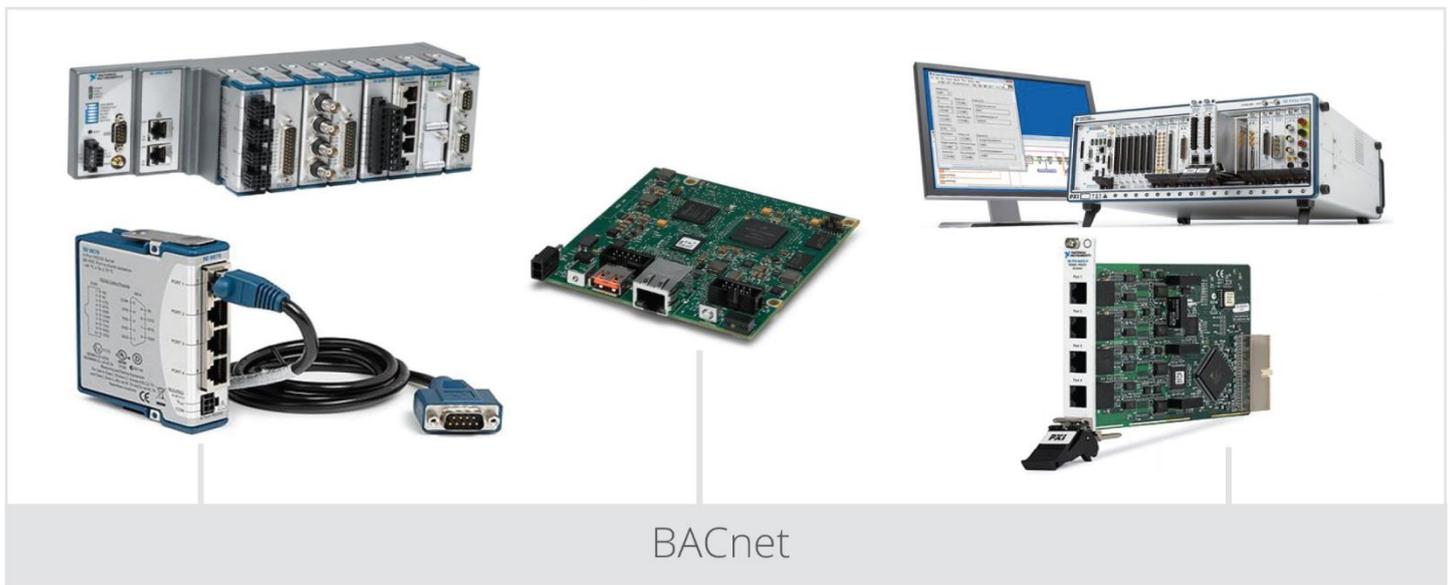


# BACnet MS/TP Protocol for LabVIEW

## User Manual

ASHRAE **BACnet**™ on **NATIONAL INSTRUMENTS**™ Platforms



BACnet

## Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction</b> .....  | <b>3</b>  |
| 1.1      | <b>Definitions and Acronyms</b> .....                                    | <b>3</b>  |
| 1.2      | <b>Purpose</b> .....   | <b>3</b>  |
| 1.3      | <b>Overview</b> .....  | <b>3</b>  |
| <b>2</b> | <b>BACnet Background</b> .....   | <b>4</b>  |
| <b>3</b> | <b>BACnet MS/TP</b> .....  | <b>5</b>  |
| <b>4</b> | <b>BACnet Protocol Implementation Conformance Statement (PICS)</b> ..... | <b>5</b>  |
| <b>5</b> | <b>BACnet Protocol Palette</b> .....                                     | <b>6</b>  |
| <b>6</b> | <b>Example Program</b> .....   | <b>6</b>  |
| 6.1      | <b>Configuring and running the example</b> .....                         | <b>7</b>  |
| <b>7</b> | <b>System Requirements</b> .....   | <b>10</b> |
| <b>8</b> | <b>References</b> .....  | <b>10</b> |
| <b>9</b> | <b>Support Information</b> .....   | <b>10</b> |

## 1 Introduction

### 1.1 Definitions and Acronyms

**AI** – Analog input of hardware.

**Device Instance** – This is the logical address that matters to BACnet. Whether on an MS/TP link or IP network, the device instance is unique across all subnets and routed links.

