Networked and distributed DAQ

Spring 2013 – Lecture #7
Data distribution and remote control

• **Sharing Data**
  – transferring the actual data among computers where you can perform different analyses on that data, depending on your needs. Some applications require streaming the actual data for additional processing, storage, or monitoring.

• **Remote Control**
  – Expands the concept of sharing data to include enabling another computer to connect to the experiment and control that experiment remotely.

• **Distributed Execution**
  – A system architecture that shares the acquisition and analysis of the test among several computers.
Scope of Communication

**Inter-process:** the exchange of data takes place within a single application context

**Inter-target:** communication between multiple physical targets, often over a network layer
Defining Inter-target Communication

- PC, RT, FPGA, Mobile Device
- Offload data logging and data processing to another target
- Multi-target/device application
- Network based
Common Pitfalls of Data Communication

Race conditions - two requests made to the same shared resource

Deadlock - two or more depended processes are waiting for each other to release the same resource

Data loss - gaps or discontinuities when transferring data

Performance degradation - poor processing speed due to dependencies on shared resources

Buffer overflows - writing to a buffer faster than it is read from the buffer

Stale data - reading the same data point more than once
DAQ setups

• **Local DAQ**
  – Signals to be measured and the computer are placed close to each other
  – Typically a plug-in DAQ-card is used in a PC, or a USB DAQ or PXI unit is connected to the computer

• **Remote DAQ**
  – Transfer data from a remote DAQ device to a single PC (host)

• **Networked (distributed) DAQ**
  – Distribute measurement data to several clients connected to a network
  – Enable a central computer to acquire all of the data from several machines and then process or store that data

**Diagram elements:**
- uC/FPGA & ADC
- RS-232
- RS-422
- Ethernet
- Data
- Save to file
How Do I Communicate with Other Systems?

- Network Data Transfer
  - TCP/UDP
  - Network Published Shared Variable (from LabVIEW)
  - Network Streams (from LabVIEW)

- Remote Application Control (from LabVIEW)
  - VI server
  - Network Published Shared Variable

- I/O Buses
  - Serial (RS-232, RS-422),
  - GPIB,
  - USB,
  - Reflective memory (Distributed shared memory)
LabVIEW Serial: RS-232

Basic Serial Write and Read.vi
Networked and distributed DAQ

- Requirements for transfer of data and/or control between widely distributed measurement and control devices
- Requires reliable communication between devices
  - **Analog communication (one-way)**
    - 4-20 mA DC current loop transmission *(used in industry)*
    - Differential line drivers *(e.g. for driving signals over long lines)*
  
- **Digital communication (two-way)**
  - *Ethernet*
  - *Fieldbus* *(a family of industrial computer network protocols used for real-time distributed control)*

![Diagram](image-url)
Networked DAQ Example

- One PC connected to a PXI chassis acquire, display, and store all measurement data from several sensor systems.
- Remote clients need access to the sensor data, to display and process the data in "real-time".

**UUT** = Unit Under Test
LAN

- A local area network (LAN) is a computer network that connects computers and devices in a limited geographical area — usually high data-transfer rates

- Ethernet is by far the most commonly used LAN technology

- Cabled networks and wireless networks
  - In some situations a wireless LAN may be preferable to a wired LAN because it is cheaper to install and maintain
Client – Server architecture

- In distributed computing a **server** can have all the data, and a number of **clients** (users) on the network can access the data

- Clients:
  1. Independent loops on a block diagram
  2. Run at different machines on a network

- Independent loops typically run at different rates

- Queues should be used for Client-server synchronization and data distribution for point 1

---

**Figure 3.27** Client-server systems. Client VIs receive data from a server VI through the use of a global variable. This permits time independence between several VIs or loops.
Wireless networks and DAQ

• Pros
  – With Wi-Fi and Ethernet DAQ devices you can perform remote measurements at distances as far reaching as the wireless network allows
  – Can be used where wiring is difficult or cost-prohibitive
  – Flexibility

• Cons
  – “Low” bandwidth
  – Less reliable than a cabled connection
  – Possible security restrictions
Rammer (engelsk: frames)

- Overføringsenhet mellom noder/adaptere
- En ramme = data (bit) som utgjør en naturlig helhet (variabelt eller fast antall bit/byte)
  - Bit som skal overføres (f.eks. pakke):
  - Bit som skal overføres, pakkes inn i en ramme:
  - Ekstra biter settes inn bak og/eller foran, og noen ganger inne i dataene som overføres. Hensikt?

- avgrense rammen
- detektere feil
- kontrollere flyt

Keywords: Ethernet frames
Generelt pakkeformat

- rammer
- pakker
- meldinger

Keywords: Network packet

PDU (Protocol Data Unit)

- Hode
- Data, for eks. 10.000 bit
- Hale

- M-adresse
- S-adresse

10101010

Sjekksum

CRC

INF1060
Jumbo frames

• In the early days of networking the maximum packet (frame) size was 1518 bytes.
• With today’s high transmission rates, the task of analyzing each packet can overwhelm the CPU.

