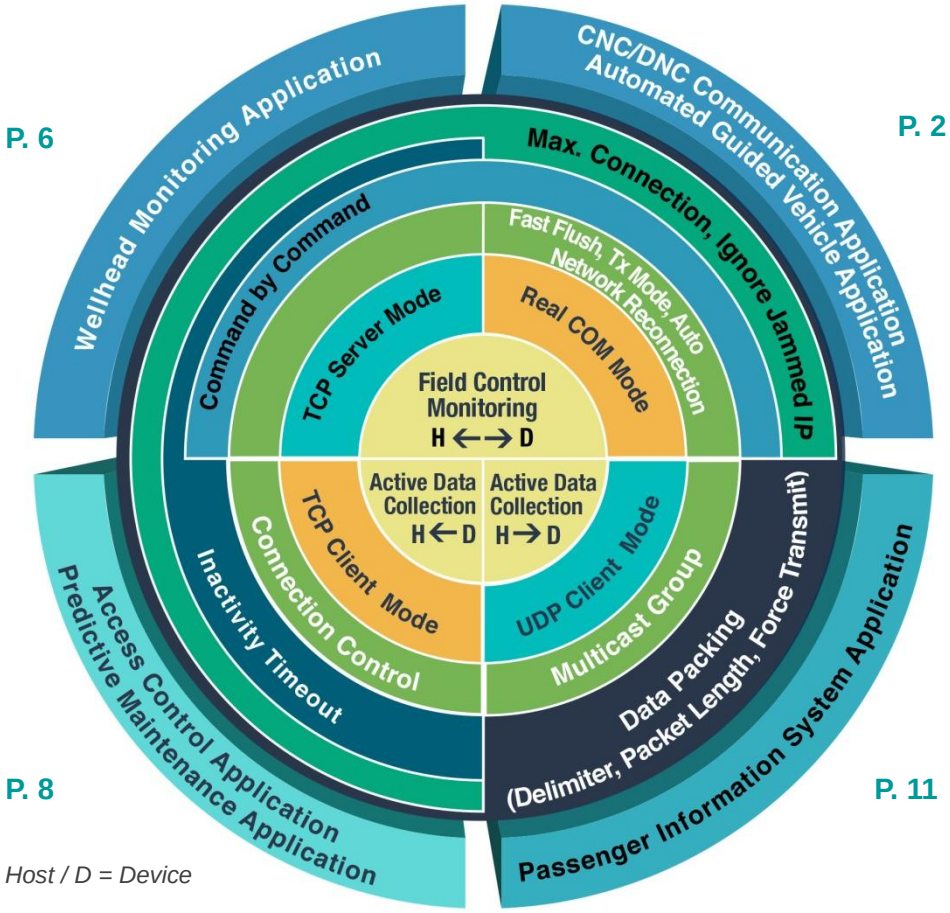
Make the Most Out of Your NPort Serial Device Server by Properly Matching Operation Modes with Advanced Functions

Preface

Moxa has been a leader in device connectivity for more than 20 years, and provides a wide selection of industrial-grade device connectivity products that are used in a broad spectrum of industrial applications around the world. With growing interest in the Industrial Internet of Things (IIoT), the NPort serial-to-Ethernet solution is one of the first and essential steps to enabling IP connectivity for all your legacy devices or serial “things.” Many people are already aware that NPort serial device servers provide plenty of operation modes to help different types of serial devices be seen on the network. What they might not already know is that within each operation mode, the NPort is also equipped with various advanced functions to assist users in streamlining operations and maximizing benefits of serial-to-Ethernet connectivity.

The following NPort Flavor Wheel introduces how to use the 4 most popular NPort operation modes to current users of serial-to-Ethernet solutions and new users who are seeking to add networking capability to their edge devices in preparation for the Industrial IoT era. In particular, the NPort Flavor Wheel illustrates how NPort’s 4 most commonly used operation modes are being adopted in real-life applications, and how matching proper advanced functions with these operation modes can better assist users in achieving their objectives.

In the core of the Wheel, three different data behaviors describe possible application scenarios. The scenarios correspond to suitable operation modes and advanced functions to help users find a better way to reach their goals. Lastly, the rim of the Wheel provides examples of how these combinations help NPort users in real-life applications!



