A new methodology for implementing a distributed clock management system for low-power design

Chadi Al khatib, Mohamed Gana, Laurent Fesquet, Chouki Aktouf

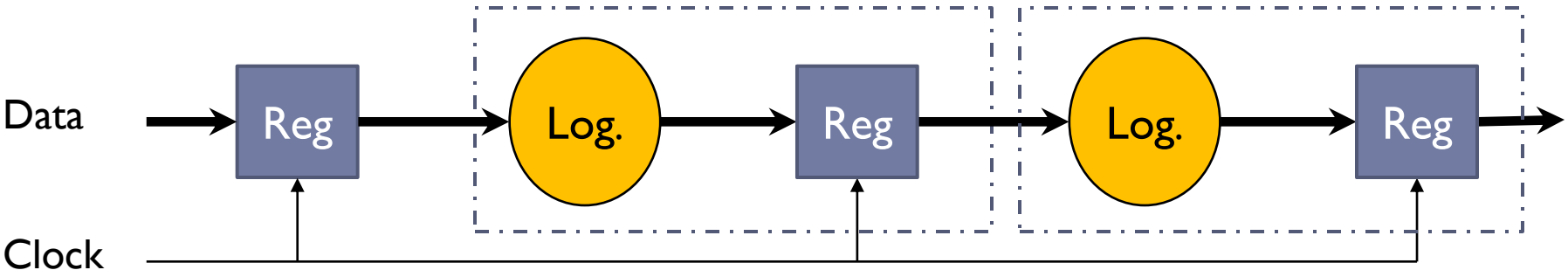
Outline

- Part 1:
  - Synchronous and asynchronous systems
  - Control structure
  - AXI bus
  - Insertion methodology
- Part 2:
  - Automatic insertion for low power structure
  - Results
- Conclusions and perspectives

Outline

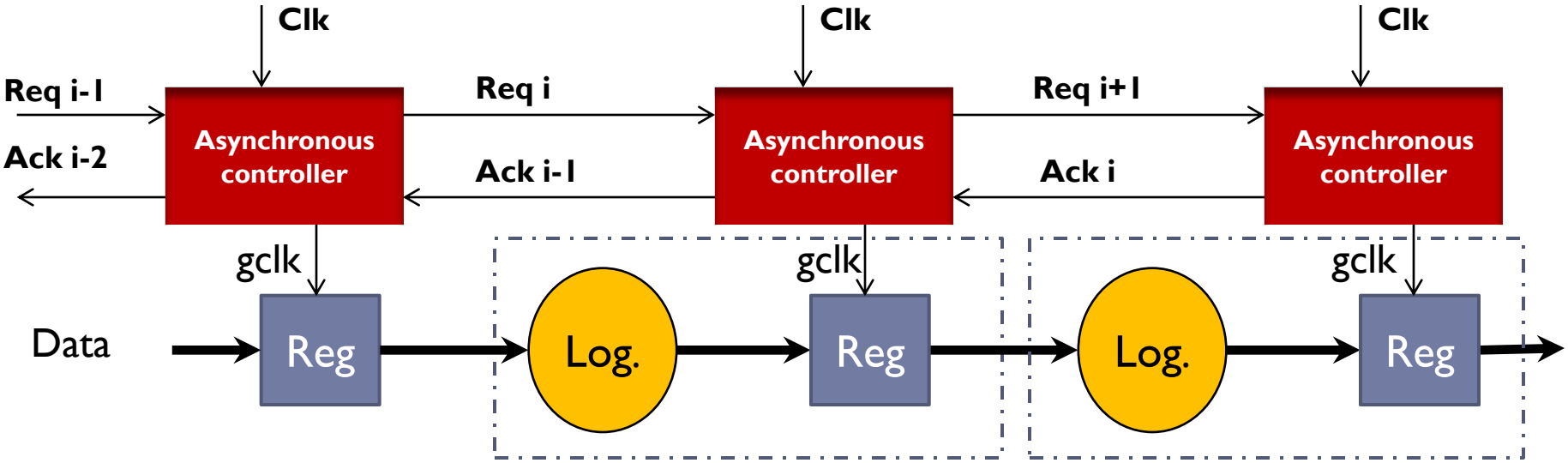
- **Part 1:**
  - **Synchronous and asynchronous systems**
  - **Control structure**
  - AXI bus
  - Insertion methodology
- Part 2:
  - Automatic insertion for low power structure
  - Results
- Conclusions and perspectives