• By using jumbo packets, you can transmit the same amount of data with fewer packets.
• Though you save a small amount of bandwidth (by using fewer headers), you dramatically reduce CPU usage because your PC spends less time analyzing packets.

A common jumbo frame size is 9 kB (8192 bytes is often used), though IPv4 supports jumbo packets up to 64 kB. Make sure that your NIC supports jumbo frames
Pakke multipleksing

- demultipleksing basert på adresser i pakkene (ruting)
Pålitelig overføring

- Pakker med feil sjekksum (CRC) kastes
- Fint om vi kan rette opp feilen
  - Må ha nok informasjon til å rette opp feil i de mottatte dataene
- Hvis feilen ikke kan rettes opp, og vi trenger pakken, må den sendes en gang til!
Hva er en protokoll?

Sammenlikning av menneskelig protokoll og en datamaskin-protokoll:

Protokoller definerer formater, rekkefølge for sending og mottak av meldinger, og de aksjoner som mottak av meldinger initierer.
IP tjeneste-modellen

- forbindelsesfri transport (datagram transport)
- "best-effort" overføring
  - ingen garanti for vellykket overføring:
    - pakker kan bli borte (sjekksumfeil & rutefeil)
    - pakker kan komme frem i gal rekkefølge
    - pakker kan dupliseres
    - pakker kan forsinkes unormalt
IP and TCP

- TCP and IP are two of the most important communication protocols used for the Internet
- TCP = Transmission Control Protocol, IP = Internet Protocol
- TCP complements the Internet Protocol (IP), which is unreliable
- **TCP/IP**: IP handles addressing and routing of message, **while TCP provides a reliable and in sequence data delivery without errors, loss (no packets are lost) or duplication**
- TCP:
  - Flow control (does not send data faster than the receiver can read)
  - Saturation control (slower transmission when network problems)
  - Retransmission of data when needed (data lost or not acknowledged in time)
- Example of use of TCP/IP: File transfer (FTP), HTTP
TCP

• TCP is a connection-based protocol, which means that a connection must be established before transferring data
  – Data transmission occurs between a client and a server
  – TCP permits multiple, simultaneous connections

• In order to establish a TCP connection you have to specify an address and a port at that address
  – The port numbers allow different applications on the same computer to share network resources simultaneously
  – In TCP (and UDP) port numbers start at 0 and go up to 65535. Numbers in the lower ranges are dedicated to common Internet protocols (like 21 for FTP and 80 for HTTP).
LabVIEW Data Communication
LabVIEW TCP Example

Demonstrates how to set up a TCP connection, and send data to a specified port once a connection (from a client) has been established.
Multicast and broadcast

- **Broadcast**
  - A transmission to all interface cards on the network

- **Multicast**
  - A transmission to a group of interface cards on the network
  - To receive data a client must join the multicast group
  - Multicasting uses the **IGMP** (Internet Group Management Protocol) and requires an IGMP-compliant switch
UDP

- Used for broadcast and multicast of data
- **Not reliable (packets can be lost)**
- UDP:
  - No flow control
  - No saturation control
  - No retransmission of data
- **UDP share the same delivery problems as IP**
- However,
  - UDP does not wait to confirm a connection before data transmission, and therefore no delay is introduced
  - Small overhead (compared to TCP)
  - UDP send rate only limited by the rate of data generation, CPU, clock rate and access to Internet bandwidth
- **Example of use of UDP:**
  - Video-conference (video distribution)
  - Sensor data distribution
  - NTP (network time protocol)
LabVIEW Example: UDP Send

**UDP Sender.vi**

- **Remote Port**: U16
- **Remote Host**: 255.255.255.255
- **Local Port**: 61556
- **Data String**: abc
- **# Repetitions**: 10
- **UDP broadcast address**: 255.255.255.255

- **localhost** = this machine; **IP** = 127.0.0.1 (used for testing)
LabVIEW Example: UDP Receive

UDP Receiver.vi

<table>
<thead>
<tr>
<th>Open UDP port</th>
<th>Read data from the port.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read number of bytes received and add them to the ones read in a previous iteration</td>
<td></td>
</tr>
<tr>
<td>Concatenate data received, and display it in the string indicator</td>
<td></td>
</tr>
<tr>
<td>If the UDP Read.vi times out, reset the error cluster so an error does not show</td>
<td></td>
</tr>
<tr>
<td>Close UDP port</td>
<td></td>
</tr>
<tr>
<td>Check for errors</td>
<td></td>
</tr>
</tbody>
</table>
LabVIEW Example: UDP Multicast sender
LabVIEW Example: UDP Multicast Receiver

Open a UDP port for Multicast Read Only.
Read from the Multicast Address
Ignore the timeout error that UDP Read may return
Concatenate previous with new messages and display them in a string indicator.
Close UDP port
Check for errors
Database (MS Access etc.)

- A database can be used to store processed data results
- A database is searchable
- Note that databases are usually not suitable for direct storage of raw data from DAQ (database communication is slow)

Database Connectivity Toolkit in LabVIEW: