Fieldbuses: FIP

Jérôme Ermont
jerome.ermont@enseeiht.fr

3SN parcours E (SEMBIIOT)

Notes

Agenda

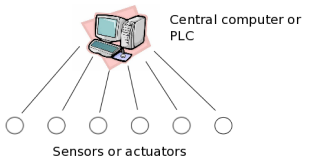
Notes

Agenda

Notes

Introduction

- Before: a centralized topology



Advantages	Disadvantages
Only one link between sensors or actuators and PLC	Globally a lot of links
→ no collision	→ increasing the costs
→ failures easily detected	→ size of the system is reduced
→ respect of real-time aspects	→ scalability ?

Notes

Introduction

- To reduce costs and to allow scalability: a shared bus
- Problem:
  - How to share a medium taking into account the respect of real-time properties ?
- Different solutions exist in real-time networks domain
- Two specific solutions in the industrial domain:
  - PROFIBUS
  - FIP

Notes

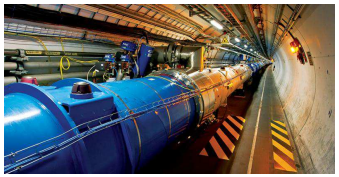
Agenda

Notes

FIP briefly

- FIP = *Factory Information Protocol*
- french initiative (1980 - NF)
- industrial domain: communication between
  - automation systems
  - sensors (variables)
  - actuators (commands)
  - controllers and maintenance systems

- Example: LHC at CERN

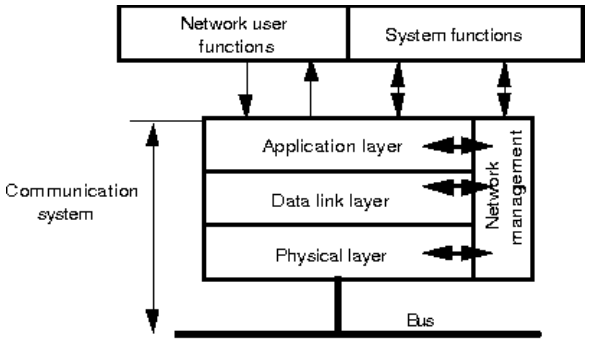


Goal:

To guarantee periodic transmission deadlines taking into account the aperiodic transmission

Notes

FIP architecture



Notes

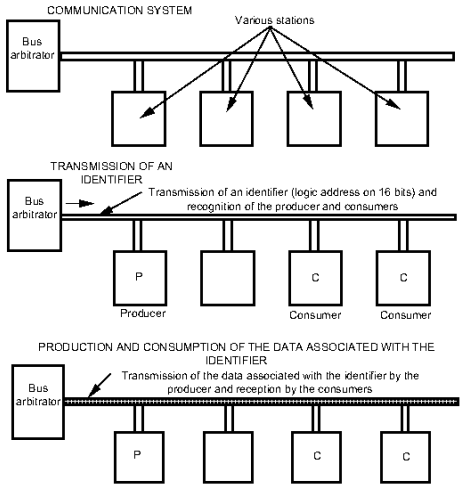
Behavior

- Bus arbitrator
  - Controls the data transmission between producers and consumers
- Cyclic polling of the bus to ensure determinism
- Different types of data transmission:
  - Cyclic data transmission (sensors, commands), priority
  - Acyclic data transmission:
    - critical (alarms)
    - non critical (maintenance), no priority
- data (variables or commands) $\rightarrow$ identifier (address length: 16 bits)
- producer: recognize the identifier of the required data and put the data on the bus
- comsumers: get data from the bus

$\rightarrow$ transmission uses *broadcast*

Notes

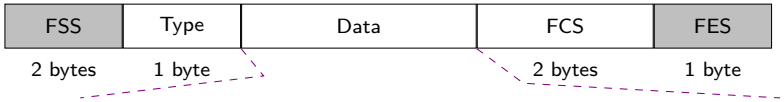
Behavior (cont'd)



Notes

Frame format

- 2 types of frames:
  - Frames sent by the arbitrator: identifiers
  - Frames sent by the producer: data



Frame type	**Content**	**Size**
Question Frame (`id_dat`, `id_req`, `id_msg`)	Identifier	2 bytes (var.), 3 bytes (msgs)
Response with data (`rp_dat`, `rp_dat_rq`)	Data	$<$ 128 bytes
Response to request (`rp_rq`)	List of identifiers	n $\times$ 2 bytes (n$<$64)
Response with msg (`rp_msg_[no]ack`)	@src, @dest, msg	3B, 3B, $<$256B
Other response (`rp_ack`, `rp_fin`)	NULL	0

Notes

Polling table

- list of identifiers of periodic data
- schedules the transmissions on the bus
- elementary cycle (micro-cycle): time slot of the bus polling by the bus arbitrator
- macro-cycle: juxtaposition of several elementary cycles
- duration of micro-cycle and macro-cycle:

$$T_{mc} = \text{gcd}\{P_i\}, \text{ where } 1 \leq i \leq p$$
$$T_{MC} = \text{lcm}\{P_i\}, \text{ where } 1 \leq i \leq p$$

with P_i periods of p transmission

Notes

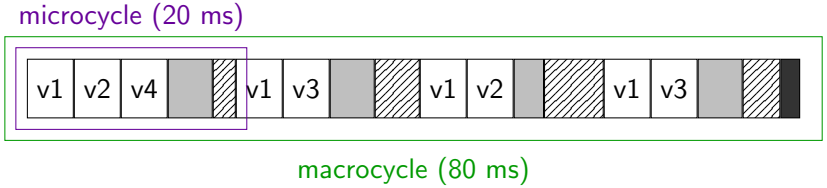
And the aperiodic messages ?

