

# Building a Functional DeviceNet Network

**D**eviceNet is gathering strength as an international defacto-standard device-level network. There are many sources of information about various aspects of implementing a network, including the specification itself, training courses and technical bulletins from several vendors. This article is a basic tutorial, discussing all aspects of designing and implementing a control system based on DeviceNet including; control architecture, communication strategies, network wiring, network power configuration, grounding, testing and fault diagnosis.

## What is DeviceNet?

DeviceNet is a real-time, device-level architecture that defines network protocols for time-critical I/O and messaging data. It also defines a standard device software model that allows for multi-vendor interoperability and interchangeability of like devices.

**Flexibility:** The DeviceNet specification has a wide range of features, but does not limit a vendor's flexibility by requiring all features in all devices. This flexibility within the specification provides for a variety of products ranging from limited capability, low cost devices to more versatile, higher cost devices.

The process of selecting which DeviceNet features are needed, and then selecting the components that provide those features is a design step that is often missed by many first-time DeviceNet users.

## Step 1: Selecting a Control Architecture

DeviceNet supports a variety of control architectures giving the control designer the flexibility to chose centralized control (a single PLC or PC-based controller), semi-distributed control (with multiple PLCs, PC-based controllers and/or smart devices) or highly distributed control (with smart devices only and no identifiable centralized control element).

Selecting a control architecture is not necessarily obvious as the first step in designing a DeviceNet based system. In the past, selecting a network protocol also restricted or eliminated the user's choice of control architectures.

## Step 2: Choosing a Communication Strategy

As with control architecture, the selection of a communication strategy has not been a typical step in system design. Many network protocols have few communication mechanisms available, limiting or eliminating choice.

DeviceNet provides a good selection of communication mechanisms that fulfill a variety of needs and may be implemented individually or all together in the same system.

**Strobe/Poll I/O:** These mechanisms are typically simple to configure and are ideally suited to systems where many I/O points change state rapidly, but generally make inefficient use of network bandwidth due to the need for frequent updates to check for changed inputs.

**Cyclic I/O:** Cyclic I/O uses an unsolicited update mechanism that reduces the input and output update rate of each station to its ideal minimum. Cyclic I/O requires more effort to configure but can provide better system performance and more efficient use of bandwidth when the minimum update rate for individual I/O devices can be identified.

**Change-of-State I/O:** Like Cyclic I/O, Change-of-state (COS) I/O uses an unsolicited update mechanism. COS I/O provides the efficiency of individually configured minimum device input and output update rates, but also provides drastically reduced I/O latency by sending immediate I/O updates when the data changes.

On the downside, COS I/O requires much more knowledge of the system being controlled to determine the impact of COS messages on network bandwidth. Commissioning a system with COS I/O adds bandwidth analysis to the normal list of tasks that should be performed. Systems with COS I/O require a bit more effort to design and test, but the payoff in system performance is worth it.

**Explicit Messaging:** If your control system has some parameters that are accessed infrequently, such as limits and thresholds, explicit messaging may be appropriate. These messages are generated only when the control logic triggers them and therefore have the minimum impact on available network bandwidth.

## Step 3: Selecting Devices

DeviceNet is open. DeviceNet is flexible. These two facts combined result in a large variety of devices that the control designer must choose from. Many factors influence the selection of particular devices, but there is a checklist of some requirements specific to DeviceNet that must be addressed in order to ensure a system will work:

### General Checklist for all Devices (Controllers & I/O)

- Does the device support the required network baud rate?

*DeviceNet does not require devices to support all 3 baud rates.*

- Does the device support the required communication strategy?

*Many devices support a subset of the available communication types.*

- Has the device passed the DeviceNet conformance test?

*The DeviceNet conformance test provides a significant level of confidence that the device functions as advertised by actually testing the device's operation using all supported features.*

### Server (I/O) Device Checklist

- Does the device conform to a standard device profile?

*Selecting devices that conform to a standard device profile provides interchangeability with other devices that conform to the same profile.*

### Controller Checklist

- Does the controller provide sufficient performance?

*Controller performance is a combination of I/O update interval, and control logic performance. The question to ask is: "How long does it take for this controller to react to an input and update an output".*

## Step 4: Designing the Network Wiring

The network wiring design has an impact on the maximum allowable baud rate and is an input to the network power system design. Careful planning of the network wiring is necessary since the total trunk length and the cumulative drop length (the total length of all drop lines) must be controlled. Some important points to remember:

- Network trunk lines may be constructed with thick, thin or combinations of thick and thin cable according to the tables in the DeviceNet Specification.
- The trunk line must be terminated at each end. That means two terminators. No more and no less.  
*DeviceNet differs from many other networks in that it requires network termination for proper operation, regardless of cable length.*
- The trunkline-dropline topology guidelines must be followed faithfully.  
*Bending the rules will usually cause more problems than it solves.*
- Building trunks with thin cable has a significant impact on network power design.

