Modeling and Simulating SoC Field Bus Communication with SystemC-AMS

Mohamad Alassir, Julien Denoulet, Olivier Romain, Patrick Garda
Université Pierre et Marie Curie – Paris 6 – EA2385
3 Rue Galilée – B.C. 252 – 94200 Ivry sur Seine, France
Email: mdibassir@hotmail.com, julien.denoulet@upmc.fr, olivier.romain@upmc.fr, patrick.garda@upmc.fr

ABSTRACT

This paper focuses on field bus communication modeling based on SystemC-AMS. Nowadays, the design of embedded systems is a key issue for the electronics industry. Many ongoing research projects address this issue by increasing the design abstraction level from the circuit to the system level. However, most embedded systems gather analog and digital electronics with embedded processor cores running large amounts of embedded software. Therefore, neither VHDL-AMS nor SystemC are well-suited to model the architecture of these systems.

Most embedded systems are built out of a set of nodes connected through field busses such as I²C or CAN. These nodes are typically mixed analog and digital SOC, including one or several processor cores and an array of peripheral interfaces. In its most usual form, the node is simply a micro-controller, connected to various sensors or actuators. Automotive electronics provide a number of such embedded systems.

In this paper, we show how to model the field bus communications between the nodes of an embedded system through the example of an I²C bus. Still, our approach is meant to be generic and could be applied for other protocols, for instance CAN.

After briefly summarizing the key features of the I²C protocol, we will describe the SystemC-AMS model of an I²C bus controller IP introduced in [1]. This model is made of two blocks: a digital block, modeled in SystemC and interfaced with a master microprocessor, manages the protocol, timing and control of specific sequences while an analog block, written in SystemC-AMS ensures the access to the external bus and models its behavior. Then we will show how this IP can be included into two kinds of embedded systems nodes: on the one hand interfaced with a 8051 micro-controller SystemC model, on the other hand a SOC using a modeling platform.

Interfacing the controller with a 8051 micro-controller allows us to build a basic mixed simulation platform to execute a C compiled code and observe the resulting commands on the analog lines of the I²C bus. To design a SOC node model, we used the SoCLib prototyping platform [2] to create a SoC that includes a MIPS microprocessor, RAM memory, and a VCI interconnect bus where we plugged our controller IP to communicate with the master. To completely validate the I²C protocol (including multi-master arbitration), we finally connected the MIPS to the 8051 through the I²C bus so they can exchange data on an I²C external memory (Fig.1).

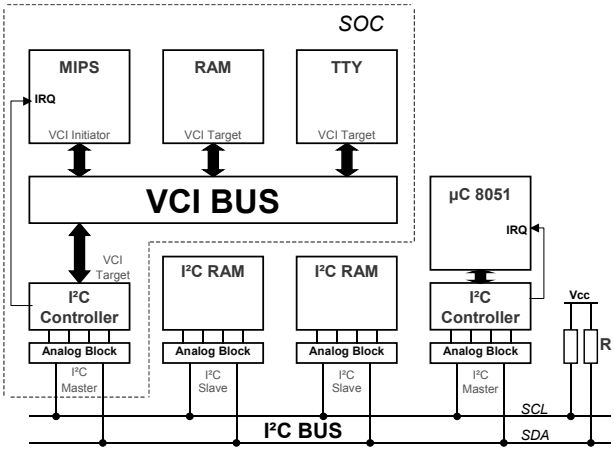


Fig.1 – Simulation platform with SOC node & 8051 node

We'll also present simulation speed results to evaluate the overhead of SystemC-AMS in the global simulation. In the Fig.1 platform (SOC + 8051), AMS simulation of the I²C bus is performed only 10% slower than a digital-only simulation.

REFERENCES

[1] M.Alassir, J.Denoulet, O.Romain, P.Garda : Modelling and Simulation of an I2C Bus Controller in SystemC-AMS, in proceedings of FDL'2006, pp.121-127, Darmstadt, Allemagne, 19-22 Septmbre 2006

[2] SoCLIB. A modelisation & simulation plat-form for system on chip, http://www.soclib.fr/Home.html

Modeling & Simulating SoC Field Bus Communications with SystemC-AMS

M.Alassir, J.Denoulet, O.Romain, P.Garda

Université P. & M. Curie, Paris, France

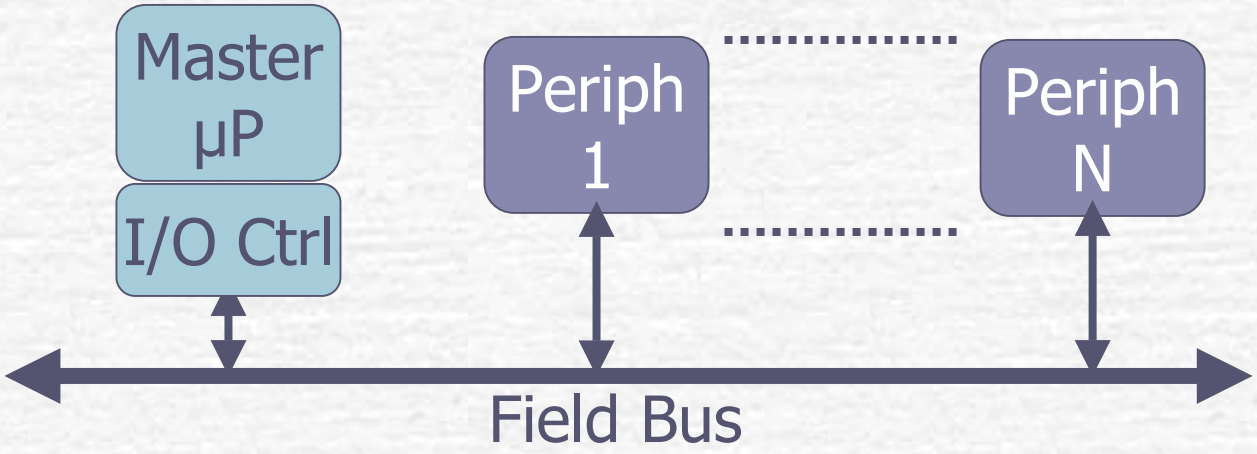
Context

- Design of embedded systems is a key issue for the electronics industry

- Most of these systems include
  - Analog electronics
  - Digital electronics
  - Embedded software.