Note: H = Host / D = Device

NPort Operation Mode Guidance

People need serial-to-Ethernet solutions in order for their serial devices to enjoy all the benefits of an Ethernet network. These benefits include the ability to centralize business operations, the ability to communicate over long distances, and the capability of extending the lifecycle of their serial assets. However, many people find that migrating legacy serial networks to Ethernet is actually more challenging than they initially anticipated.

Scenario 1: Field Control Monitoring (H ⟷ D)

Challenges

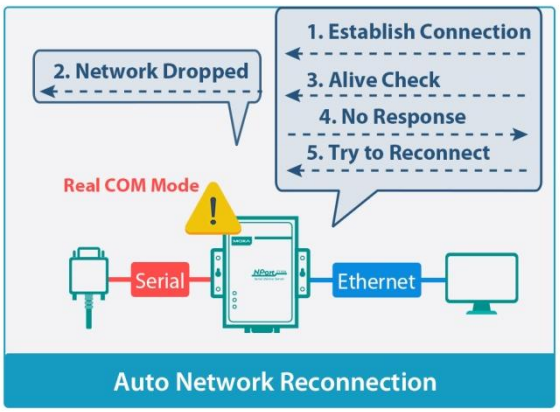
- The user would like to be able to remotely control and monitor serial devices (e.g., CNC machines) through Ethernet.
- The user lacks sufficient budget or resources to re-develop the application software to adopt new network protocols; everything is written for serial applications.
- The user needs to ensure that network communications stay connected.
- The user does not want the read/write performance to be affected by adopting serial-to-Ethernet solutions.

How NPort Can Help

Real COM Mode | **Auto Network Re-connection + TX Mode + Fast Flush**

In industrial applications, legacy devices often only work with legacy software on an outdated operating system, such as Windows CE. What if the user does not have sufficient budget or resources to rewrite the application software with this new serial-to-Ethernet solution? That's what Real COM mode is for. Moxa NPort device servers come with what we call a "Real COM" driver, which creates a virtual COM port on the host computer to map the IP port on the network and allow serial devices at the edge to communicate with the host computer as if they were physically connected by serial cables without modifying any serial software application.

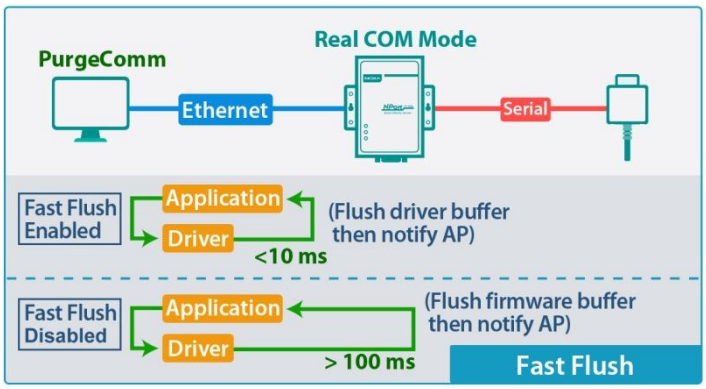
Nonetheless, serial and Ethernet communications remain two distinct methods of data transmission so there may be concerns over the stability of TCP/IP networks and read/write performance when migrating from serial to Ethernet networks. That is why the Real COM driver provides a rich set of advanced functions to ensure that the virtual COM port performs just like a native COM port.