**Object** – The general reference to sensors, actuators, and other functional elements that make up a BACnet device. The objects fall into categories specified by BACnet protocol. Analog Input object and Analog Output object are a couple of the most commonly used objects.

**Object Property** – Each object has several properties required by BACnet protocol. The most commonly used property is Present Value. Other common properties include reliability and status flags. Optional properties for objects such as analog input include minimum and maximum range, high and low limits, etc.

**UI** – User Interface.

### 1.2 Purpose

This document is designed to provide necessary information to LabVIEW developers of how to use “BACnet MS/TP Protocol for LabVIEW” toolkit for program NI equipment as slave device in the building control systems.

### 1.3 Overview

The BACnet (Building Automation and Control Network) is a communication protocol that defines the services used to communicate between building automation end-devices and building control systems. The BACnet protocol specification like all protocol specification defines both how details represented on the network and the services that are used to move data from one BACnet node to another. There are no proprietary chip sets or special electronics required to implement. Thus, BACnet meets the needs of users, integrators, and equipment vendors.

Historically, building automation and control systems have used proprietary communication networks. In this kind of closed system, building automation equipment supplied from different manufacturers cannot communicate with each other. Building owners and facility managers were forced to rely on products from a single vendor. Modern building automation and control systems provide a variety of building services such as heating, ventilating, and air-conditioning (HVAC), lighting, fire and life safety systems, security, and vertical transportation. There can be significant safety and operational advantages to combine these building services through integrated control networks. Closed network systems provide a major barrier to integrated building facilities with the kind of flexibility and expandability that building owners want. In order to solve these problems, the American Society of Heating, Refrigerating, and Air-2 Conditioning Engineers (ASHRAE) developed BACnet, the only consensus developed communication protocol standard in the world specifically designed to meet the needs of building automation and control networks. BACnet defines a set of standard objects whose properties represent the information that is exchanged between

components of the building automation system and an application layer protocol that is used to access and manipulate this information. It also provides a way to convey the information across a variety of local and wide-area networks that may be interconnected to form an internetwork.

## 2 BACnet Background

BACnet has a layered protocol architecture based on a collapsed version of the Open Systems Interconnection (OSI) Basic Reference Model. Layers 1, 2, 3, and 7 of the OSI model are used as shown in Figure 1. The common object model and application layer protocol can be used with any of four LAN technologies or a point-to-point (PTP) protocol suitable for dial-up telephone communications. BACnet also provides wide-area networking capability (not shown in Figure 1) by using Internet Protocols (IP). The network layer provides a way to interconnect any combination of BACnet networks into an internetwork of arbitrary size and complexity. This allows flexibility in configuring various kinds of network systems, and satisfies real-world requirements of building control systems in terms of speed, throughput, and cost.

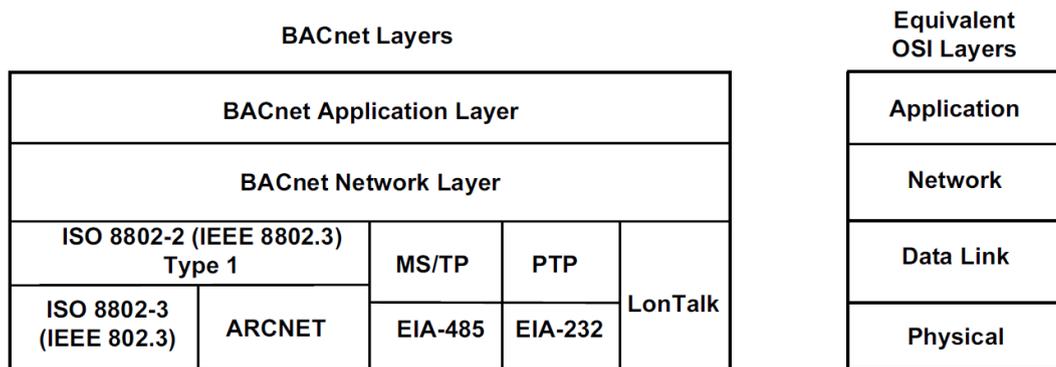


Figure 1. BACnet collapsed protocol architecture.