Automotive Application Example

- Set of nodes connected through a field bus
- Nodes are mixed A/D SOCs, including:
  - Processor cores (µc, µp)
  - Peripheral interfaces (sensors, actuators).



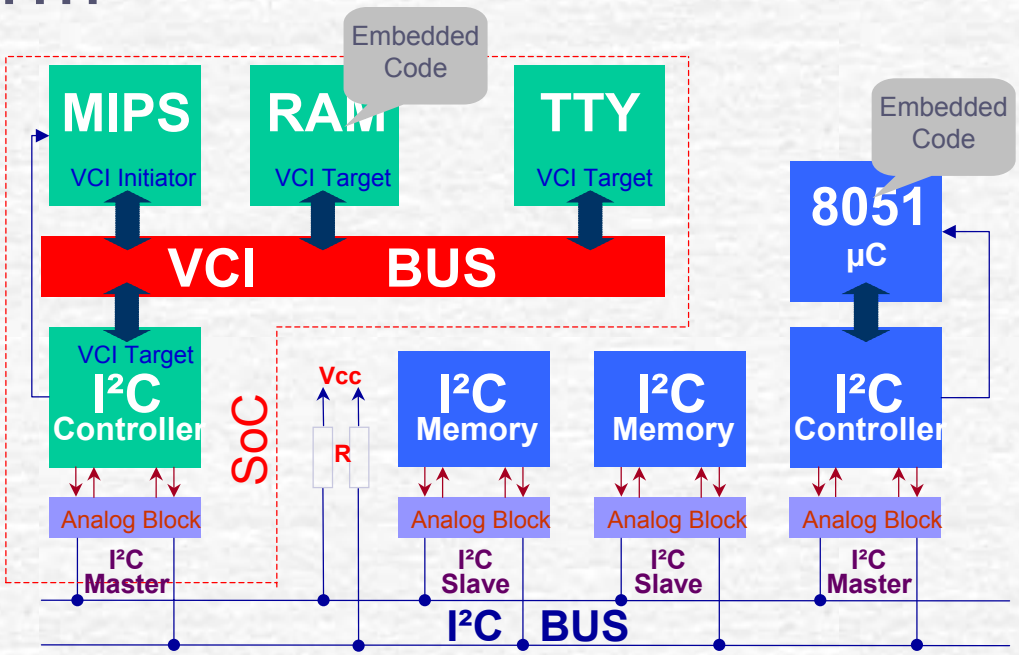
Model & Simulation

- System complexity requires system-level models of platform
- Simulation of A/D Hardware and Software with a single environment

➢ SYSTEMC + SYSTEMC-AMS

Mixed Simulation Platform

- Automotive Context
- Generic platform
- I²C Example



Outline

- I^2C Controller IP
- Modeling Platform
  - Microcontroller Node
  - SoC Node + Microcontroller Node
- Simulation Performances
- Conclusion

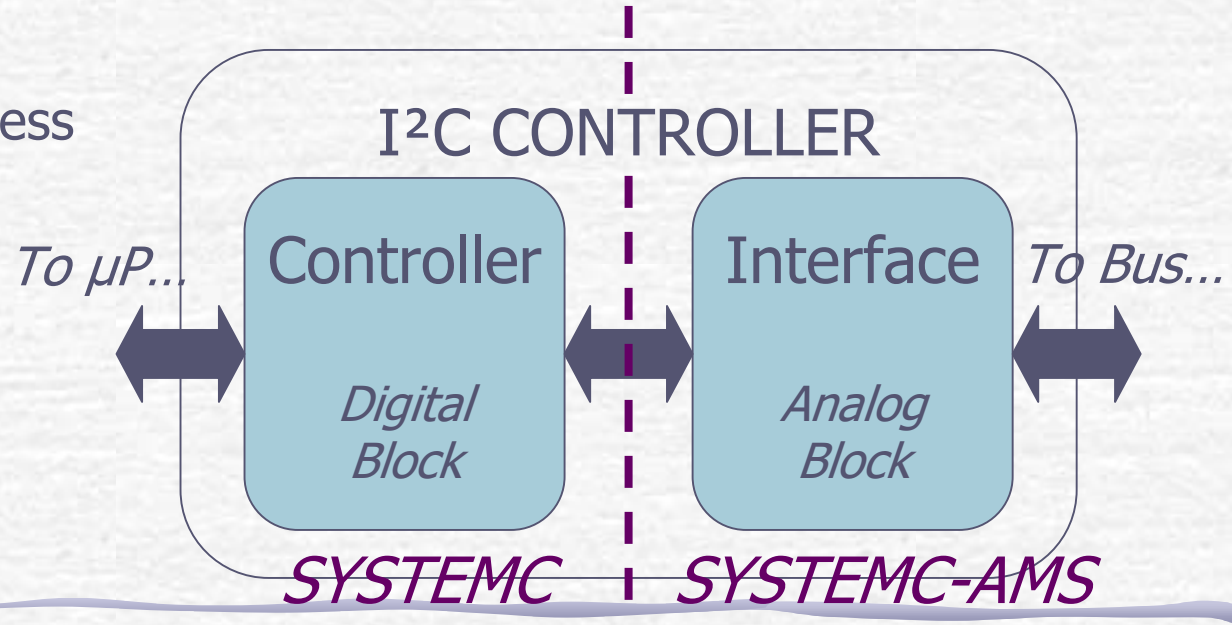
I^2C Protocol

- Serial Transmission
- 2 Bidirectionnal Lines
  - SCL: Clock
  - SDA: Data
- Data Frame