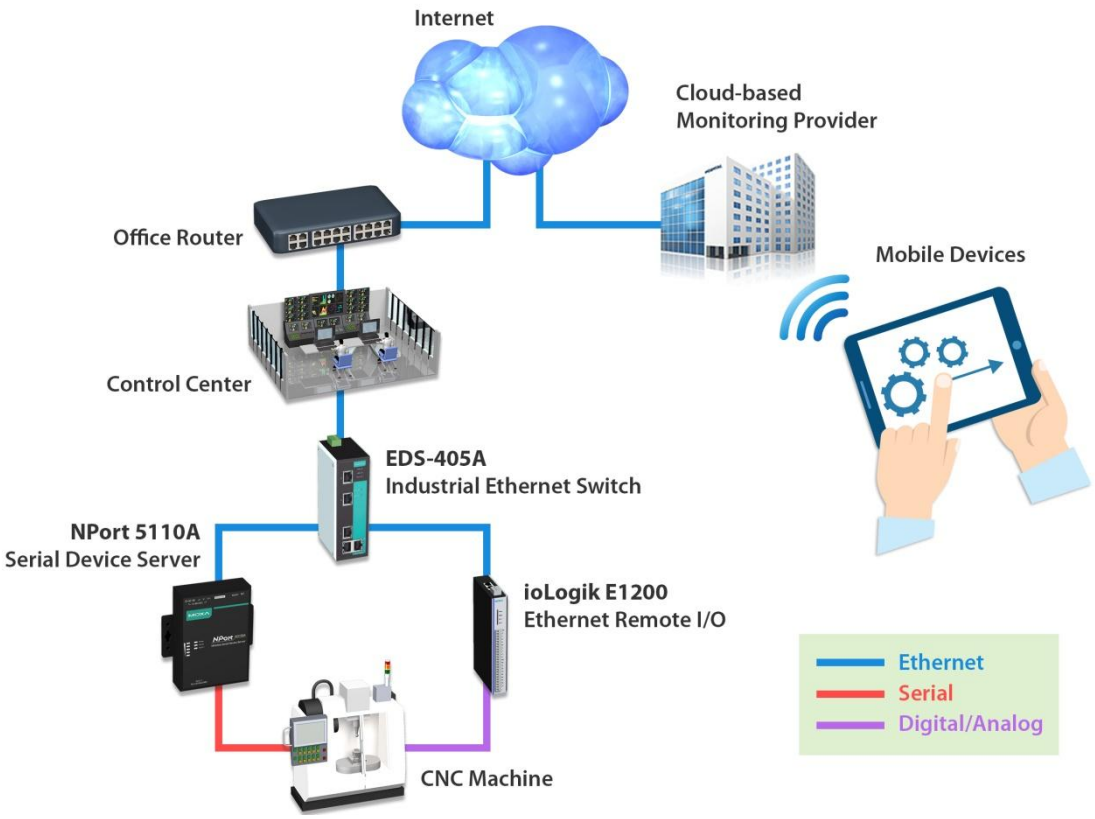
When the TCP connection is dropped (or timed out), i.e., when the NPort does not respond to background "alive

check" packets, the Real COM driver will repeatedly attempt to re-establish the TCP connection if the user enables the **Auto Network Reconnection function**. The user's software does not need to close the port and re-open it. In most cases, user applications cannot afford long latency times. To achieve the same operation performance as a native COM port, Real COM driver provides **Tx Mode** and **Fast Flush** to enhance data performance. There are two options under Tx Mode: **Hi-Performance** and **Classical**. In **Hi-Performance** Mode, the NPort driver will notify the application program to return when the data has been sent out from the **driver**. It works more quickly and is suitable for applications that require better performance and throughput.

Fast Flush, another useful function built into the Real COM driver, works with the PurgeComm function (IOCTL_SERIAL_PURGE) to directly flush the local buffer rather than querying the NPort firmware continuously. If the function is disabled, it would take more time (about several hundred milliseconds) compared to native COM ports.



Application Sharing: CNC/DNC Communication



Scenario 2: Field Control Monitoring (H ⟷ D)

Challenges

The user's application program has a specific timeout limit to receive and process data. If the serial device cannot reply to a request from the application in time, the application will continuously send out the request, which will result in very inefficient overall communication.

How NPort Can Help

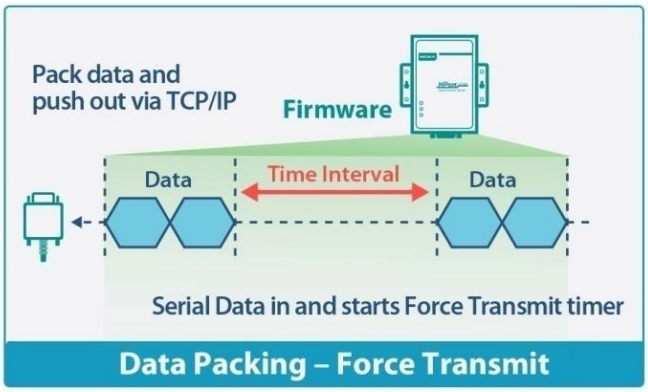
Real COM Mode | Data Packing

Real COM mode, or in fact, every NPort operation mode supports the **Data Packing** function. Data Packing can be used to customize the packet size of your serial data for transmission by packing the data according to the packet length or according to a special character or delimiter in the packets' header or trailer. If the packet length is unknown, or you still want to send incomplete data packets, the Force Transmit function allows you to preset a maximum time interval to wait for a complete serial data stream.