Synchronous Vs. asynchronous systems



At RTL level

Synchronous Vs. asynchronous systems

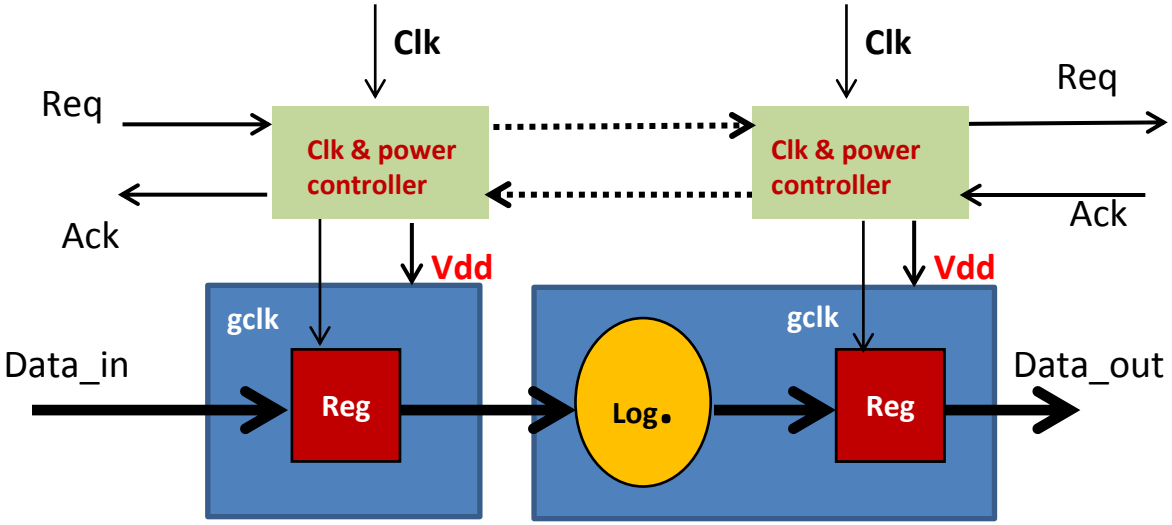


- Local control strategy
- Automatically inserted

At RTL level

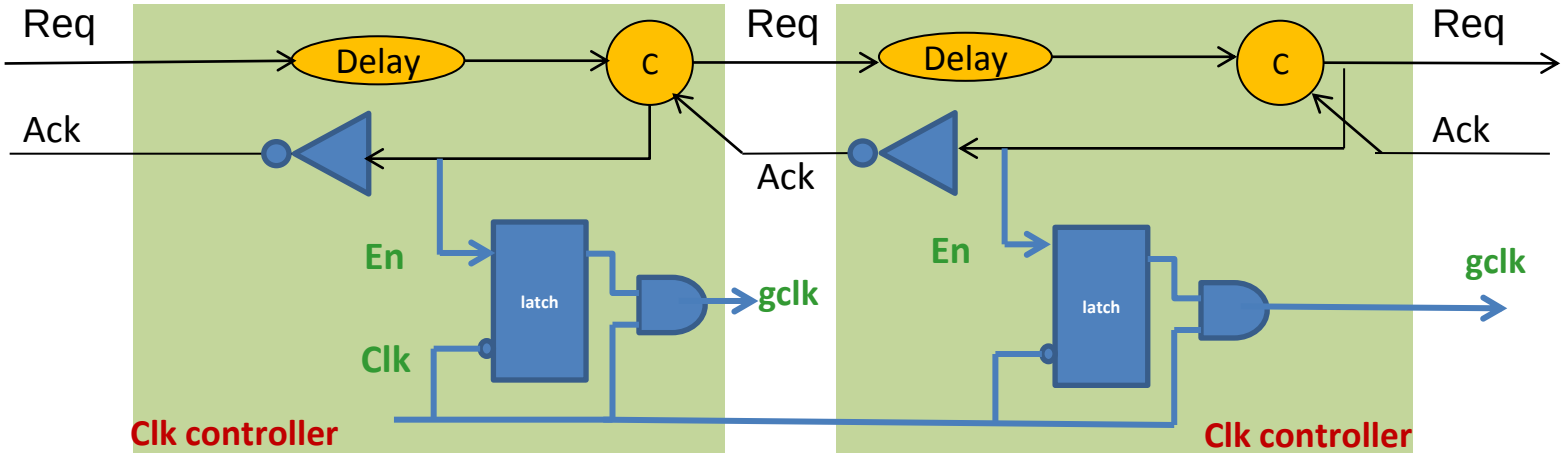
- Reduce power consumption **(≈ 40%)**
- Find new technics, validate methodology of insertion to meet the industry requirements
- Automated implementation procedures using EDA tools → makes the system more robust and avoid the errors of manual connections.