BACnet devices are defined on the network as a collection of “objects”. Typical objects include Analog Inputs, Analog Outputs, Binary Inputs, Binary Outputs, and more complex objects such as Scheduler. The messages on the network deal with reading and writing Object Properties. The most frequently referenced property for an analog input is “present value”, which typically means data derived from a sensor or physical device. Other properties associated with an analog input object, for example, include fault status, reliability, object name, minimum and maximum limits, etc. BACnet protocol standards define required and optional properties for each object type. The manufacturer’s documentation for the BACnet device will list which object types are included in the device, along with which optional properties are included in each object.

Along with the definition of objects, BACnet protocol defines “services”. These include object access services, alarm and event services, file access services, and a few more. Object access services are the most commonly used since these provide the fundamental “read/write” access to object properties.

Accessing object properties requires specifying these parameters:

- device instance (i.e., which device on the network)
- object type (analog input, binary input, etc.)
- object instance (i.e., which analog input)
- property (present value, object name, status flags, etc.)

Input objects are simple, they just have a “present value” associated with them. Output objects are known as “commandable” and these get a bit more interesting. Because there is the possibility that more than one control algorithm might want control over an object, the command requests are prioritized and only the highest priority command will take effect. When the higher priority command is relinquished, the next lower command will take effect. All of the commands are maintained by the object in its command priority array (containing 16 priority levels).

Writing to an output object requires specifying all of the parameters shown above for accessing object properties, plus two additional parameters:

- command priority
- command level for given priority

The present value obtained when a commandable object is read will be the command level having the highest level that has not been relinquished. To pass command to some other process, you simply write a “relinquish” to the commandable object.

### 3 BACnet MS/TP

MS/TP stands for Master Slave Token Passing. Each device on the link is considered the “master” when it has the token. If it does not have immediate need to use the token, it is required to pass the token along to the next device. This is the “token passing” part. All devices on the link, which do not currently have the token are regarded as slaves, and are expected to listen to any messages the current master may have for it. Because all devices take turns being master, the link is effectively peer-to-peer.

The Master-Slave/Token-Passing (MS/TP) protocol was designed to be implemented using a single-chip microprocessor with a universal asynchronous receiver/transmitter (UART). It uses EIA-485 signaling over a twisted-pair line and is the lowest cost LAN option in BACnet. MS/TP supports transmission rates of 9.6, 19.2, 38.4 and 76.8 Kbps.

With “BACnet MS/TP Protocol for LabVIEW” toolkit the NI hardware can be used as a BACnet MS/T slave device.

### 4 BACnet Protocol Implementation Conformance Statement (PICS)

The "Protocol Implementation Conformance Statement" or "PICS" describes the BACnet capabilities of a particular BACnet implementation. All devices conforming to the BACnet protocol shall have a Protocol Implementation Conformance Statement (PICS) that identifies all of the portions of BACnet that are implemented.

Here is PICS for “BACnet MS/TP Protocol for LabVIEW” toolkit:

|   |                                  |
|---|----------------------------------|
| BACnet Standardized Device Profile (Annex L)              | BACnet Smart Sensor (B-SS)       |
| BACnet Interoperability Building Blocks (BIBBs) (Annex K) | Data Sharing – Read Property – B |
| Standard Object Types Supported                           | Analog Input, Device             |
| Data Link Layer Option                                    | MS/TP slave                      |

## 5 BACnet Protocol Palette

The BACnet MS/TP Protocol for LabVIEW function palette has the following structure.

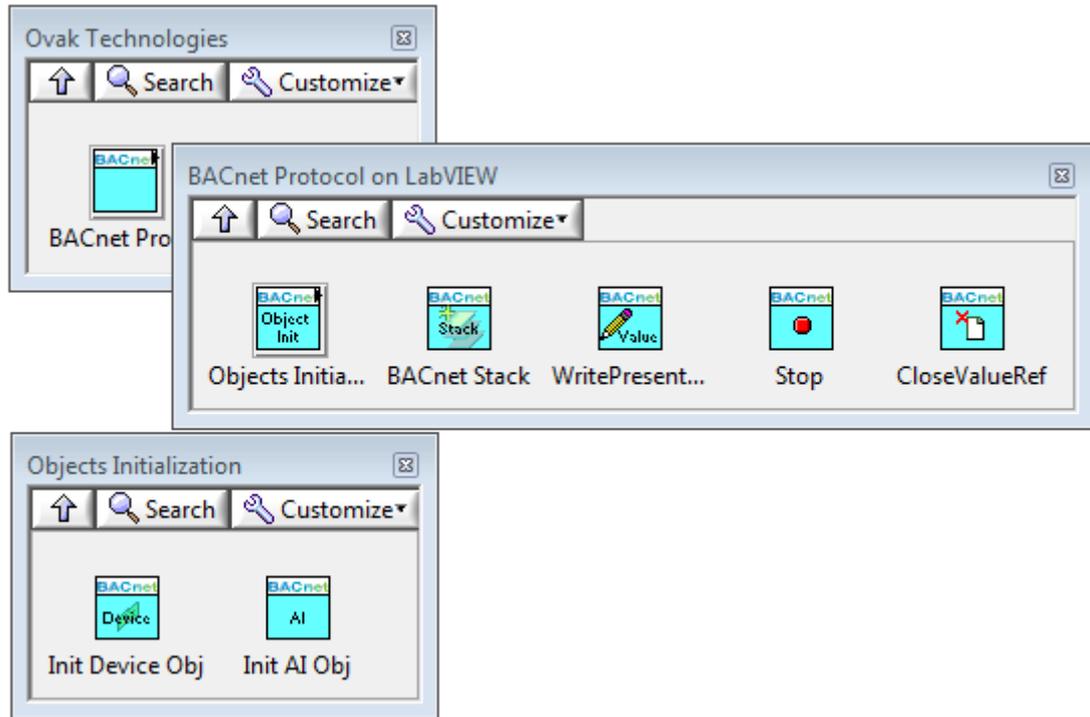


Figure 2. LabVIEW VI function palette BACnet MS/TP Protocol for LabVIEW.

“BACnet MS/TP Protocol for LabVIEW” has the following functions:

- **Init Device Obj** - initiates BACnet Device Object properties and returns reference to this object.
- **Init AI Obj** - initiates BACnet Analog Input Object properties and returns reference to this object.
- **Bacnet Stack** - implements application, network, data link and physical layers of BACnet protocol.
- **WritePresentValueByRef** - writes BACnet Analog Input Object Present\_Value property value.
- **Stop** - used to stop all BACnet stack loops.
- **CloseValueRef** - closes all BACnet Object references.

## 6 Example Program

Below is the BACnet example program that can be found from the NI Example Finder. This example was designed with a sbRIO-9636 as the basic hardware. In order to run this example, you will need to configure a project to run your specific hardware connected to your computer.

This example demonstrates how to use "BACnet MS/TP Protocol for LabVIEW" toolkit for program NI equipment as slave device in the building control systems.

