NIST

IEEE 1588 Precision Clock Synchronization Standard Update

June 17-18, 2008

23rd Transducer Workshop

Buffalo, New York

Kang Lee

kang.lee@nist.gov

National Institute of Standards and Technology

Outline

- Time synchronization problems
- Brief overview of IEEE 1588
- Comparison with other protocols
- Application areas
- Status of IEEE 1588

Time synchronization problems in distributed measurement and control systems

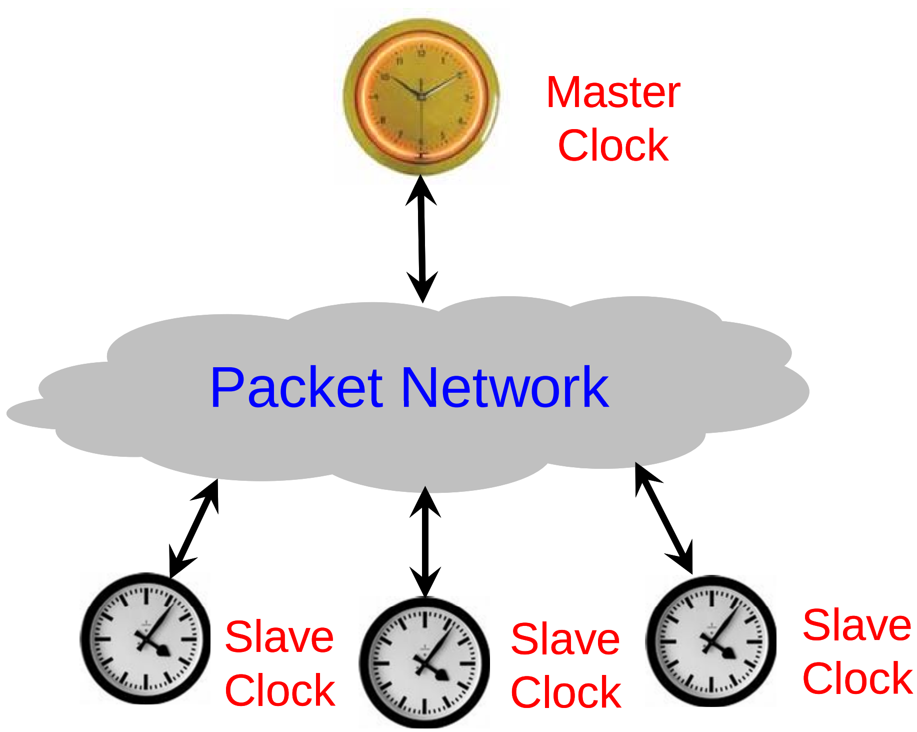
- Time synchronization is a problem in automated manufacturing, process control, networked systems, …
  - Events and data are received out-of-order
  - Out of order data due to poor time-stamping cause “False Positives” in fault detection systems bringing equipment down unnecessarily
  - Poor data quality due to delay or variability in delay cause out-of-control situations
  - Poor synchronization of data across multiple systems (equipment and measurement systems)
- Major problems in network-based synchronization: latency & fluctuations

Synchronization of Clocks

so one has a sense of the reference of time

What is IEEE 1588 ?

- ***IEEE 1588 is a protocol designed to synchronize real-time clocks in the nodes of a distributed system that communicate using a network.***



Why a new standard?

IEEE 1588 and other Time Dissemination Networks

- NTP does a good job for many years
  - target is autonomous systems widely dispersed on the Internet.
  - but some applications demand for much higher accuracy
- Specialized sync networks can do this job more accurate, but at much higher cost
  - e.g. IRIG-B, a specialized dedicated sync network
  - e.g. GPS, allows for global synchronization, requires outdoor antenna

