OCTIV VI Technology - Industrial Communication Protocols

A. Silvestre, D. Gahan and P. Scullin

INTRODUCTION

In this technical note, communication via industrial protocols such as EtherCat and Ethernet/IP are described. The Octiv VI probe is an advanced and versatile radio-frequency (RF) voltage and current sensor. It can be used in a variety of installation environments and has a wide range of applications. It sees widespread deployment on RF processing equipment used in the semiconductor (and related industries) and in the medical device market. For research and development (R&D) activities, the Octiv application software supplied with the product is sufficient for data management. For industrial applications, customers want to process and store data in their own systems and therefore the flexibility to transmit the Octiv data using a range of protocols is critical.

An automated industrial environment brings in requirements of efficiency, safety, and scale that make it very hard to communicate data efficiently through a universal serial bus (USB) or serial communication protocol. Both have strict power and distance limitations.

The industrial Octiv is the first device of its type to address the needs of the industrial customer, in terms of communication standards. It is equipped with one USB, one serial and two RJ45 Ethernet ports, and can be programmed to communicate through any TCP/IP network.

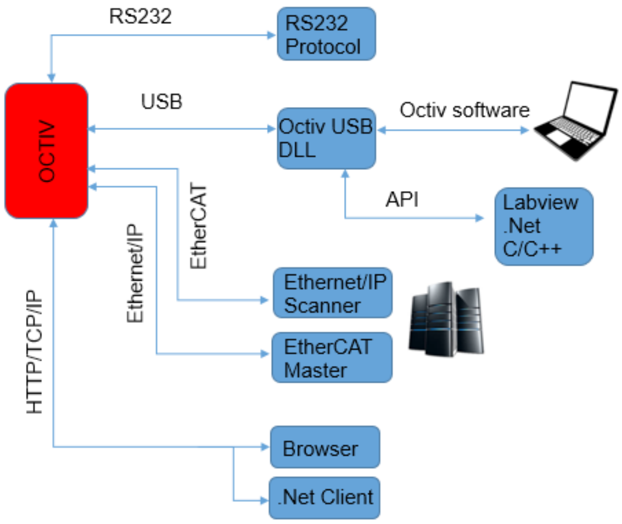

Figure 1. Schematic of the Octiv VI probe digital communication architecture.

Two major industrial communication protocols are available, EtherCat and Ethernet/IP. Data can be transmitted over greater distances using Ethernet architecture (~100 m) compared to USB (3 m) and RS232 (30 m). This makes Octiv a fully enabled internet network node that paves the way for monitoring and control of automated industrial plasma and/or RF processes in real-time to increase efficiency in ways impossible until now.

STANDARD PROTOCOLS

The Octiv sensor is fitted with USB, serial and Ethernet connectors. The Octiv proprietary software runs on a PC and communicates with the Octiv through a dynamic-link library (DLL) via the USB protocol.

An application programming interface (API) is provided to enable communication with the device using a variety of programming languages including LabVIEW and C/C++ as well as visual basic (VB) and C# through the .NET framework. Example code is available for the programming languages mentioned.

As an alternative to USB, an RS232 port is provided to accommodate users who may need to communicate through a serial interface. This may be necessary when interfacing with hardware that pre-dates USB or it may simply be a matter of user preference. Again, the protocol documentation along with sample code for several applications is provided.

REAL-TIME CONSTRAINTS

In computer science "real-time computing" (RTC) has a very specific meaning. Real-time components guarantee a fixed maximum response time to a specific external output event. For example, in a non-real-time computer, there is no guarantee of the speed at which the inputs from the keyboard will be processed, which is why sometimes when the computer is busy with something else, inputs can lag. The Octiv sensor, is capable of real-time communication, with a very high level of reliability and very low jitter (<1 µs for precise synchronization purposes). This is an essential prerequisite for Ethernet-based industrial protocol implementation.

INDUSTRIAL PROTOCOLS

TCP/IP

Undoubtedly, not all applications require that level of efficiency. The industrial Octiv sensor can be connected to any TCP/IP network and data can be reviewed through a simple HTTP/JSON protocol. Once plugged in and correctly configured, the sensor can be left to run for months and data can be retrieved from any computer connected to the same network. Impedans provides code examples on how to integrate data from the sensor, into a variety of major programming languages and environments.

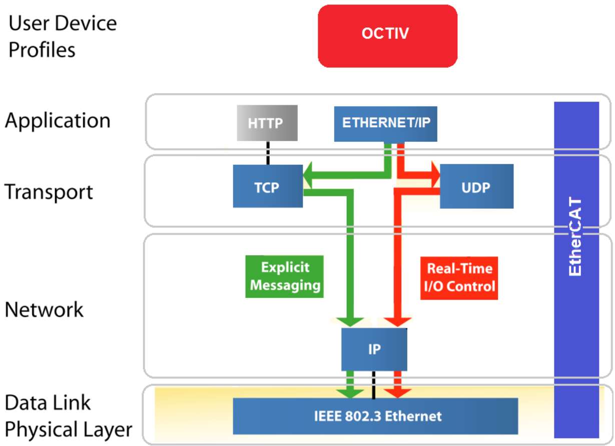

Figure 2. Open System Interconnection (OSI) layers for the Ethernet protocols.

Ethernet/IP

Ethernet/IP is an Industrial protocol that expands on IP communication to add features of reliability and ease of integration into industrial applications. An Ethernet/IP network is composed of Scanners (that open connection and initiate data transfers) and Adapters (provides data to Scanners). It is an Application layer protocol, meaning that Ethernet/IP components can communicate over a regular network without any expensive infrastructure investment (although, only Ethernet/IP components will be able to understand each other on this network).

Transfer of I/O data (i.e. the Octiv sensor measurements) and monitoring of state changes are done continuously through UDP datagrams which are quick and lightweight. Uploading and downloading of parameters is done through TCP (a bit slower but with better reliability).

Ethernet/IP is handled by an international trade and standard development organization whose members are all suppliers of components for industrial automation. It is relatively easy to buy pre-made components or software capable of Ethernet/IP communication.

Ethernet/IP is an implementation over Ethernet of the "Common industrial protocol" a very robust communication protocol that encompasses all industrial automation applications (control, safety, synchronization, motion etc.)..

EtherCAT

EtherCAT is an Ethernet-based fieldbus system, that guarantees real-time responses and with one of the highest levels of reliability in the world. Communication in an EtherCAT network is done from a master to a network of slaves. With EtherCAT, the standard Ethernet packet is no longer received, interpreted and copied as process data at every node. The EtherCAT slave devices read the data addressed to them while the telegram passes through the device, processing data "on the fly". Similarly, input data are inserted while the telegram passes through.

A frame is not completely received before being processed; instead processing starts as soon as possible. Sending is also conducted with a minimum delay of small bit times. Typically, the entire network can be addressed with just one frame.

That means that EtherCAT automated networks can react to a change of situation with a delay set by the frame propagation speed. The only downside is that an EtherCAT network must be dedicated to the protocol and requires a substantial investment in compatible components.