

3_1_Introduction

Distributed Control Systems

Design Flow of DCS

Components

Processing Units

Communication Over Bus

Communication Tasks

Application Tasks

Data Dependencies

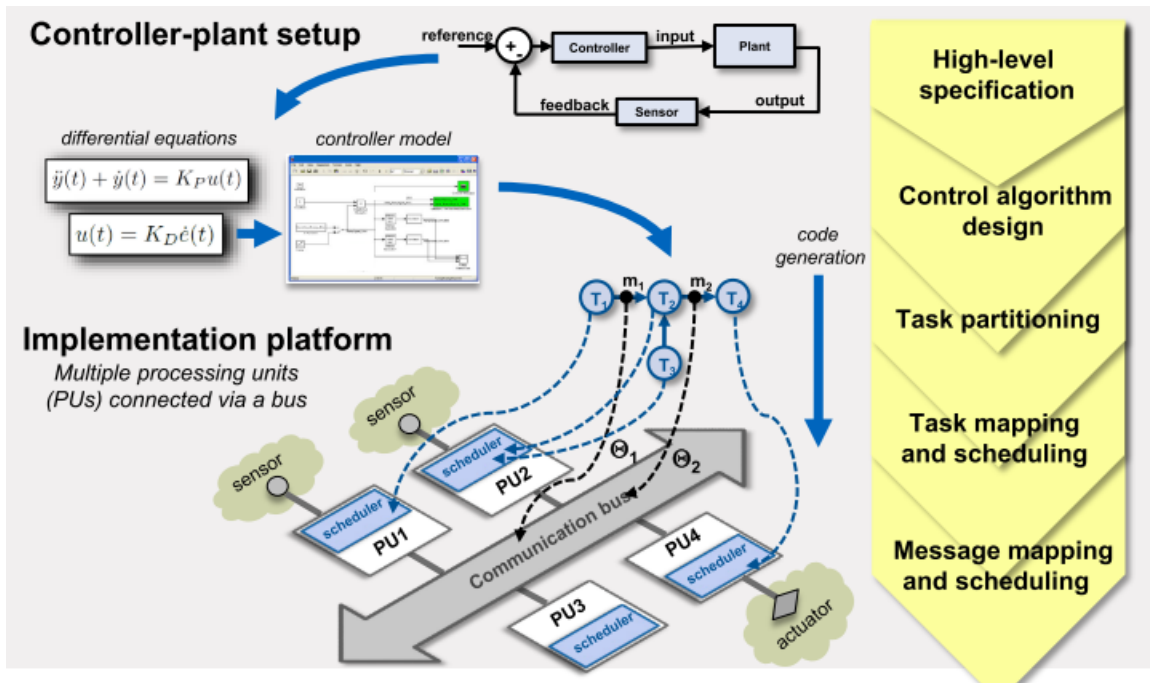
End-to-end Delay

Communication Protocols

Distributed Control Systems

- **Multiple processing units** connected over communication bus systems;
- The processing units execute one or more tasks; the tasks are scheduled by **a scheduler or real-time operating system**
- The tasks running at different processing units may communicate with each other over a communication bus
- The communication bus implements **specific communication protocol**
- The communication uses **input and output buffers** to store and dispatch the incoming and outgoing messages

Design Flow of DCS



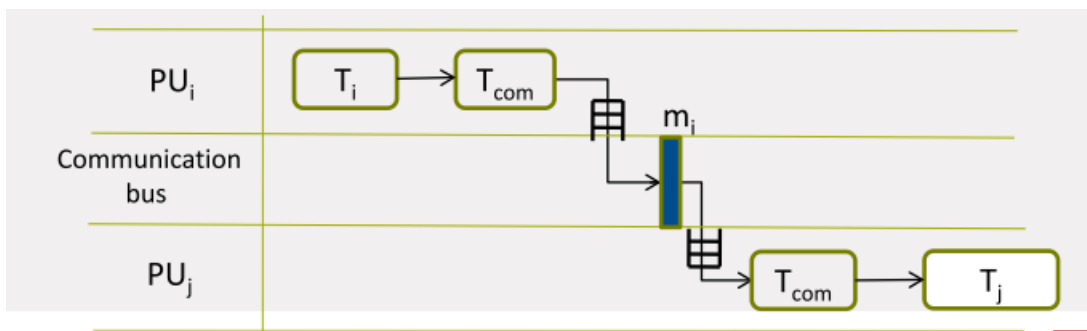
Components

Processing Units

- Host Micro-Controller: Scheduler or RTOS
- Communication Controller: Communication Controller
- Bus Driver: Bus Driver

Communication Over Bus

- Application task T_i in PU_i sends message m_i to application task T_j in PU_j
- T_{com} in PU_i packetizes m_i and places it in the output buffer
- PU_j receives m_i and places in the input buffer
- T_{com} reads the input buffer in PU_j and depacketizes m_i and passed on the task T_j



Communication Tasks

- A communication task T_{com} writes the output message m_i of the sending application task T_i to the dedicated output buffers of the communication controller
- In the receiving PU, the communication task T_{com} reads the corresponding input buffer and forwards the unpacked data to the application task T_j for further processing

Application Tasks

A dispatch event for a task can be defined as $T_i : \{p_i, D_i, e_i\}$

Data Dependencies

- Task T_i in PU_i sends message m_i to Task T_j in PU_j ; data-dependency between T_i and T_j
Task T_j in PU_j sends message m_j to Task T_i in PU_i ; data-dependency between T_j and T_i

End-to-end Delay

- End-to-end path: $T_i \rightarrow \text{bus} \rightarrow T_j \rightarrow \text{bus} \rightarrow T_k$
- End-to-end delay is important to meet the application-level requirements

Communication Protocols

- Time-triggered protocol: Time-triggered Ethernet
- Event-triggered protocol: CAN
- Hybrid protocol: FlexRay