Thus IEEE 1588 is defined to meet industry need…

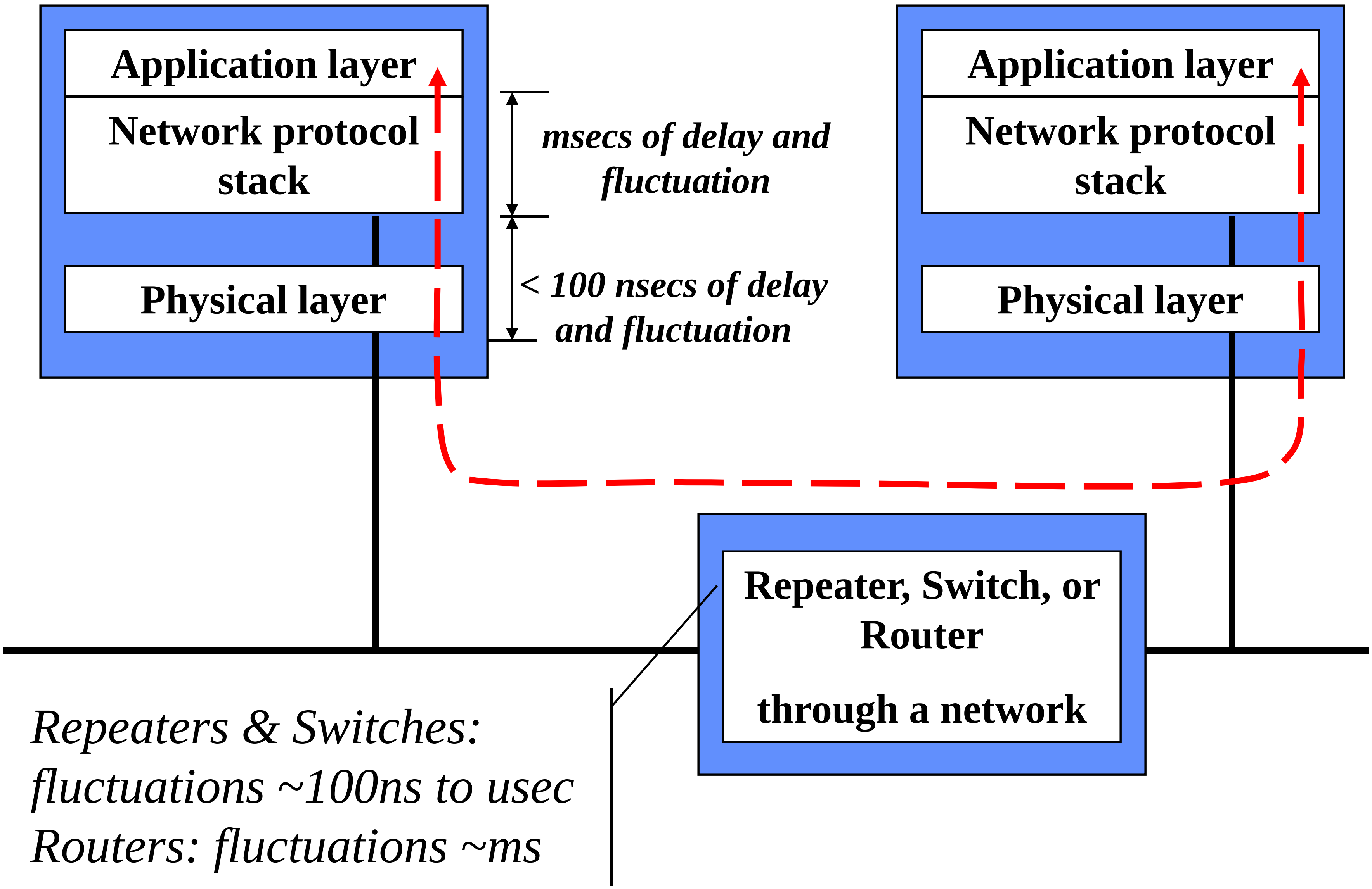
IEEE 1588

- enables a new methodology for networked measurement and control systems
  - ✓ based on time,
  - ✗ Difference from that based on the time-of-receipt in event notification
- can be applied to local area networks supporting multicast communications (including but not limited to Ethernet)
- supports heterogeneous systems of clocks with varying precision, resolution, and stability
- achieves sub-microsecond synchronization of real-time clocks with hardware assist

What applications require submicrosecond clock synchronization Accuracy ?

- **Automation and control systems**
  - **Synchronize multi axis drive systems**
  - **Synchronize subsystems with cyclic operation**
- **Measurement and automatic test systems**
  - **Correlation of decentrally acquired values**
  - **Timestamping of logged data**
- **Power generation, transmission and distribution systems**
  - **Control of switching operations**
  - **Reconstruction of network activities and events**
  - **Isolation of problems (distinguish cause and impact)**
- **Ranging, telemetry and navigation**
  - **Triangulation**
  - **Large sensors for seismic or submarine applications**
- **Telecommunications**
  - **Distribution of frequency and time in Next Generation Networks**
  - **Emulation of TDM circuits through packet networks**
  - **Synchronization of wireless base stations**
  - **Backup for other time sources (loss of GPS signal)**

Timing latency & fluctuation in various devices and layers in the network



Reducing Timing Latency & Fluctuation

- Within a node:
  - Make timing measurements as close to the physical (PHY) layer as possible to eliminate protocol stack and operating system fluctuations.
  - Use statistical techniques to further reduce residual fluctuations from PHY layer, network and repeaters and switches.

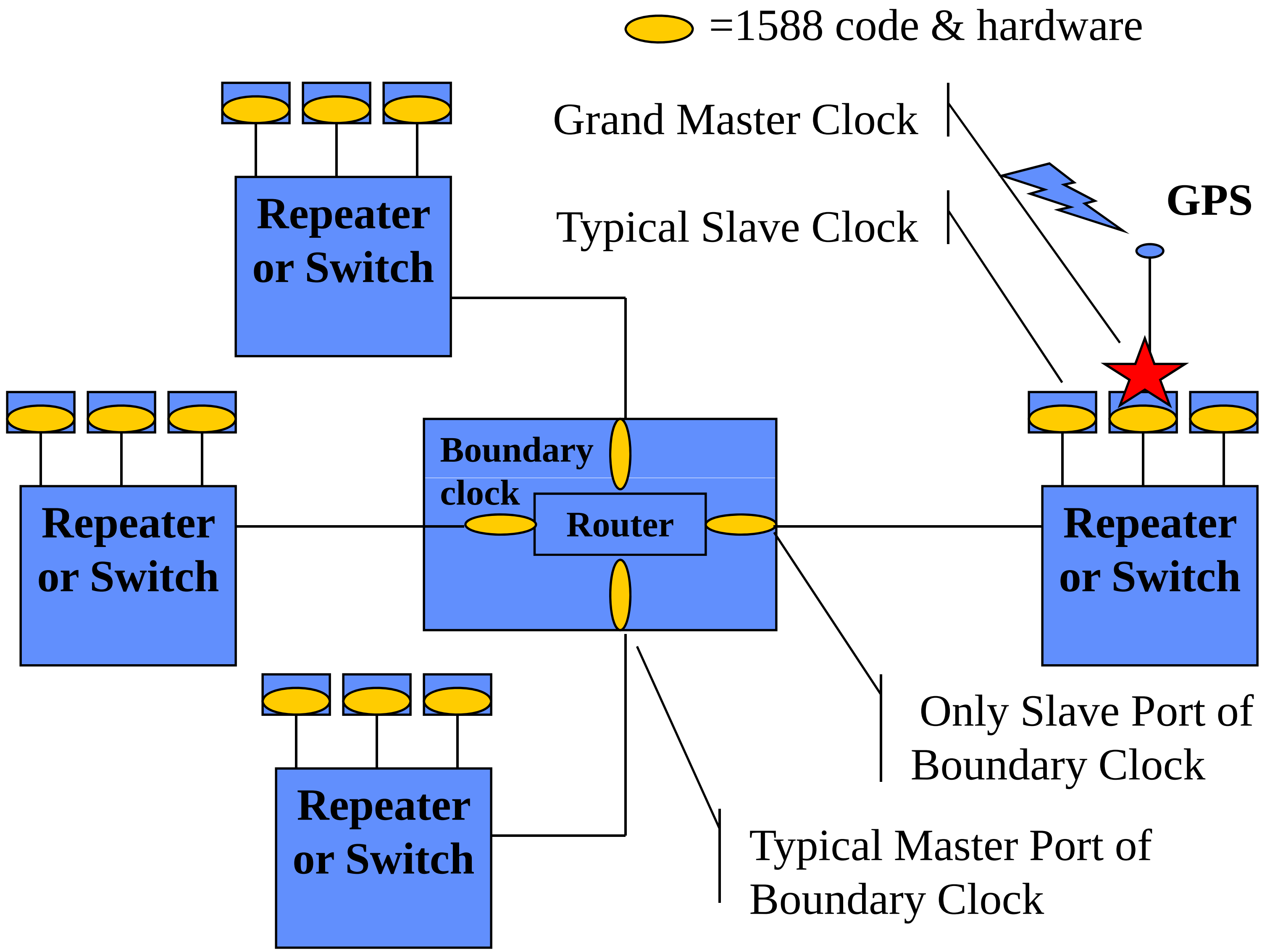
- For routers:
  - Use transfer devices (IEEE 1588 boundary and transparent clocks) to reduce router latency and fluctuations