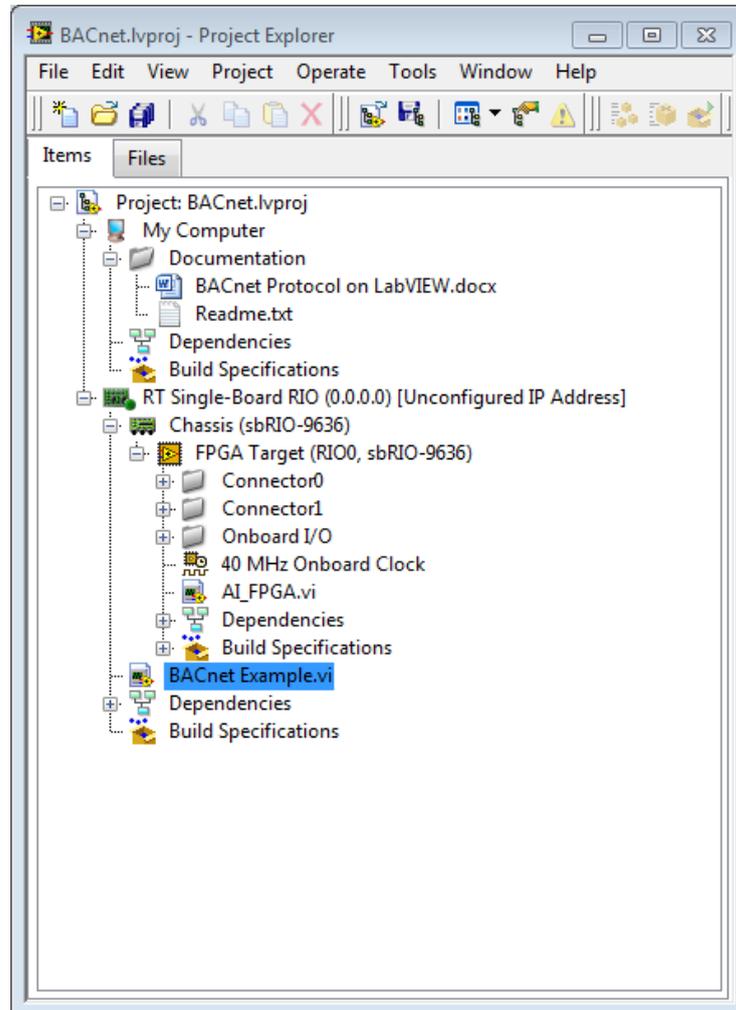


Figure 3. BACnet example project.

### 6.1 Configuring and running the example

The BACnet MS/TP master device is needed for reading AI values of NI device. In this example sbRIO-9636 RS485 port is used to connect with BACnet MS/TP master device. In order to use other NI device it should have RS485 serial interface (port or connected module).

User can choose baud rate and NI device MAC address. The Object identifiers of AI objects also can be changed.

After configuring NI device user can run it.

The BACnet MS/TP master device should be configured with the same baud rate. And after the running it can request the present values of AIs by MAC address and AI object identifiers.

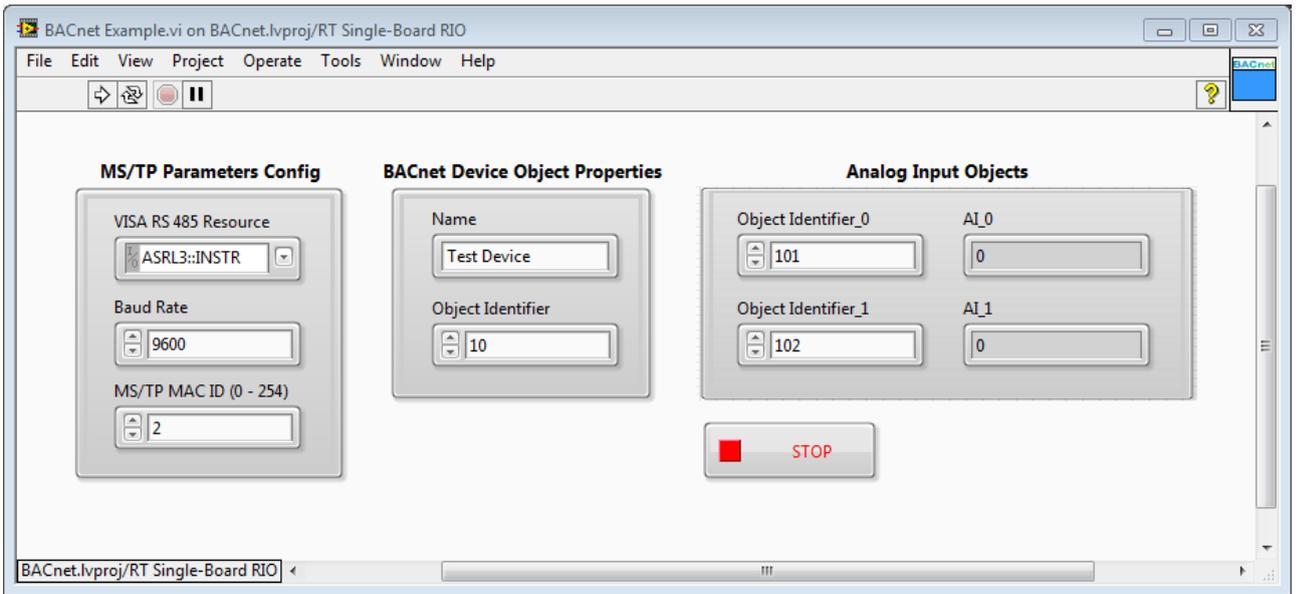


Figure 4. BACnet example Front Panel.

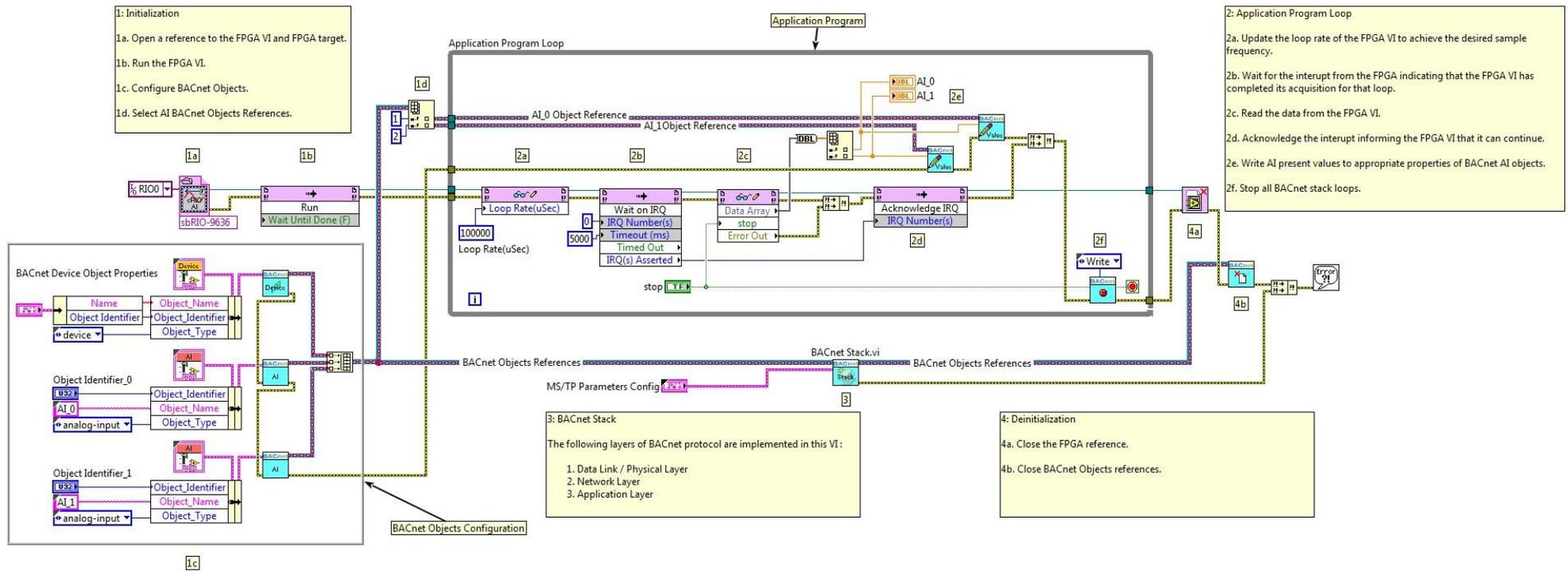


Figure 5. BACnet example Block Diagram.

## 7 System Requirements

LabVIEW Base, Full, or Professional Development System.

## 8 References

1. [www.bacnet.org](http://www.bacnet.org)
2. [www.bacnetinternational.org](http://www.bacnetinternational.org)
3. "Standard 135-2012-- BACnet--A Data Communication Protocol for Building Automation and Control Networks (ANSI Approved)". 2012.

## 9 Support Information

For technical support, please, contact Ovak Technologies at:

**Phone:** + 374 (10) 21-97-68

**Email:** [se@ovaktechnologies.com](mailto:se@ovaktechnologies.com)

**Web:** [www.ovaktechnologies.com](http://www.ovaktechnologies.com)