S	Address (7)	R/W	Ack	Data (8)	Ack	Data (nx8)		P

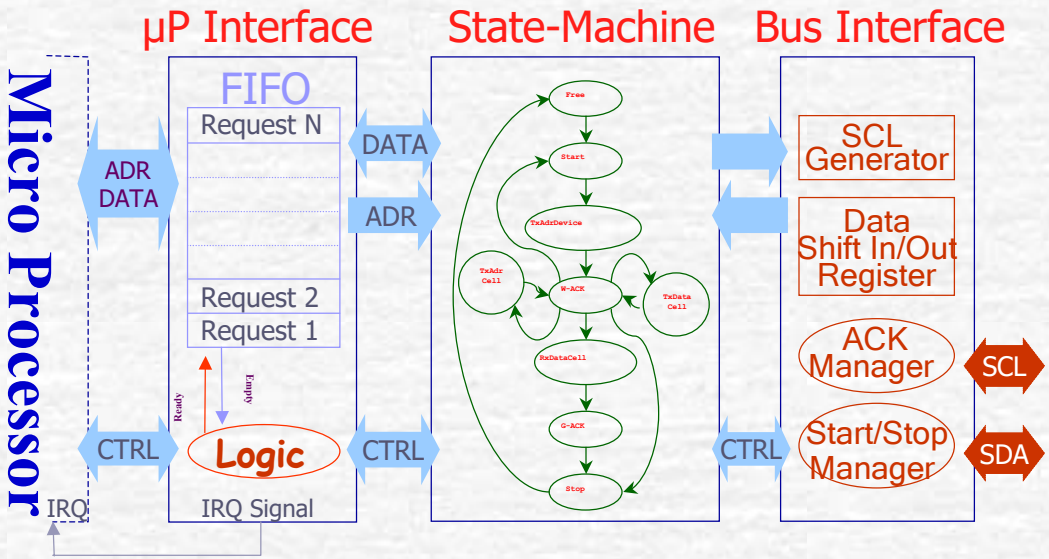
I²C Controller IP

- Mixed IP
  - Digital Block:
    - Protocol, Timing Management, Microprocessor Interface
  - Analog Block
    - External Bus Access



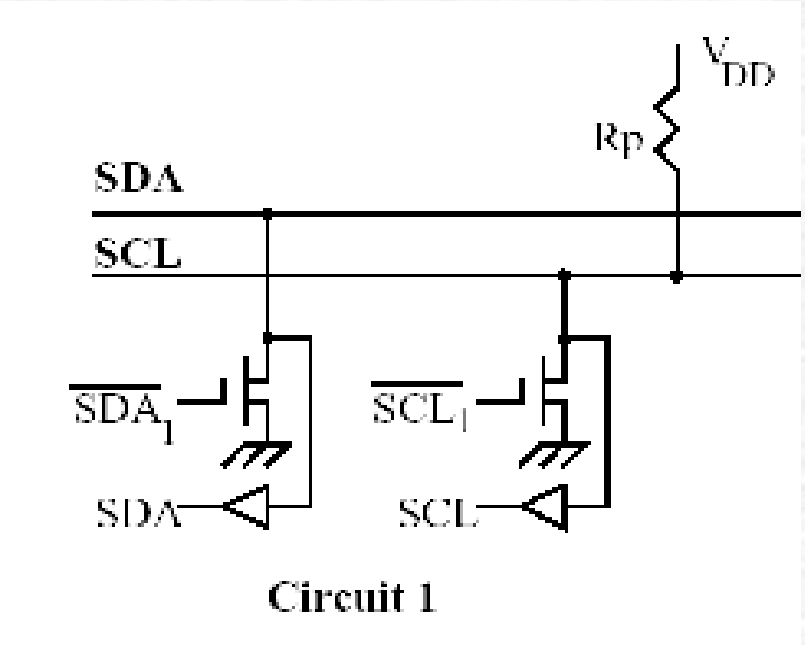
Digital Block Architecture

- Generic Structure
- 3 Blocks
  - µP Interface
  - FSM
  - Bus Interface
- Modelled in
  - SystemC 2.1



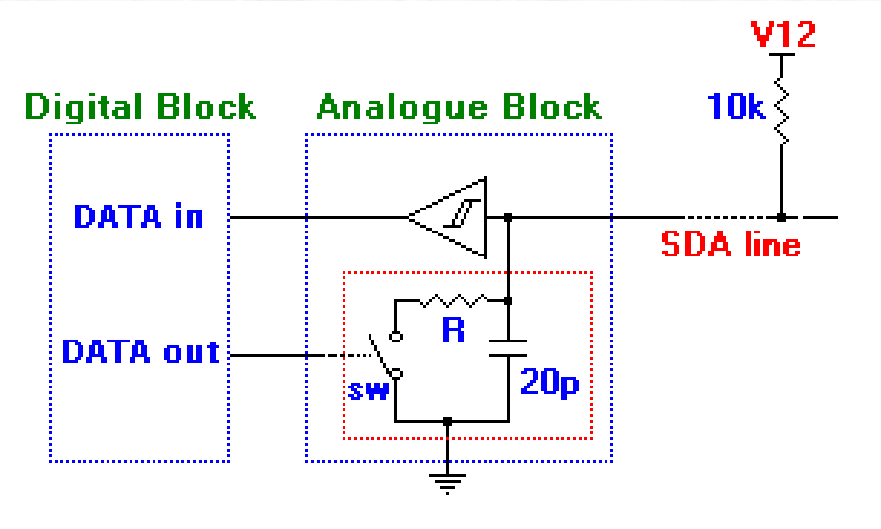
Analog Block

- I^2C Specifications
  - Open-Drain Output
  - Wired-AND Function
  - Pull-up Resistor