IEEE 1588 Clock Types

IEEE 1588 defines

- **Ordinary Clock (OC)**
  - **A PTP clock with a single PTP port**
  - **Typically an end system**
- **Boundary Clock (BC)**
  - **A clock with more than a single PTP port**
  - **Typically a switch/bridge of the communication network with its own clock**
- **Transparent Clock (TC) - sends transit time information out**
  - **Peer-to-peer (P2P) TC– corrects for transit time in TC**
    - **Rapid recovery with changes in network topology**
    - **Only used in homogeneous P2P systems.**
    - **Requires a boundary clock at the edges.**
  - **End-to-end TC - provides transit time to other clocks, no correction**
    - **Good for linear systems of large # daisy chained clocks**

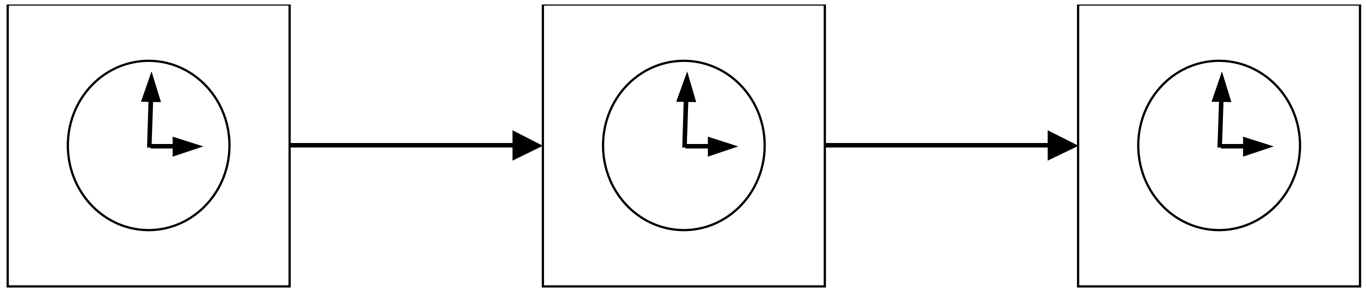
IEEE 1588 Multiple Subnet Topology
(simplified for illustration purpose)



IEEE 1588 Synchronization Basics

Step 1: Organize the clocks into a master-slave hierarchy (based on observing the clock property information contained in multicast Sync messages)

Step 2: Each slave synchronizes to its master (based on Sync, Delay_Req, Follow_Up, and Delay_Resp messages exchanged between master and its slave)