- duration of transmissions on the bus << micro-cycle duration

→ it remains time

→ used to send aperiodic messages

- Example of polling table:
  4 periodic variables v1 (20 ms), v2 (40 ms), v3 (40 ms), v4 (80 ms)



Notes

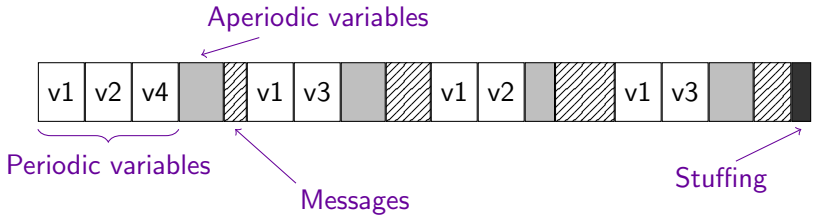
And the aperiodic messages ?

- duration of transmissions on the bus << micro-cycle duration

→ it remains time

→ used to send aperiodic messages

- Example of polling table:
  4 periodic variables v1 (20 ms), v2 (40 ms), v3 (40 ms), v4 (80 ms)



13 of 19

Notes

Scheduling of messages

- Out of the scope of the FIP normalization
- 2 possible methods:
  1. priorities associated to messages (dynamic priorities)
  2. priorities associated to flows (static priorities)
- Principles:
  1. priority assigned to each message according to the number of messages by flow
  2. priority assigned to a flow of messages using a scheduling algorithm as, for ex., RM or ED

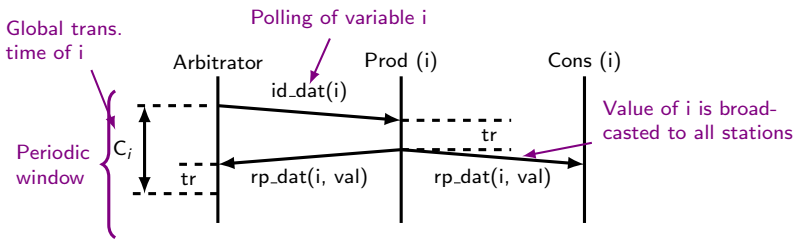
Notes

Different kinds of variable transmission

- Periodic polling of periodic variables
- Periodic (direct) polling of aperiodic variables
  - The arbitrator asks periodically to the stations if they have aperiodic variables
- Non-direct polling of aperiodic variables
  - during the periodic window, stations indicates that they have aperiodic variables to send
  - after the periodic window, the arbitrator asks for the aperiodic variables

Notes

Periodic variable transmission

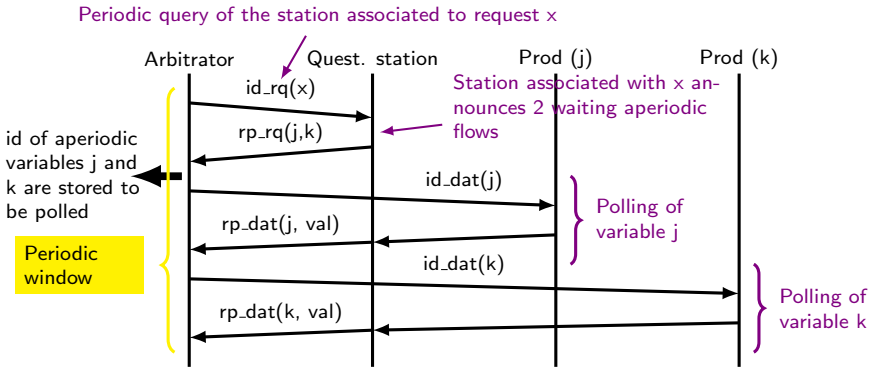


tr = response time

- at 31.25Kb/s: 22.4μs<tr<320μs
- at 1Mb/s: 10μs<tr<70μs
- at 2.5Mb/s: 4μs<tr<28μs

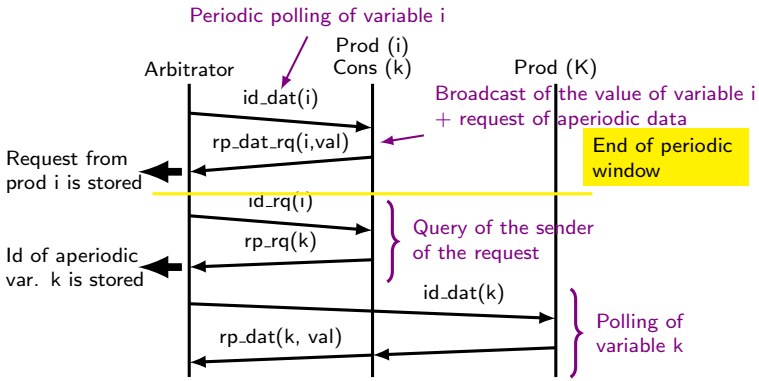
Notes

Aperiodic variable transmission: direct polling



Notes

Aperiodic variable transmission: non-direct polling



Notes

Exercise: Construction of a polling table

Lets consider the different messages characteristics:

Variable	Period (msec)	Transmission Duration (μsec)
a	4	100
b	16	150
c	4	150
d	8	200
e	64	450

Questions:

1. Microcycle and Macrocycle values ?
2. Construct a polling table which allows to schedule the different messages. Do messages respect their temporal constraints ?

Notes