Analog Block

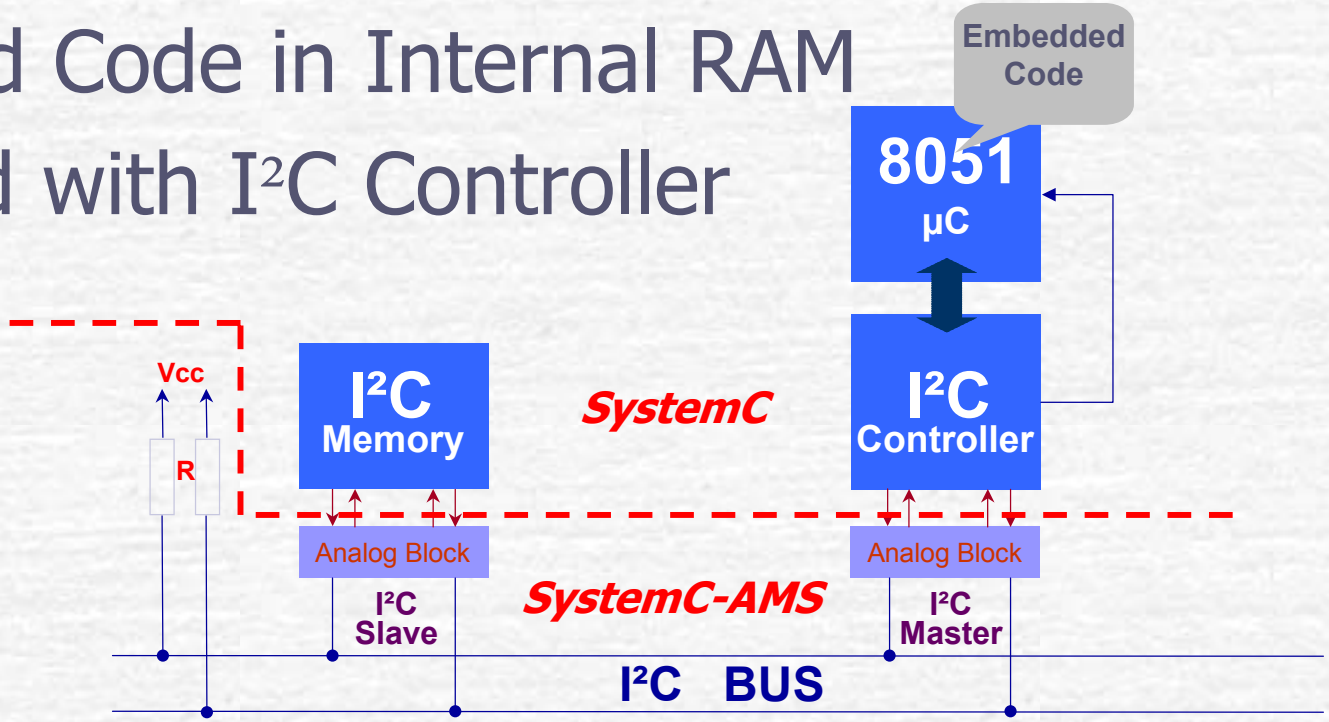
- Analog Interface
- Output
  - "Transistor" model
    - Switch+Resistor
  - Capacitor
    - Rising/Falling Time
- Input
  - Threshold Detection
    - SDF Module



Modelled in SystemCAMS 0.15

Modelling Platform – Case 1

- SystemC-Model of 8051 µC
- Embedded Code in Internal RAM
- Interfaced with I²C Controller



Platform Simulation (1/3)

- Embedded code sends requests to I/O controller
  - Example: Byte Writing Operation

```
main:    MOV     R0, #A0
         MOV     A, #B4
         MOVX    @RO, A
         MOV     R0, #A0
         MOV     A, #4E
         MOVX    @RO, A
```

Platform Simulation (1/3)

- Embedded code sends requests to I/O controller
  - Example: Byte Writing Operation

```
main:   MOV     R0, #A0
        MOV     A, #B4
        MOVX    @RO, A
        MOV     R0, #A0
        MOV     A, #4E
        MOVX    @RO, A
```