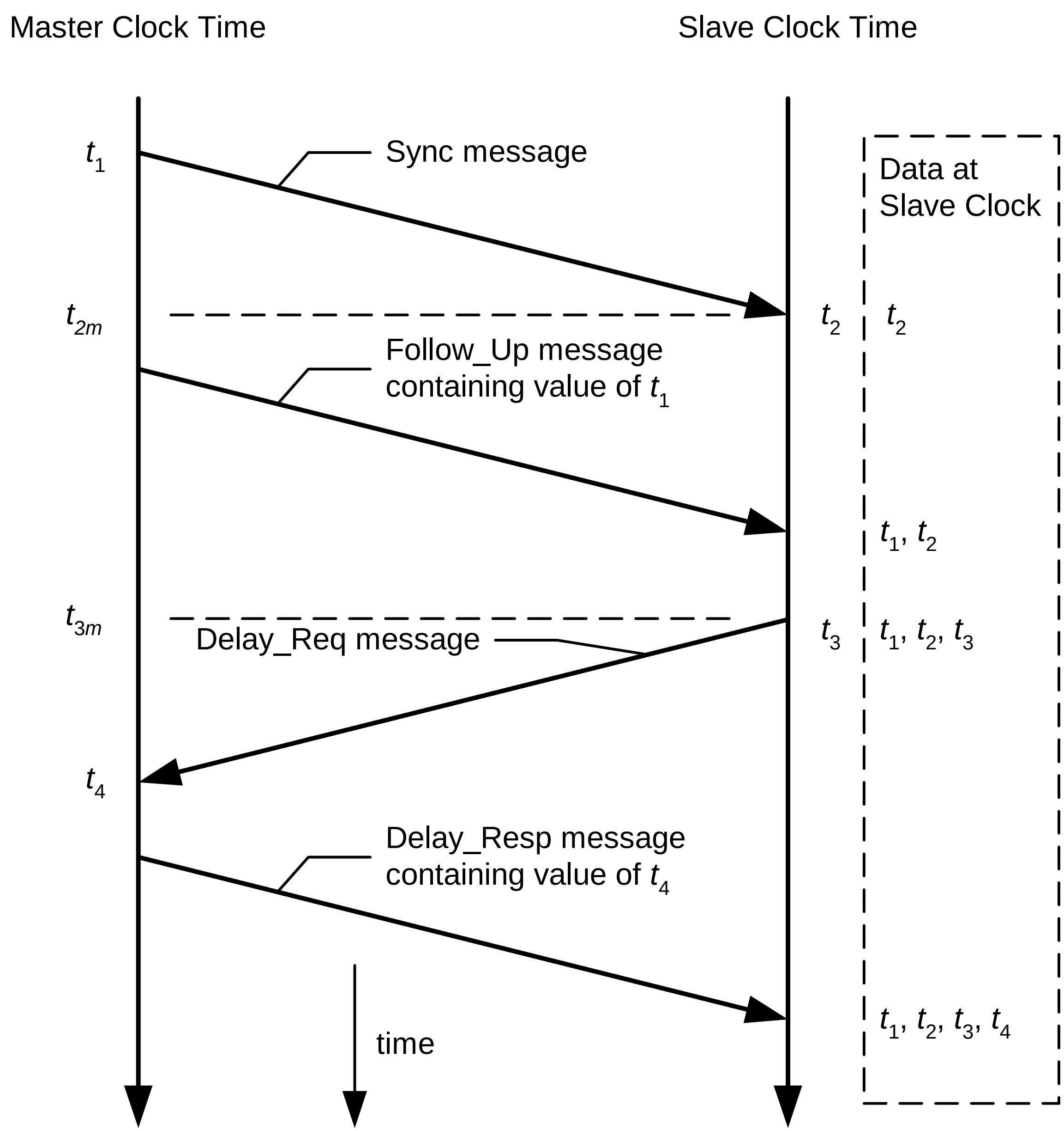


Grandmaster Clock
This clock determines the time base for the system

Slave to the Grandmaster Clock and Master to its Slave

Slave to its Master

Synchronization Basics (continued)



Synchronization Basics (continued)

- **To synchronize a pair of clocks, First:**
- **Send a message, (Sync message), from master to slave and measure the apparent time difference between the two clocks.**
- **MS_difference = slave's receipt time – master's sending time = $t_2 - t_1$**
- **MS_difference = offset + MS delay (by inspection)**
- **For example:**

  MS_difference = slave's receipt time – master's sending time
  90 minutes = 11:30 – 10:00

Master Clock:
10:00AM

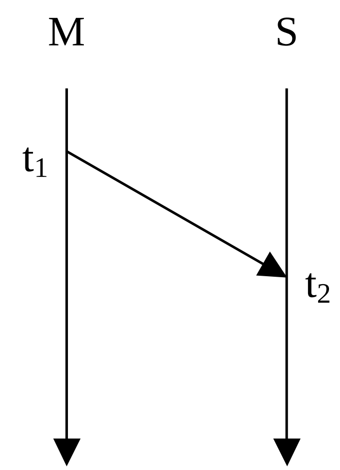
Offset = 1 hour

Slave Clock:
11:00AM

Sending time:
10:00AM

Send message with Propagation
Time = 30 minutes

Receipt time:
11:30AM



NIST

Synchronization Basics (continued)

Second:

- **Send a message, (Delay_Req message), from <u>slave</u> to <u>master</u> and measure the apparent time difference between the two clocks.**
- **SM_difference = master's receipt time – slave's sending time = $t_4 - t_3$**
- **SM_difference = – offset + SM delay (by inspection)**
- **For example:**

SM_difference = master's receipt time – slave's sending time

– 20 minutes = 11:10 – 11:30

Master Clock:
10:30AM

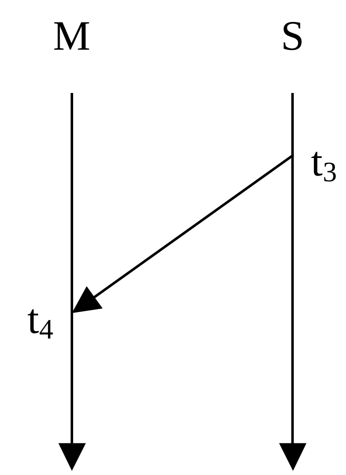
Offset = 1 hour

Slave Clock:
11:30AM

Receipt time:
11:10AM

Send message with Propagation
Time = 40 minutes

Sending time:
11:30AM



Synchronization Basics (continued)

- The result is that we have the following two equations:
- MS_difference = offset + MS delay
- SM_difference = – offset + SM delay

With two measured quantities:

- MS_difference = 90 minutes
- SM_difference = – 20 minutes

And three unknowns:

- offset , MS delay, and SM delay

Synchronization Basics (continued)

Rearranging the two equations:

- MS_difference = offset + MS delay
- SM_difference = – offset + SM delay

We get:

- offset = {(MS_difference – SM_difference) – (MS delay – SM delay)}/2
- MS delay + SM delay = {MS_difference + SM_difference}

ASSUME: MS delay = SM delay = one_way_delay

Then:

- offset = {MS_difference – SM_difference}/2
- one_way_delay = {MS_difference + SM_difference}/2

Synchronization Basics (continued)

- offset = {MS_difference – SM_difference}/2
- one_way_delay = {MS_difference + SM_difference}/2

<u>In our example using the two measured quantities:</u>

- MS_difference = 90 minutes
- SM_difference = – 20 minutes

We get:

- offset = {90 – (– 20)}/2 = 55 minutes (not actual 60)
- one_way_delay = {90 + (– 20)}/2 = 35 minutes (not 30 or 40)

Comparison with Other Protocols

	1588	NTP	GPS	TTP	SERCOS	
Latency correction	Yes	Yes	Yes	Configured	No	
Protocol specifies security	No	Yes	No	No	No	
Administration	Self organizing	Configured	N/A	Configured	Configured	
Hardware?	For highest accuracy	No	RF receiver and processor	Yes	Yes	
Update interval	~2 seconds	Varies, nominally seconds	~1 second	Every TDMA cycle, ~ms	Every TDMA cycle, ~ms	

NIST

Comparison with Other Protocols (Continued)

	1588	NTP	GPS	TTP	SERCOS
Spatial extent	A few subnets	Wide area	Wide area	Local bus	Local bus
Communi-cations	Network	Internet	Satellite	Bus or star	Bus
Target accuracy	Sub-microsecond to less than a nanosecond	Few milliseconds	Sub-microsecond	Sub-microsecond	Sub-microsecond
Style	Master/slave	Peer ensemble	Client/server	Distributed	Master/Slave
Resources	Small network message and computation footprint	Moderate network and computation footprint	Moderate computation footprint	Moderate	Moderate

IEEE 1588 Enhancements

- New requirements in applications: industrial automation, test and measurement
- New requirements in new application areas: telecommunication, IEEE 802.1as, power industry, military
- Higher accuracy
- Varied update rates
- Linear topology, in addition to hierarchical topology
- Rapid reconfiguration after network changes
- Fault tolerance
- Support for QoS
- Security

Some Application Areas

Distributed motion control – controllers, drives, MMI, and I/O are connected through Ethernet switches using IEEE 1588 for time synchronization.

- Control of multiple robots in a production line
- 'Web' handling equipment speed profile control, e.g. printing presses, rolling mills
- Complex packaging machines

60 miles/hour = 1 mil / microsecond

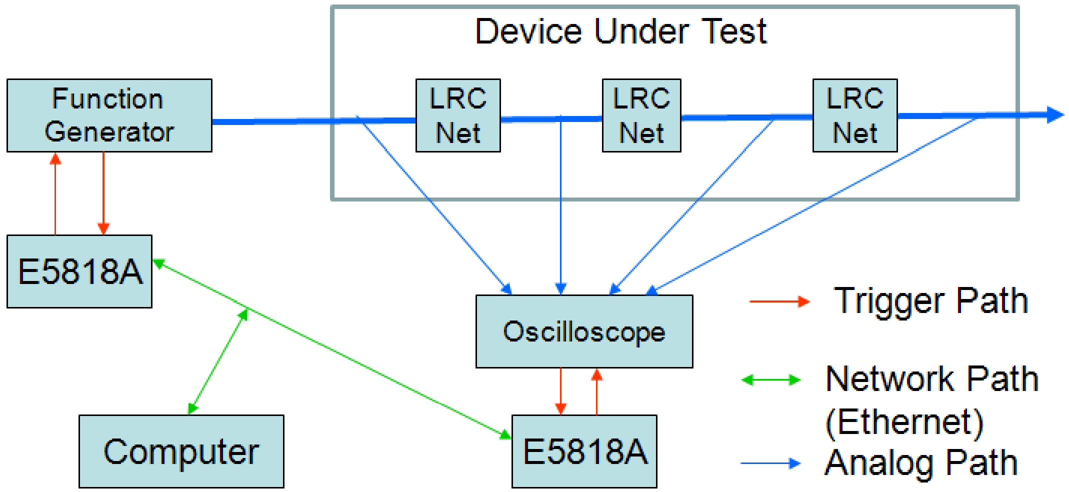
Some Application Areas (continued)

- Test and Measurement
  1. Moving from bus-based instrument systems (IEEE-488 or GPIB) to network-based modular systems (LAN or Ethernet).
  2. Synchronization needs vary widely with application
     a. Low to sub-nanosecond for most demanding
     b. Microseconds to milliseconds for less demanding
- LXI Consortium
  1. Consortium of test and measurement equipment vendors and users
  2. LXI Specification:
     a. Mandates the use of IEEE 1588 for LXI Class B instrumentation
     b. Specifications on how to use IEEE 1588 in instruments
     - Timestamp data and events
     - Time-triggers
     - Peer-to-peer LAN messages containing event timestamps

Moving from bus-based to network-based

IEEE 1588-based Trigger Box (an example)

- E5818A Trigger Box (Agilent) handles the IEEE-1588 Precision Time Protocol (PTP) Synchronization over the Ethernet and
- Manages the precision trigger control signals to the legacy instrument that does not comply to LXI standard.
- LXI Class-B triggering device
- Synchronization with IEEE 1588 PTP to 20 nanoseconds
- Easier troubleshooting with event time stamping