## Step 5: Configuring Network Power

Network power is a new concept to most control systems designers. DeviceNet supplies 24Vdc in the network cable to power simple devices, or just the isolated network interface in larger devices. The power system design goal is to deliver a minimum of 11Vdc at each device while limiting common mode voltage to less than 5Vdc.

The design process involves placing one or more power supplies on the network to ensure that the voltage drop in the cable between a power supply and each station it supplies does not exceed 5Vdc and the current does not exceed the cable/connector limit. The DeviceNet Specification provides an easy table lookup method for power system design. Some important points to remember:

- Don't forget to use the maximum inrush current specification for each device when designing the power system, or you may experience startup problems.
- Use multiple power supplies with care. You must guarantee that the power supply common (V-) does not vary by more than 5V between any two points in the network.

## Step 6: Grounding

While network grounding could have been covered under step 4 or 5, it is both sufficiently important, and (often) so poorly done as to deserve its own section. Network grounding is achieved by connecting the DC power supply common (V-) wire and the shield to a low-impedance ground at the power supply. If multiple power supplies are used, the ground connection must be made at only one power supply, preferably the power supply nearest the center of the network. All splices and taps in the network must connect the shield as well as the signal and power lines.

## Step 7: Testing

After the network is installed, there are several tests you can perform to diagnose common problems before communication faults occur. All of the following tests are performed with all devices installed and all power supplies turned on.

*Do not perform these tests while the system is operating (no communication)!*

**Network Termination & Communication Wires:** Check the resistance from CANH to CANL at each device. It should be between  $50\Omega$  and  $66\Omega$ . If the value is greater than  $66\Omega$  there could be a break in one of the signal wires or missing network terminator(s). If the value is less than  $50\Omega$  there may be a short between the network wires, extra terminating resistor(s), faulty node transceiver(s) or unpowered nodes (unpowered node transceivers have an unacceptably low input impedance).

**Grounding:** Break the shield at a few points in the network and insert a DC ammeter. There should be zero current flow in the shield. If there is current flow, the shield is connected to DC common or ground in more than one place (possibly within a device).

Connect an ammeter (one that can handle the maximum power supply output current) from DC common to the shield at the opposite end of the network from the power supply (or both ends if the power supply is centrally located). There should be significant current flow. If there is no current, check for breaks in the shield at each connector, tap and junction, also check that the shield is connected to DC common (V-) at the power supply. This test can also be performed at the end of each drop if practical.

**Network Power:** Measure the power supply voltage at each device. It should be 11Vdc or higher. If it is not, check for faulty or loose connectors and verify power system design calculations by measuring current flow in each section of cable with an ammeter.

Measure and record the voltage between the shield and DC common at each device. This voltage should be between -5Vdc and 5Vdc at each device and the difference between the highest and lowest measurements should be less than 5Vdc. If the values are out of range check power system design, verify current flows in each section of cable and check the integrity of the shield.

**Station Addresses & Baud Rate Settings:** The Network Status LED included in many products is an excellent diagnostic tool for this purpose. The LED should be flashing green on all devices. Solid RED indicates a communication fault (possibly incorrect baud rate) or a duplicate MAC ID (station address). Connecting the non-functioning devices one-on-one to a network configuration tool is an easy way to diagnose the problem.

Using a network configuration tool to perform a “network who” verifies that all stations are connected and capable of communicating.

## Step 8: Diagnosing Faults

DeviceNet is subject to the same types of problems as any other network; faulty devices, opens and shorts in the network wiring, electrical interference, signal distortion due to incorrect termination or failure to adhere to topology guidelines and signal attenuation due to faulty connectors and loose terminal blocks.

There are some additional problems that are more specific to DeviceNet; excessive common mode voltage, low power supply voltage and excessive signal propagation delay.

**Low-tech Approach:** Many problems can be identified using the low-tech approach of disconnecting parts of the network and watching where the fault goes. Another extremely successful diagnostic method for previously functioning networks is simply to ask the question “What has changed?”.

The low-tech approach does not work for other problems such as excessive common mode voltage, ground loops, electrical interference and signal distortion because disconnecting part of the network frequently solves the problem. The test procedures in step 7 can be helpful in diagnosing these types of faults.

**Using an Oscilloscope:** Looking at the network signal with an oscilloscope can be misleading since many perfectly good differential signals look perfectly awful when viewed individually. Also consider that most network problems you are tempted to diagnose with an oscilloscope are intermittent. Unless you are able to trigger your scope on the bad signal, you will probably spend your time looking at good signals.

Viewing network signals with an oscilloscope is seldom fruitful except when done by an experienced diagnostician with access to supplementary tools to trigger the scope on specific network occurrences.

**Your Brain is Your Best Diagnostic Tool:** Be a detective. Write down the symptoms including as much detail as possible. Ask yourself things like “Do intermittent problems occur when other un-related equipment is in use?” Look for patterns in the symptoms: do some nodes communicate correctly? What do they have in common? What is the difference between the functioning nodes and the others? Consider factors such as proximity to the power supply, proximity to the terminator and proximity to the scanner. Even if you cannot locate the problem yourself, your notes are invaluable when you enlist assistance.