Slave Adress + Write Command

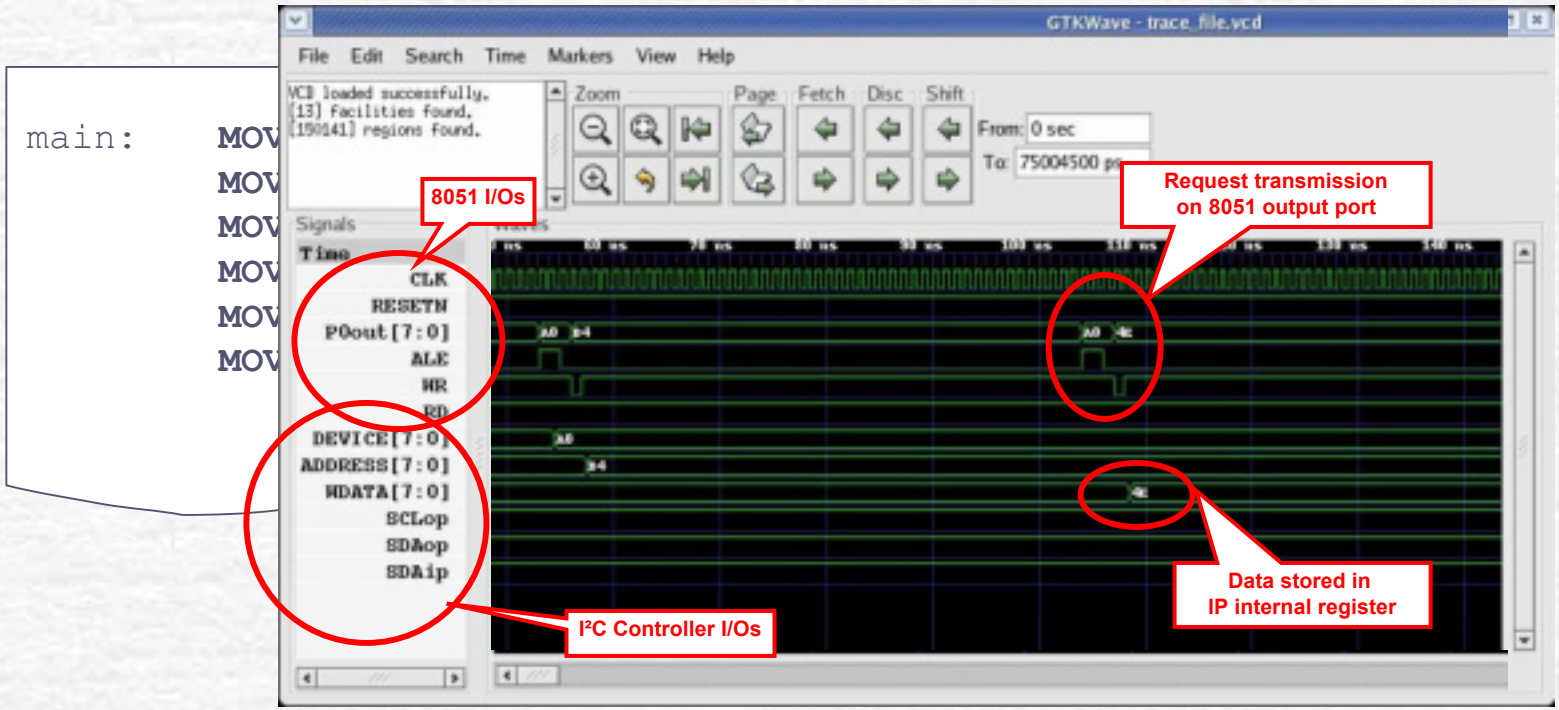
Cell Adress

Data Byte

Request Transmission

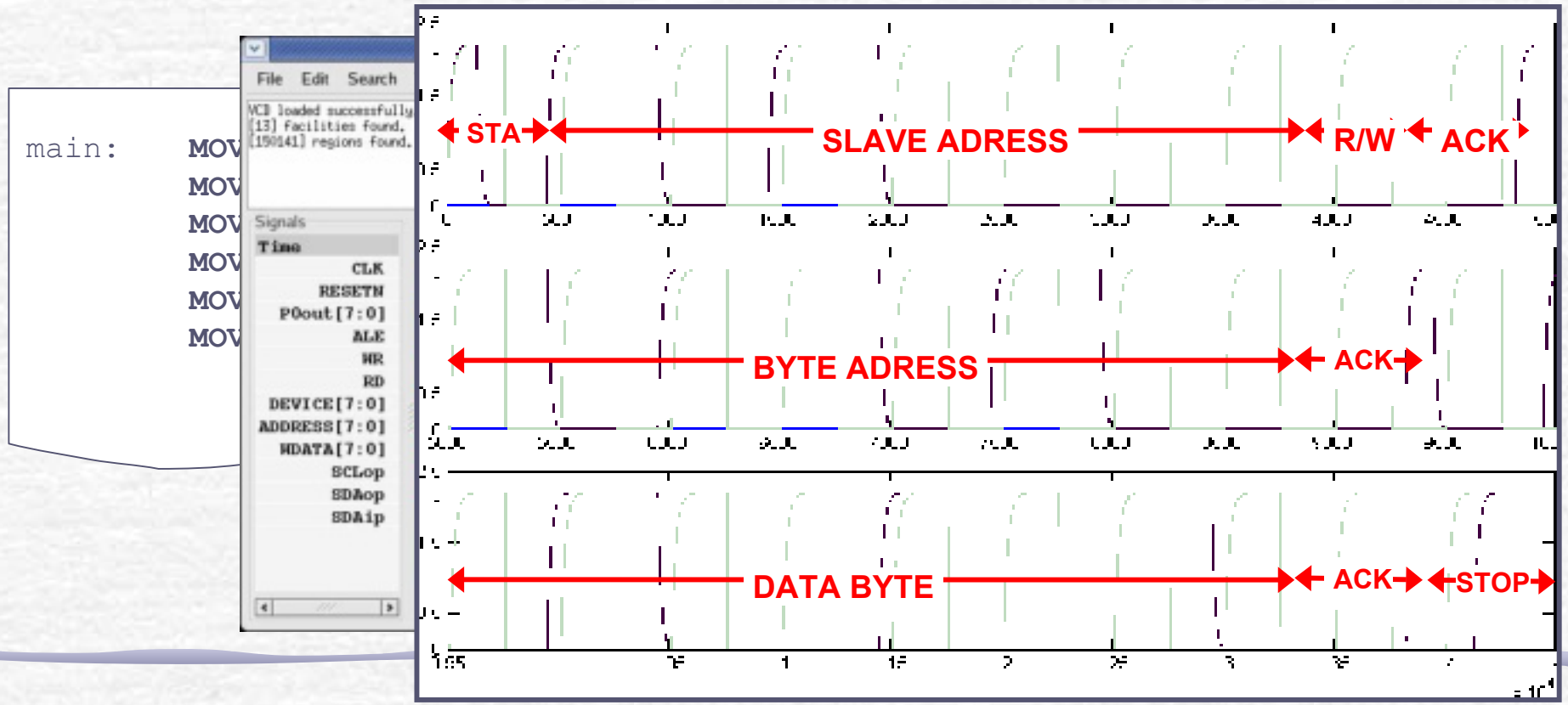
Platform Simulation (2/3)

- Requests are processed by I/O Controller



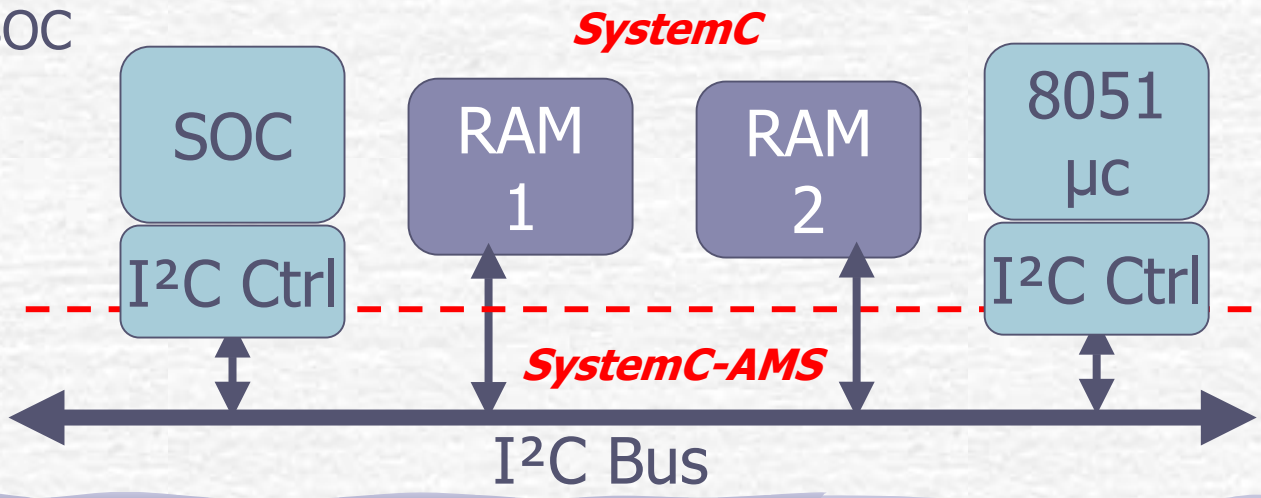
Platform Simulation (3/3)