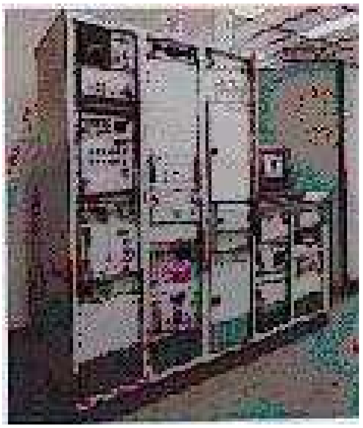


NIST

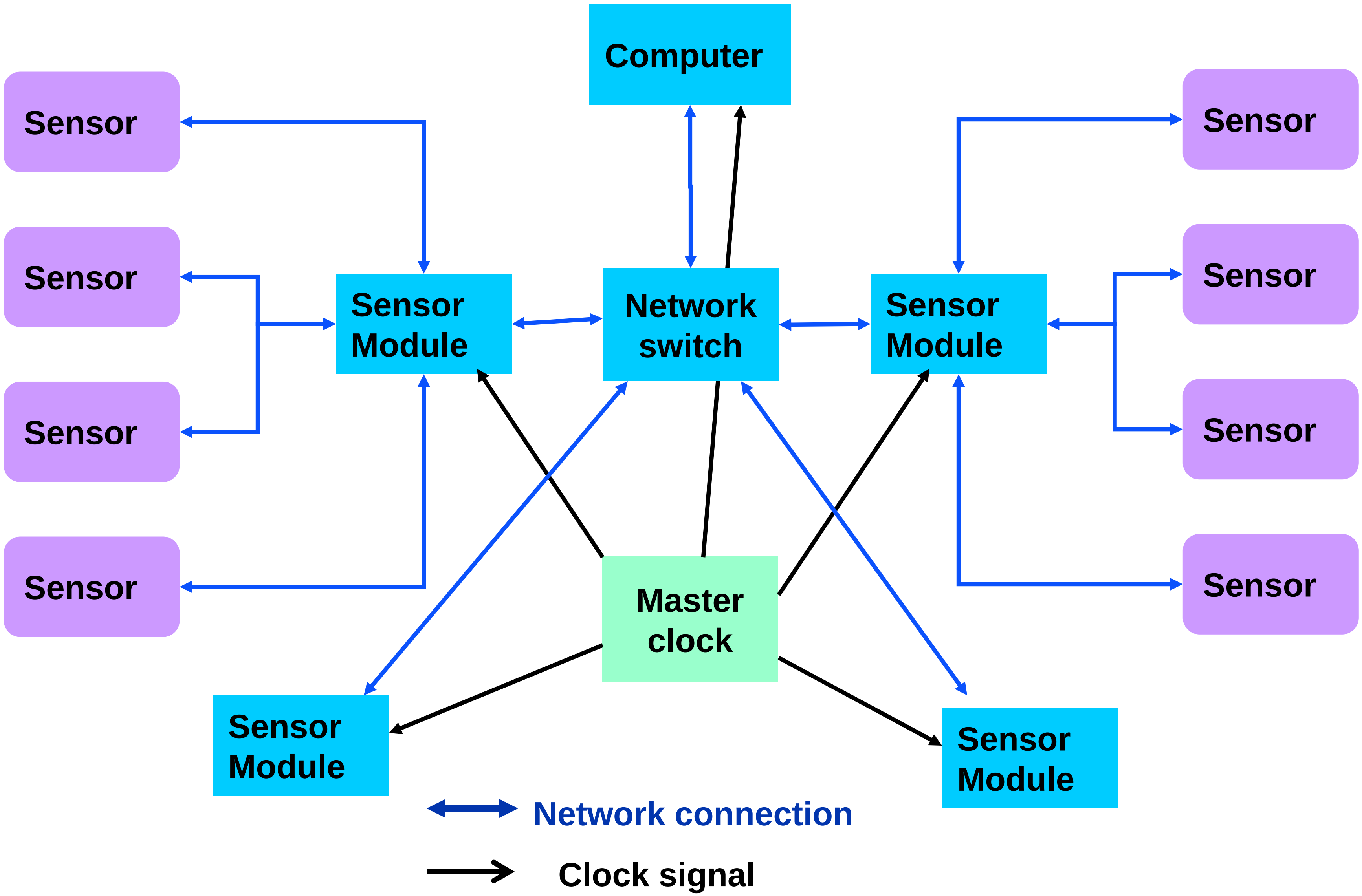
Applications in Military Systems

1. **Variety of potential applications**
   a. **Depot and test ranges**
   b. **Flight test & qualification**
   c. **Operational systems**

