e envie dados do arduino. O aplicativo Gateway, portanto, também deve operar como um servidor TCP/IP.

Finalmente, deseja-se a implementação de um cliente TCP/IP que seja capaz de se comunicar com o aplicativo Gateway, remotamente, no sentido de enviar e receber dados para o arduino. A figura 1 ilustra a situação descrita.

Figura 1 – Arquitetura do Gateway OPC-DA



O trabalho, consiste, portanto, no desenvolvimento das duas aplicações: o aplicativo Gateway (Cliente OPC-DA e servidor TCP/IP) e da aplicação cliente TCP/IP que acessa o Gateway remotamente, através da internet.



CARACTERÍSTICAS DA APLICAÇÕES

A. Gateway

O aplicativo gateway deve comunicar com o servidor OPC-DA arduino (<https://www.st4makers.com/>) de forma a ler os dados de entrada e escrever os dados de saída. A leitura deverá ser feita periodicamente, através de um evento timer e a escrita, sempre que houver qualquer alteração com relação ao estado anterior.

O gateway deverá ter uma interface mínima, capaz de mostrar os dados de leitura e de obter os dados de escrita.

Do ponto de vista da conexão internet, o gateway deverá operar como um servidor TCP/IP. Assim sendo, deve oferecer os serviços de conexão e de transferência de dados. Uma vez conectado ao cliente, o gateway deverá enviar os valores atuais das variáveis de processo e, sempre que houver atualização das leituras, deverá enviar os dados atualizados. O gateway também deve receber os dados de escrita oriundos do cliente TCP/IP e enviá-los à placa, através e sua conexão OPC-DA.

O cliente OPC-DA do gateway deve ser desenvolvido a partir do protótipo funcional SimpleOPCClient e utilizando-se o Microsoft Visual Studio, modificando-se as partes de código de forma a que se tenha um cliente OPC que envie e receba dados do Arduino OPC server.

B. Cliente TCP/IP

O cliente TCP/IP deverá ser capaz de conectar-se ao servidor (Gateway) através de uma rede TCP/IP e da interface de *sockets*. Uma vez conectado, o cliente deve receber os valores atuais de estado da placa e mostrá-los. Deve ainda haver uma interface mínima de entrada de dados, semelhante àquela da aplicação Gateway, para atualização das variáveis de escrita e leitura do arduino.

NOTAS

1. Instale o Arduino OPC Server de <https://www.st4makers.com/> . Siga as instruções de instalação de <https://www.st4makers.com/category/arduino-opc-server/installation-guides/> para que você tenha o Arduino comunicando com o servidor.
2. SEMPRE QUE VOCÊ FOR UTILIZAR O ARDUINO OPC-DA SERVER, TENHA O ARDUINO FÍSICO CONECTADO À MÁQUINA, PARA EVITAR BUGS DE COMUNICAÇÃO.
3. Instale o servidor e o cliente OPC-DA, MatrikonOPCSimulation, de <http://www.matrikonopc.com>, para viabilizar testes de comunicação OPC mais facilmente. Este software possui um servidor que gera números aleatórios constantemente e um cliente genérico, o MatrikonOPC Explorer, capaz de se conectar a qualquer servidor OPC instalado na máquina.
4. Teste a comunicação OPC-DA entre o MatrikonOPC Explorer e o Arduino OPC Server.
5. Se você tiver problemas para instalar o MatrikonOPC no Windows (erro .NET Framework 3.4 Features cannot be installed on this operating system....) consulte <https://www.brainlatam.com/blog/troubleshooting-installing-net-framework-3-5-335#:~:text=Here%20we%20show%20you%20how,proceed%20with%20the%20software%20installation>.