- Analog Behaviour of the bus lines



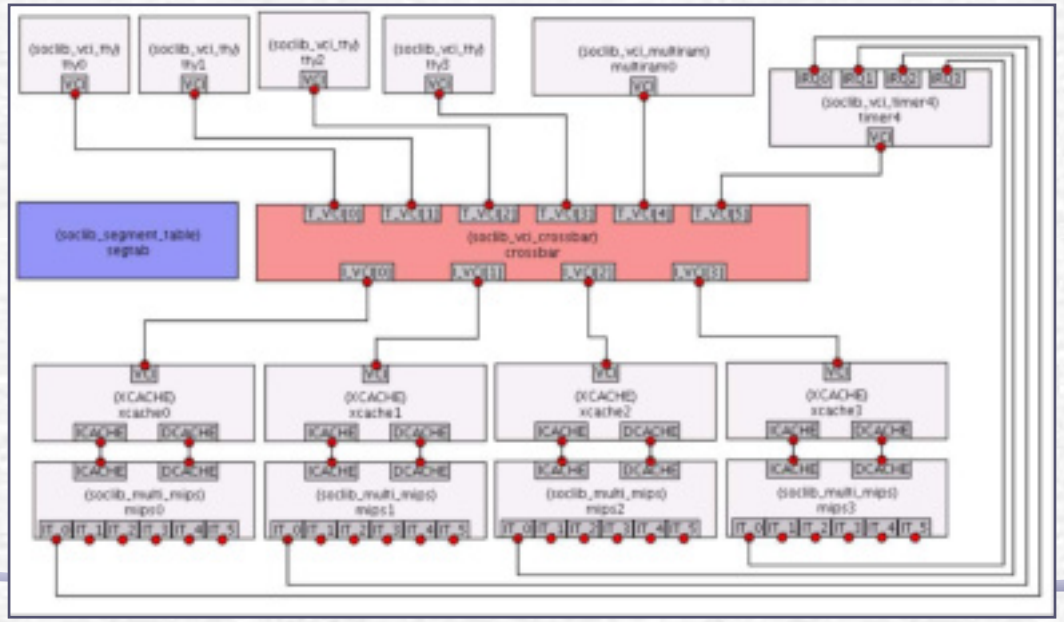
Modelling Platform – Case 2

- Protocol Validation
- Platfom features 4 nodes
  - 2 Masters
    - 8051 Microcontroller
    - MIPS-Based SOC
  - 2 Slaves
    - I²C RAM



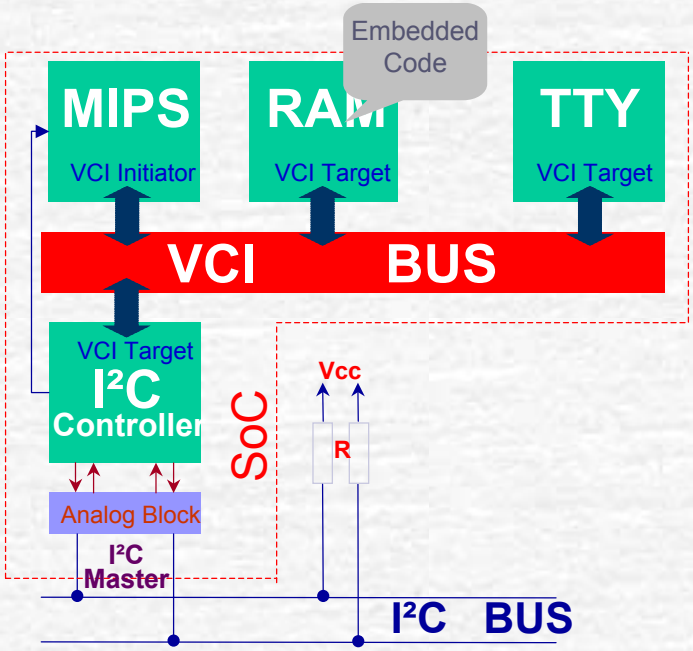
SoC Node

- SoCLib: SoC Prototyping Platform
  - Started in 2007 - 17 partners (11 academic, 6 industrial)
  - Based on SystemC IP Library