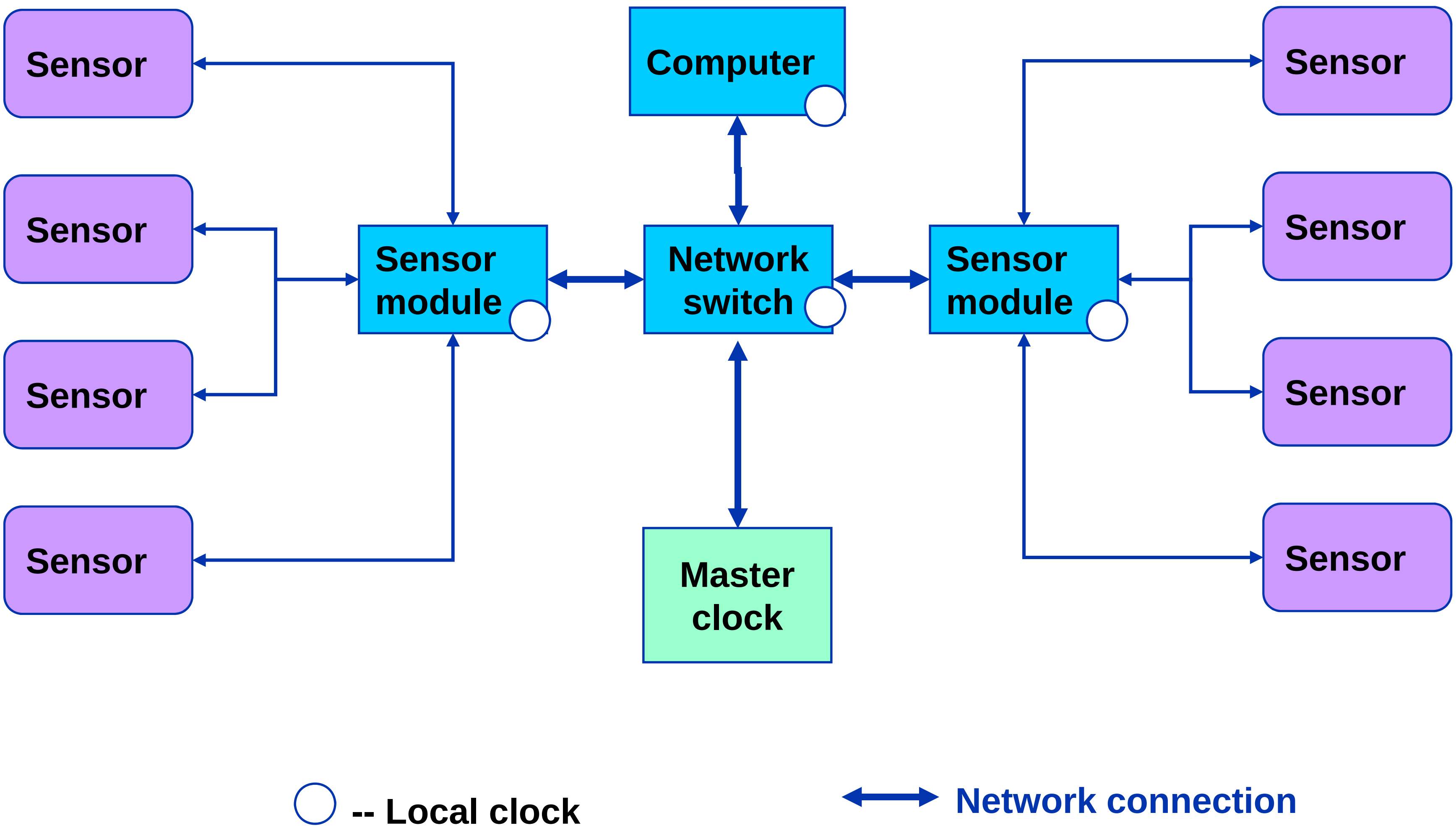
2. **Requirements are very similar to test and measurement**



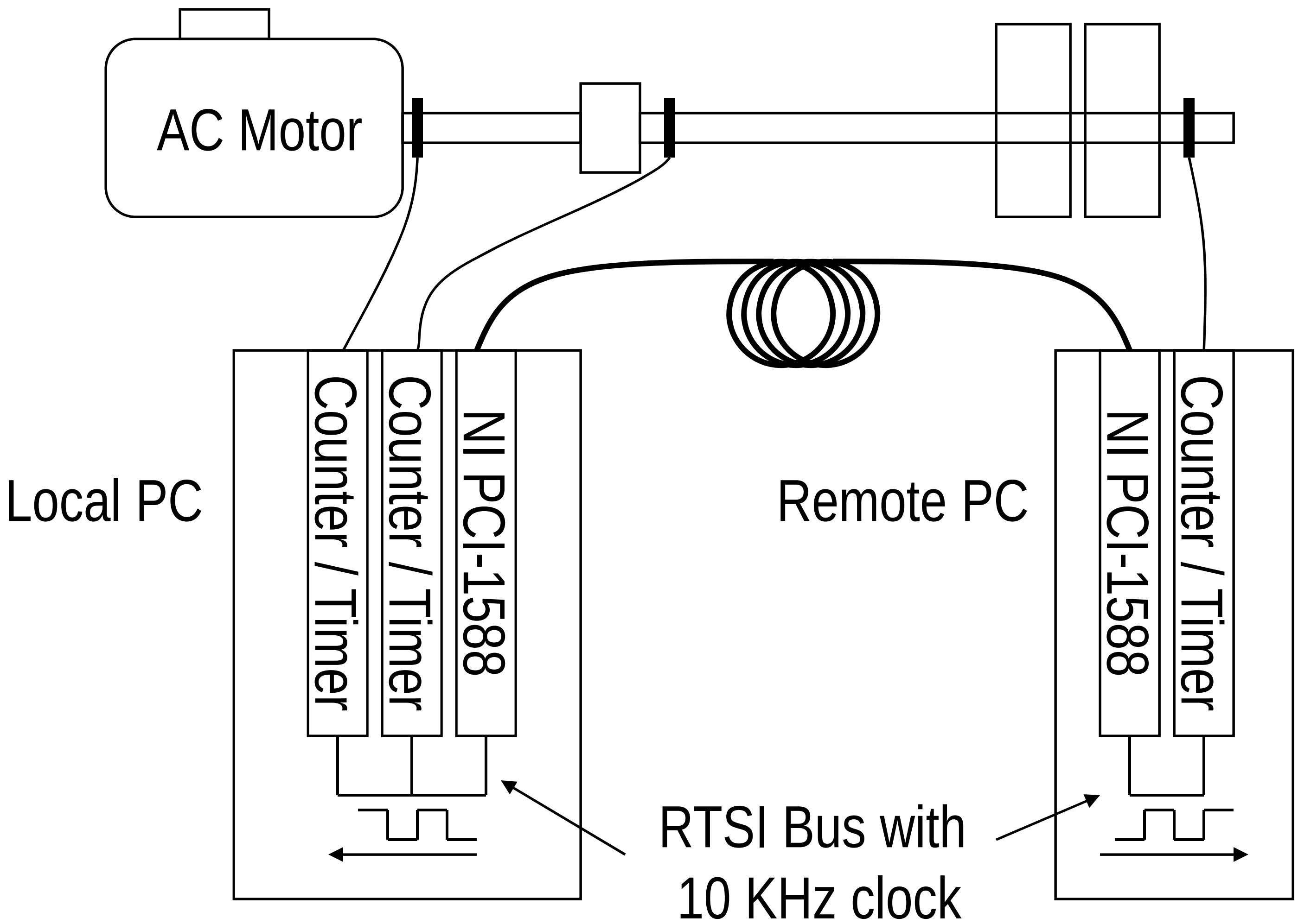
Time Stamping of Sensor Data (traditional)



