Testing Embedded Control Software of Mechatronic Systems with Software-In-The-Loop Simulation
Structure

• Océ

• Why Software-In-The-Loop (SIL) Simulation

• Assignment
Océ

• Océ, Venlo NL
  • sells office printing and copying systems, high speed digital production printers and wide format printing systems for both technical documentation and color display graphics as well as related software and other services
Océ
Océ
Mechatronic Systems

- collaboration of people from multiple disciplines
Challenges: Mechatronic Systems Development

- Plant is needed to test software.
- Development of the embedded software (ESW) depends on other disciplines:
  - Early stages of project

*Time (not to scale)*

- Architecture
- Mechanics
- Electronics
- Embedded Software
Challenges: Mechatronic Systems Development

- In most cases not feasible to test on prototype
  - Limited resources
  - Long preparation
  - Error injection/reproduction
  - Expensive hardware
Possible Solution

- Use simulated plant, emulate plant behavior
Possible Solution

• Use simulated plant, emulate plant behavior
  – Testing on local computer
  – Shorter feedback cycles
  – Error injection / reproduction
  – When putting together software is already tested
  – Automated testing
Challenges: Simulation based testing

- Plant behavior has to be modeled/simulated sufficiently accurate
  - Who models? On which level of abstraction?

- Include ESW with as little as possible changes

- ESW and plant model communicate via sensors and actuators

- Level of abstraction of communication

- Testing facilities: test case creation, execution, evaluation
Concrete Case

- SIL simulation for transport of paper through the printer (Paper handling or Sheet logic) in use
- Has to be extended to test rest of the printer
  - Ink, toner transport
  - Climate
  - Moving parts e.g. carriage

- Two main assignments
  - Refine SIL architecture
  - Develop plant modeling approach
Refine SIL architecture

• Current situation
  – One plant model, Paper handling
  – Some simple handmade plant models
  – ESW is compiled with SIL simulation I/O Layer
  – Dedicated interface between ESW and Sheet Logic
  – SIL core sets function pointers in ESW for each simulated entity in Sheet logic
  – i.e. digital sensor/actuator, analog sensor/actuator, different motors
Refined SIL architecture

• Goals:
  • Refine interface between ESW and plant
    – More generic
    – High Extensibility
    – High modularity
    – “Plug and Play”

• Communication based on variables

• ESW and plant model modules get registered with a set of input and output variables in the SIL core

• Variable database in the SIL core holds the current state of all variables

• Variables are matched by name
Plant modeling approach

• Sheet Logic (paper handling)
  – Sheet Logic consists of several I/Os like digital/analog sensors, digital/analog actuators, different motors
  – Motors drive pinches that move the sheets of paper which move over paper path segments.
  – Sensors sense, for example, the sheet movement or speed of a motor.

• This data is read from automatically generated xml files and the corresponding elements are instantiated and connected (no hardcoded model)
Plant modeling approach

- New approach similar to Sheet Logic but much more expressive through more generic plant element concepts

- Data over plant elements is gathered from an interdisciplinary model that is developed in parallel

- Plant model is automatically instantiated from metamodel with data from interdisciplinary model

- Also Domain Specific Language is possible
Questions