The following application presents Automated Guided Vehicles (AGV) used in a manufacturing facility or warehouse to automate lifting tasks to increase efficiency and reduce labor costs. One of the methods to guide AGVs is to use magnetic tape for the guide path that the AGV follows with its built-in guide sensor (e.g., RFID). The application software will constantly send a location request to the AGV to verify its current location every 50 ms until the response is obtained, and then provide task commands after the location is determined.

To make sure that the location response is sent back within the proper time interval to prevent repeated requests from being sent over and over again, one type of **Data Packing** function can be used—the **Force Transmit** function.

Force Transmit, which is defined using the **Interval Timeout** between characters, forces the NPort's firmware to pack serial data into the same data frame during the specified time. The NPort will transfer data stored in the buffer via TCP/IP only if: 1) the internal buffer is full, or 2) the inter-character interval times out.



NPort Operation Mode Guidance

Application Sharing: Automated Guided Vehicle

Scenario 3: Field Control Monitoring (H ⟷ D)

Challenges

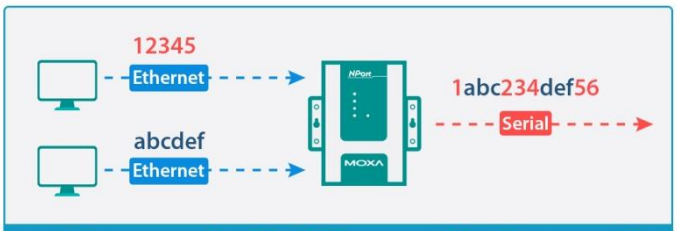
- Multiple hosts must be able to send requests to edge devices to obtain field data but also avoid data collisions if requests are sent simultaneously.
- For field monitoring and control applications where edge devices are usually located far away, users often need to be present on-site to restart the device if network problems occur at the host end.

How NPort Can Help

TCP Server Mode | Max. Connection + Command + TCP Alive Check Timeout

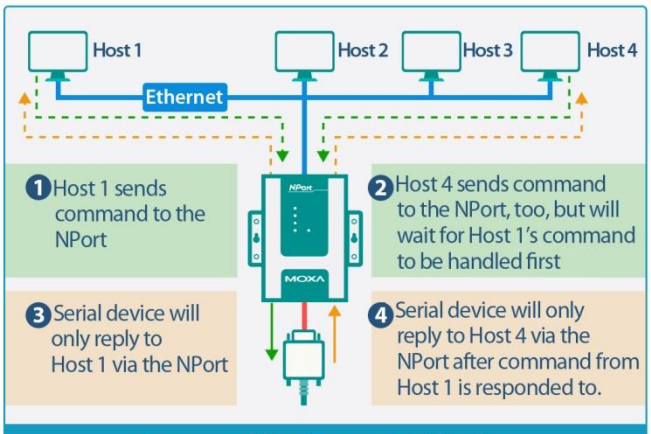
In many industrial applications today, such as oil and gas fields, the concept of real-time monitoring has been widely adopted. The benefits this brings include improved production efficiency (performance and cost), as well as significant reduction in downtime.

The essential factors that influence the productivity of oil and gas, as well as the safety of field operations, involve proper management and monitoring of flow rate, pressure and temperature, etc. NPort's **TCP server** mode is often used in such field monitoring and control applications to connect with field sensors, such as pressure gauges, flow meters, and other types of devices. Hosts running TCP client programs initiate contact with the NPort, establish the connection, and receive data from serial devices. TCP server mode also supports up to 8 simultaneous connections (the NPort 5000 series supports up to 4 hosts while the NPort 6000 series supports up to 8), enabling multiple hosts to collect data from the same serial device(s) at the same time. When the **Max. Connection** function is set, **Command by Command** mode is recommended to prevent data collisions if different hosts are allowed to send requests to edge device(s) at the same time.



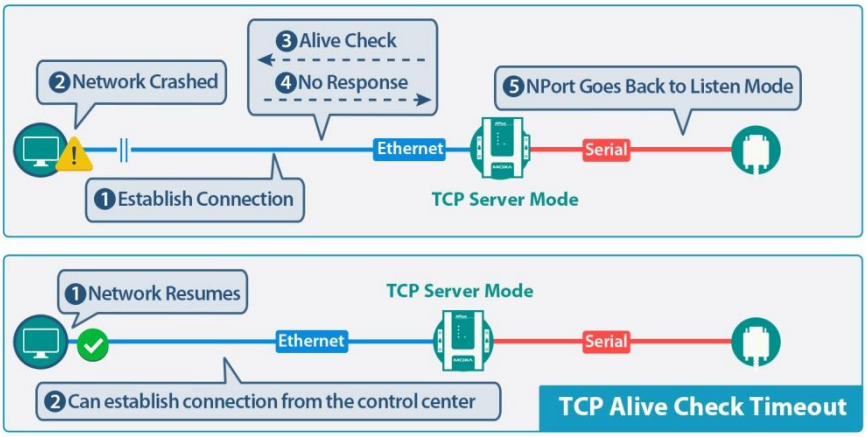
Data out of order if Command by Command Mode not Enabled under Max. Connection

If **Command by Command** mode is enabled and the NPort receives a command from any host on the Ethernet, the NPort will store the command in



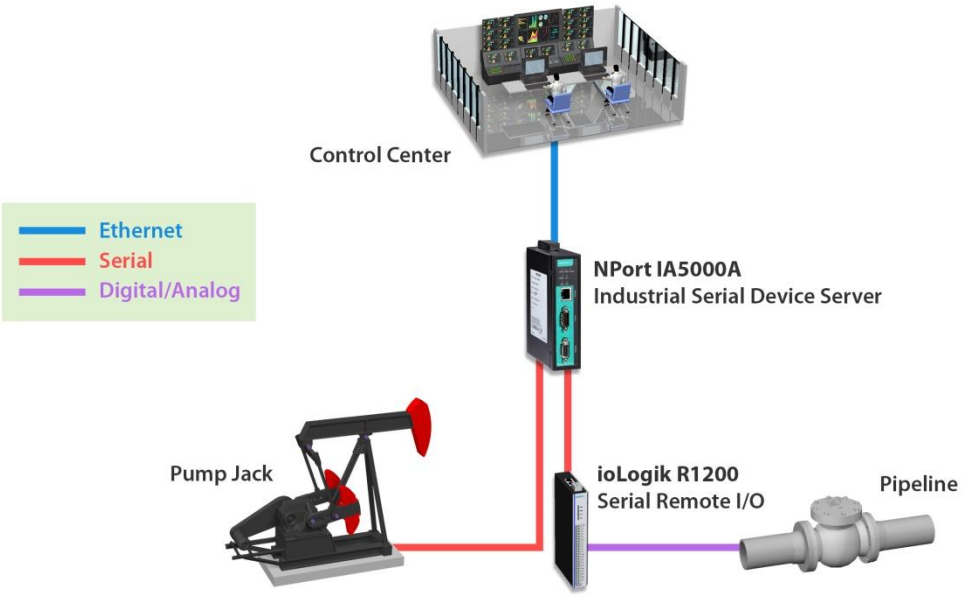
Command by Command Mode

its buffer. Commands will be sent to the serial ports on a FIFO (first in first out) basis. Once the device responds, the NPort will save that response to its buffer, assuming that the response is correct, and then send the response back to the originator of the command.



A potential problem when the host is operating in an active role to establish a TCP connection (where the NPort is acting as a TCP server to passively wait for the client to connect) is that the NPort has no way to determine when the network has crashed and will think that the connection is still there. Even if the network connection resumes, the client won't be able to reestablish connection with the device because the resource has been occupied. Consequently, users will need to send someone to the field site to reboot the NPort in order to free up the resource. This is extremely inefficient in terms of both labor and time costs. TCP server mode includes a function called **TCP Alive Check Timeout**. Using this function gives the NPort a fail-safe mechanism if the network is disconnected or the remote computer is in panic mode. In other words, this function provides the Ethernet connection status by checking the TCP/IP connection status periodically.