Time Stamping of Sensor Data
using a time-based method

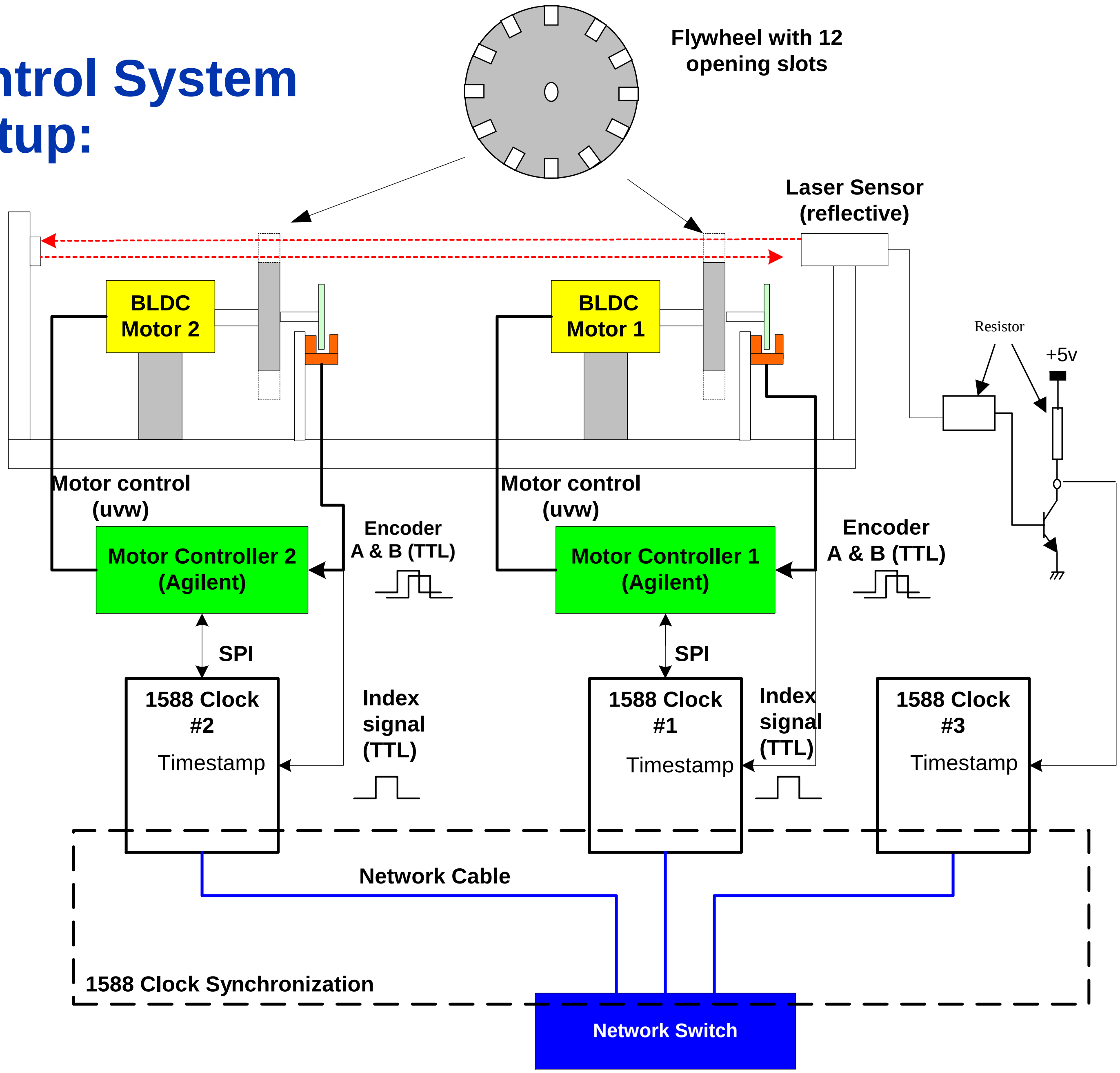


National Instruments Distributed System - Clock Signaling



Courtesy: National Instruments

Motion Control System Setup:



Courtesy: Agilent Lab

The Status of IEEE 1588 and IEEE 1451

- IEEE1588-2008 (version 2) standard - target publication date: 6/2008
- Standard is available from the IEEE http://standards.ieee.org/reading/ieee/std/numerical.html
- Conferences on IEEE 1588 held in 2003, 2004, 2005, 2006 2007.
- Next conference will be held on Sept 22-26, 2008 at the University of Michigan, Ann Arbor, MI.

  See http://www.ispcs.org for more detail.
- Latest information on IEEE 1588 precision synchronization protocol and IEEE 1451 smart transducer standards may be found at http://ieee1588.nist.gov and http://ieee1451.nist.gov, respectively.
- Further contact: Kang Lee at kang.lee@nist.gov