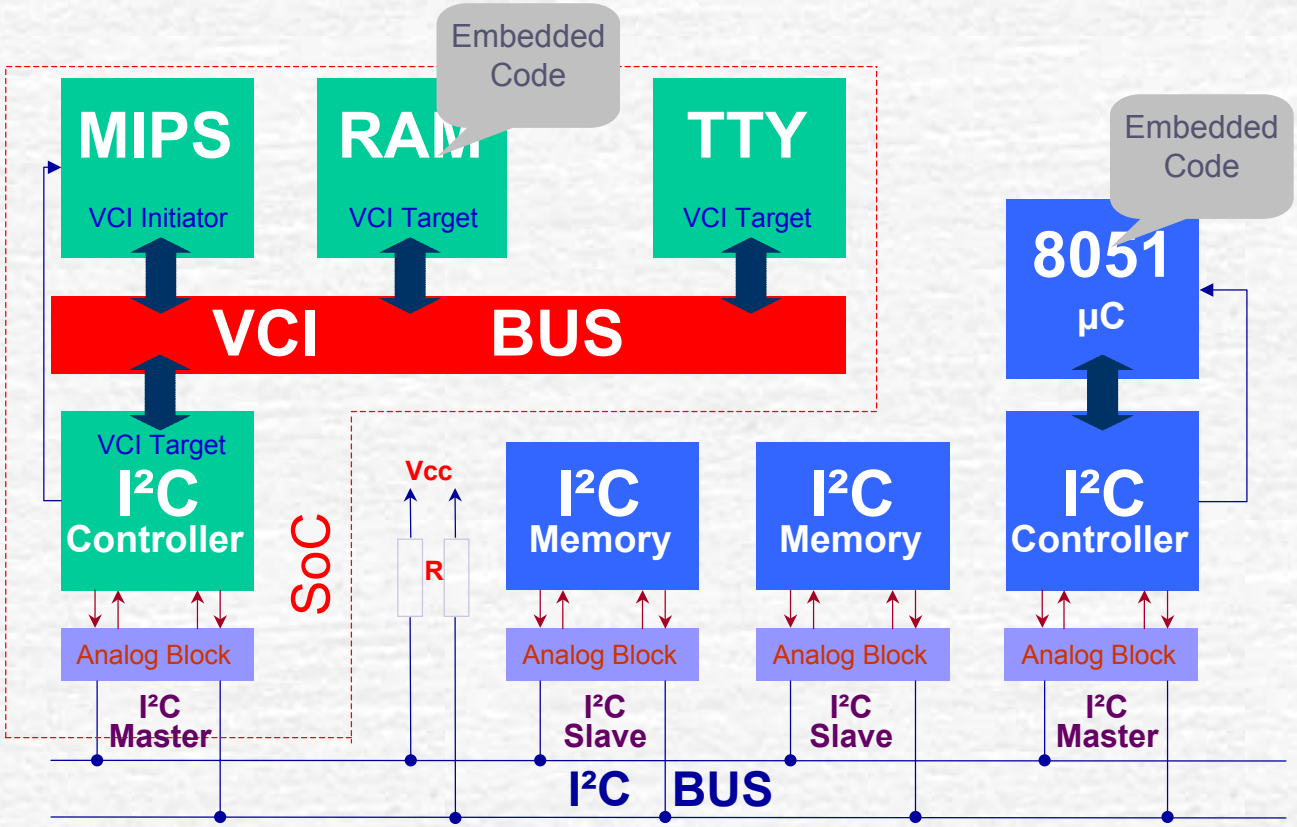


SoC Node

- SoCLib: SoC Prototyping Platform
  - Integration of our Controller to the IP library
  - Development of a VCI wrapper
  - MIPS Processor
  - RAM with embedded code



Modelling Platform – Case 2



Platform Simulation (1/3)

- Embedded code in MIPS & 8051

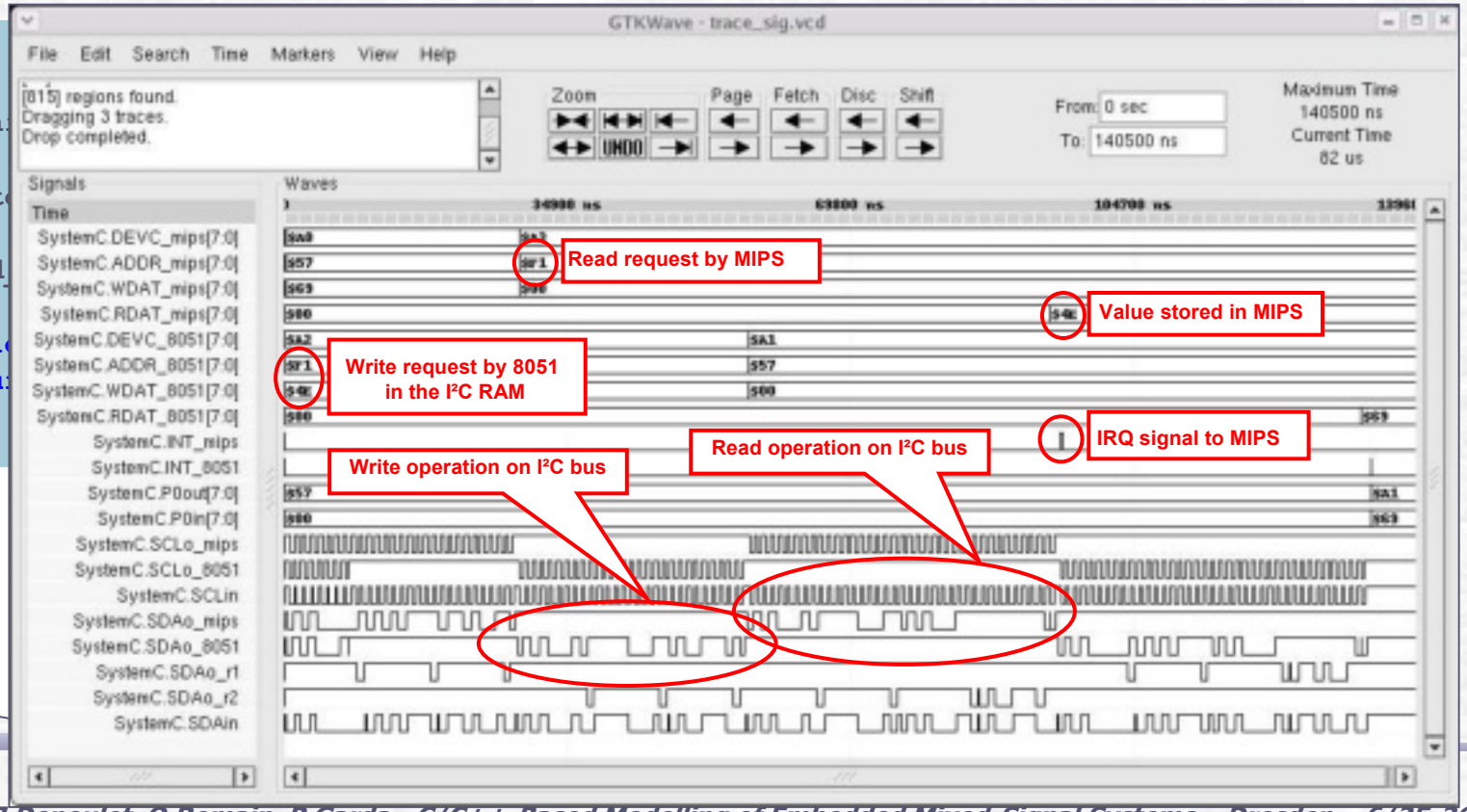
```
int main(void)
{
  write_byte(0x50,0x57,0x69);

  read_byte(0x51, 0xf1);

  while (1);
  return 0;
}
```

```
MOV     R0, #A0
MOV     A, #4E
MOVX    @R0, A
```

Platform Simulation (2/3)