Application Sharing: Wellhead Monitoring



Scenario 4: Active Data Collection (H ← D)

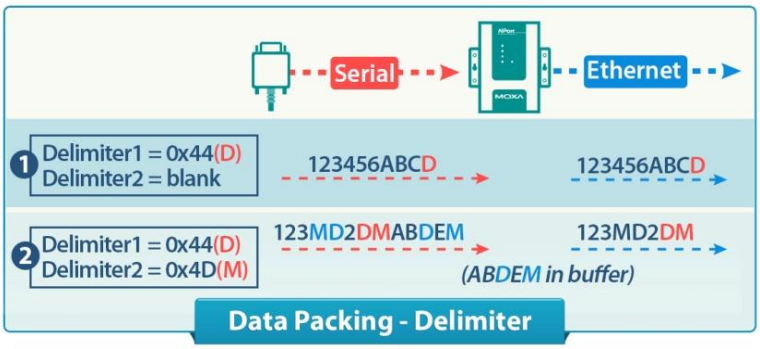
Challenges

- Implementing serial devices in a new environment that already has Ethernet wiring.
- Prevent the host from receiving a packet or datagram containing partial serial data, or else the access information cannot be recognized (i.e., the application program recognizes the "#" sign as the end of data stream).
- The user would like to limit the number of TCP connections to the hosts in order to free up the resources for other uses.
- The user would like to set up backup hosts for accessing information from serial devices.

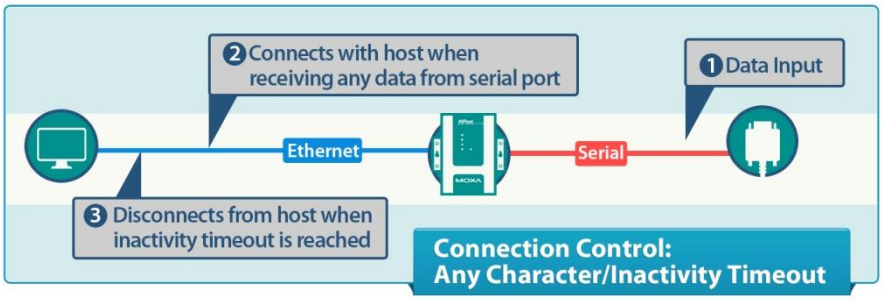
How NPort Can Help

TCP Client Mode | Max. Connection + Data Packing + Connection Control

The NPort's **TCP Client** mode is often used in access control systems to connect with serial card readers, fingerprint readers, and other devices. In this scenario, the data is actively sent back to the host application program for further processing. One problem with transporting serial data across a TCP/IP network is the possibility of data being divided into separate Ethernet packets that could cause the application program to fail. As mentioned earlier, the NPort's **Data Packing** function ensures that the serial data arrives in a complete and recognized packet so that the application can receive and process requests properly. We have already introduced Force Transmit as one of the Data Packing options, but **Delimiter** can also assist users in this example. Since the application program recognizes a specific character as the end of a data stream, when the **Delimiter** function is used, the NPort will immediately pack and send all data in the buffer to the Ethernet when a specific character is received through its serial port.