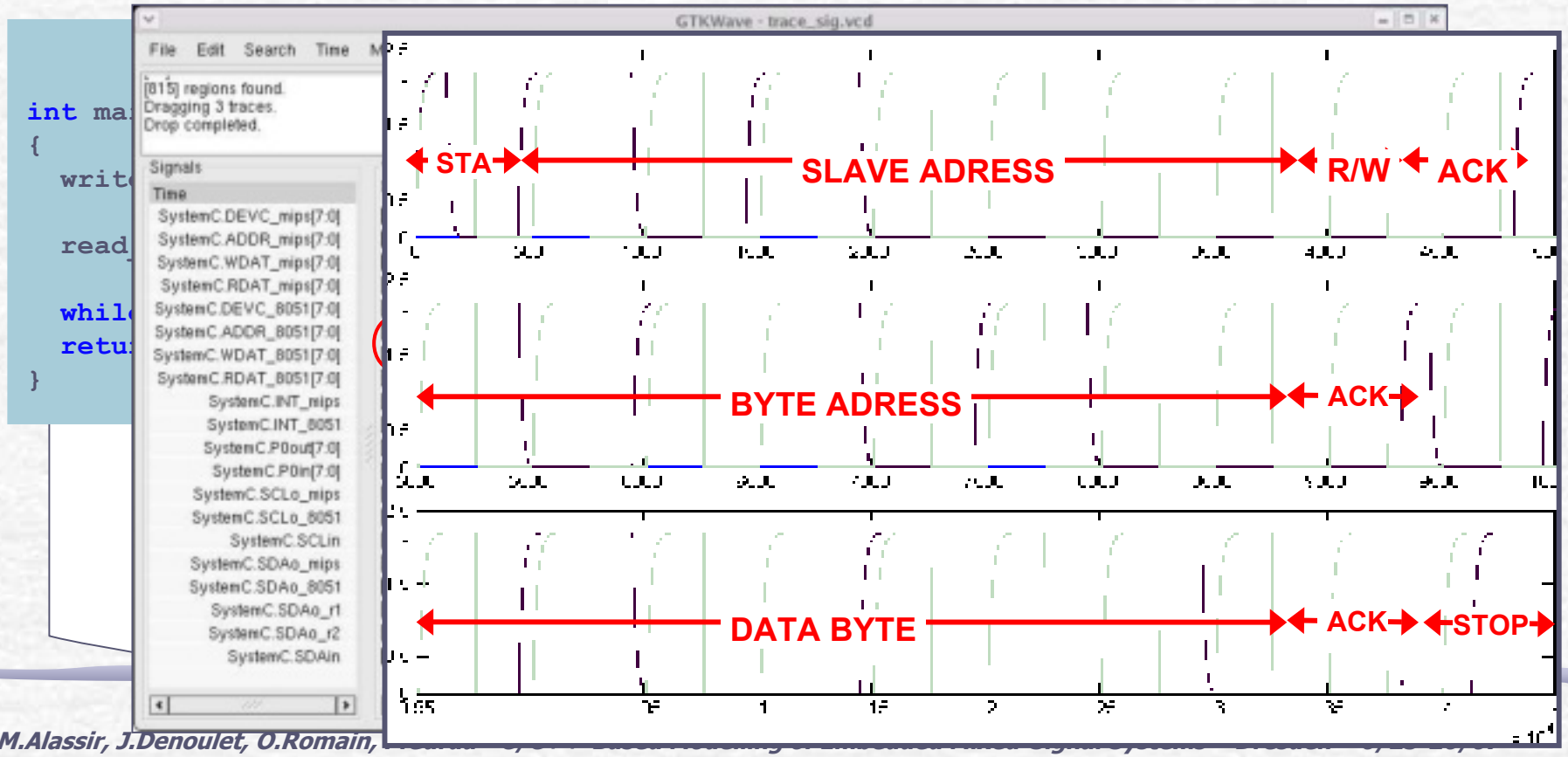
- Communication between MIPS & 8051

```
int ma
{
  writ
  read
  whil
  retu
}
```



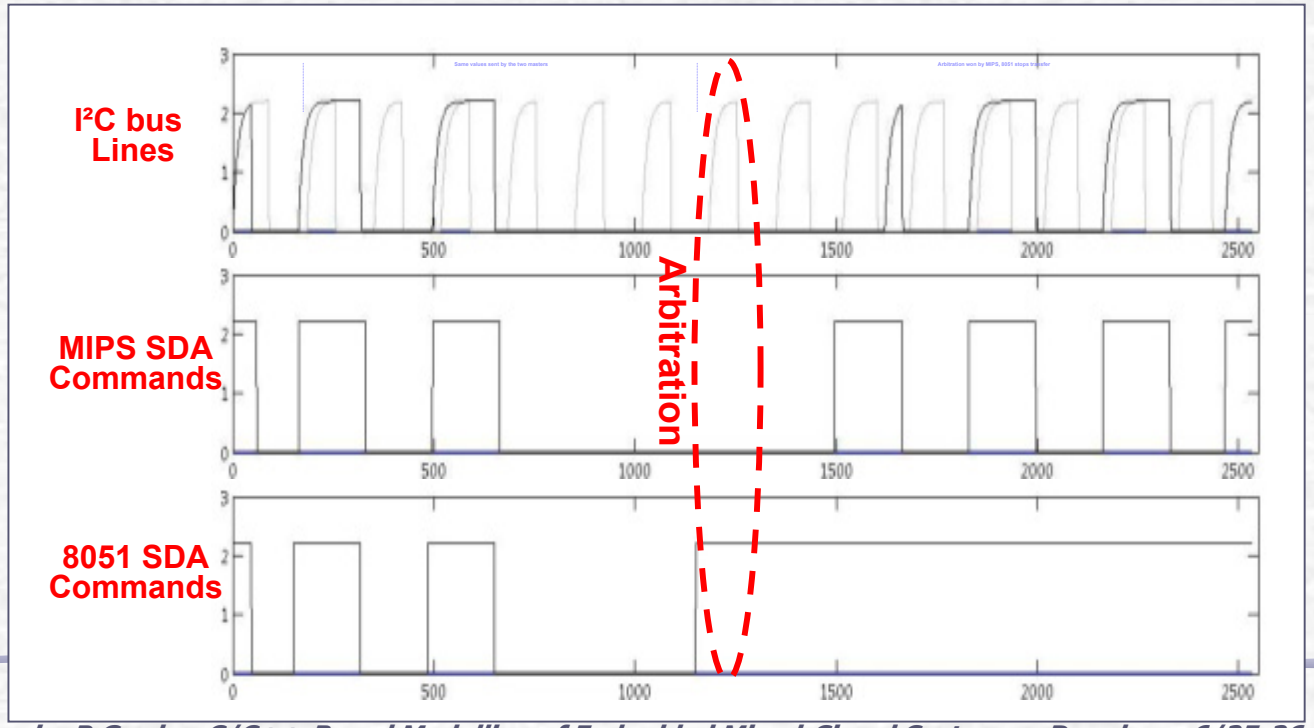
Platform Simulation (3/3)

- Analog behaviour of the bus lines



Multi Master Arbitration

- Wired-AND function
- The node at 0 takes control over the node at 1



Conclusion

- SystemC-AMS modeling platform
  - HW(A/D)-SW mixed simulation
- Generic architecture
  - Appliable for other bus protocols (CAN)
- Analog interface refinement