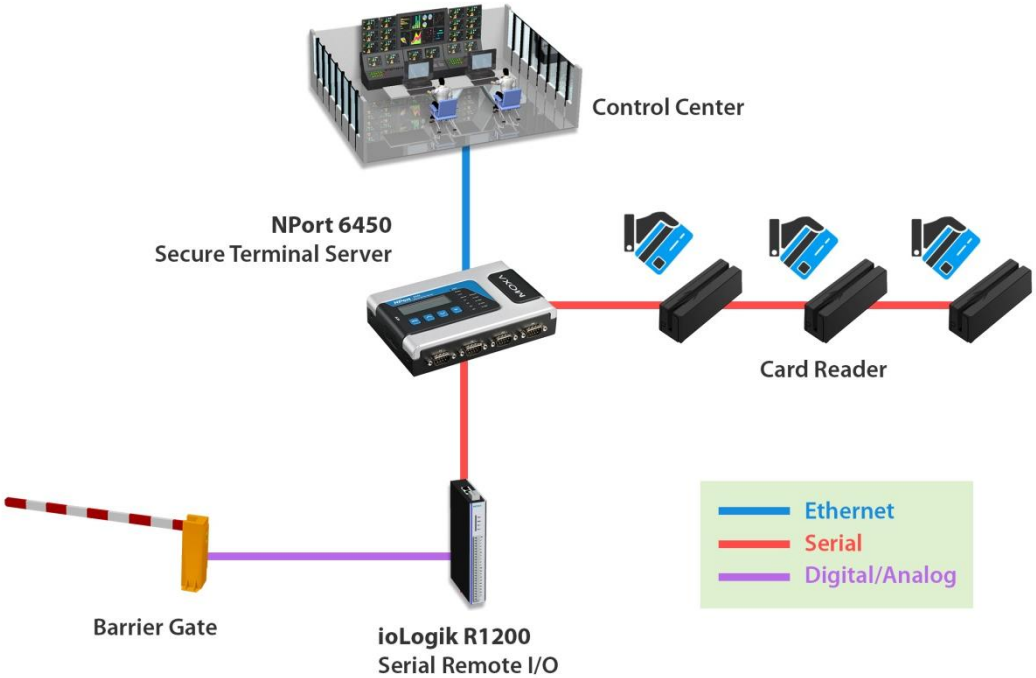
NPort Operation Mode Guidance



When the NPort is configured in TCP Client mode, it can decide when to establish or disconnect a TCP connection with the host by enabling the **Connection Control** function. The benefits of this function are that it allows the user to limit the number of TCP connections to those in need and increase host server efficiency by disconnecting unused connections automatically. Many different events can be defined to establish or disconnect a TCP connection. A very common one is **Any Character/Inactivity Timeout**. In this case, whenever there is any serial data activity, the NPort is triggered to establish a TCP connection with the host. And when the serial end is idle for a specified time, the NPort will disconnect the TCP connection until serial data activity is resumed.

Lastly, TCP Client mode also supports multiple host connections—up to 4 connections on the NPort 5000 series and up to 8 on the NPort 6000 series.

Application Sharing 1: Access Control System



Application Sharing 2: MRI Predictive Maintenance

The application shown in this section uses similar operation modes and advanced functions to achieve what the application requests. The main purpose of predictive maintenance is to help

determine the health status of in-service equipment in order to predict when the equipment service provider should perform maintenance jobs. This concept is introduced to enable remote monitoring of in-service equipment, such as MRI machines or elevators, to prevent unexpected failures that increase operation costs or even safety issues.

In the application shown here, the NPort allows the MRI to be monitored not only locally but also remotely where the usage data can be documented for downtime prevention. When the MRI scanner is functioning, the usage record of a particular machine will be sent by the NPort to the service provider who can schedule periodic maintenance based on utilization data. Not only can the usage data of the MRI be transmitted back to the service provider's monitoring system, but it can also be transmitted to the archive storage server for asset management (e.g., asset retirement).

Scenario 5: Data Broadcast (H → D)

Challenges

- Have messages sent from one host to multiple locations in real time.
- Be able to define groups of devices to receive different sets of messages

How NPort Can Help

UDP Mode | Multicast Group

If the application requires real-time data transmission and the socket program uses the UDP protocol, you can set the NPort to **UDP** mode. The major difference between UDP and TCP Server/Client modes is that with UDP mode, a connection does not need to be established before transmitting data. UDP mode can send data faster than TCP Server/Client because the time required for TCP's 3-way handshakes is eliminated. UDP mode is suitable for applications that require real-time transmission, but can tolerate possible data losses.

Some applications, such as passenger information systems on railway station platforms, need to broadcast (or multicast) the same messages to a set of LED displays to show the destination information or train schedules. In UDP mode, a multicast IP address can be set for every serial port, and all devices that "subscribe" to the same multicast IP address will receive the message that's assigned to that IP address. The benefit of multicast is that it not only efficiently sends the message to multiple destinations, but also saves valuable bandwidth because the same data is not transmitted to different destinations multiple times.

Application Sharing: Passenger Information System