Control structure : Clk & power controller



Muller gate

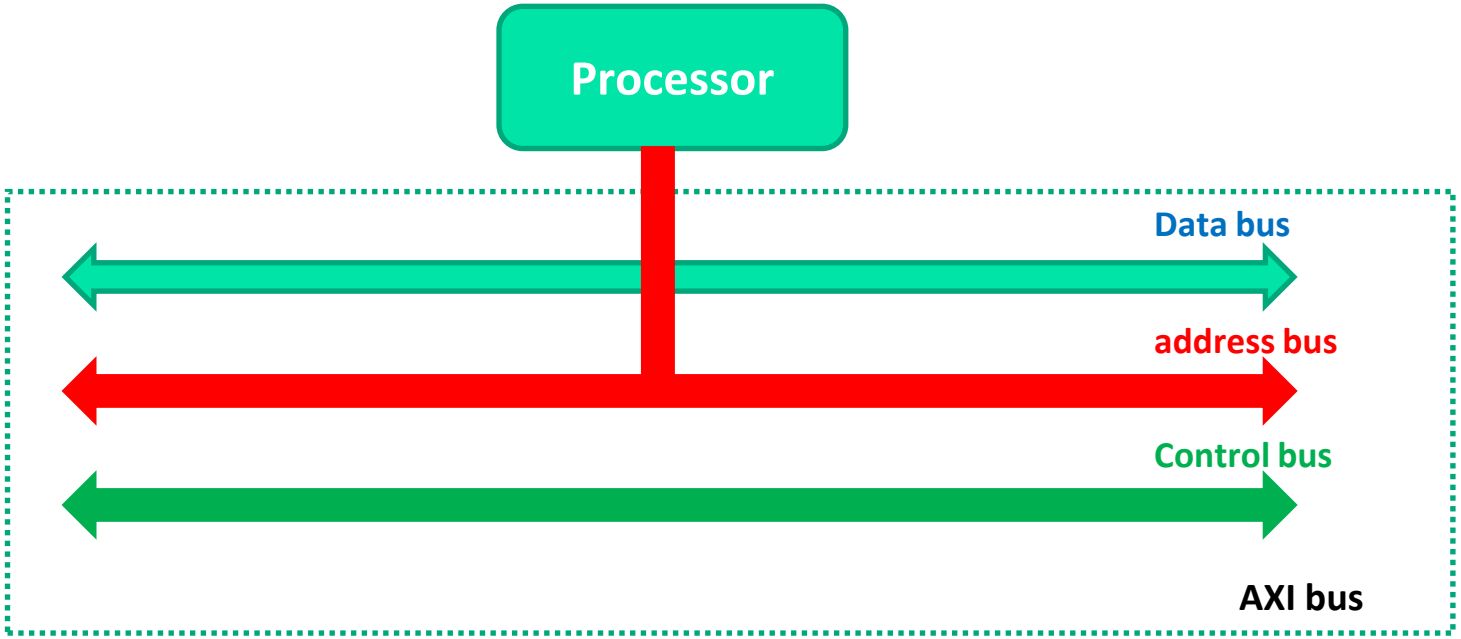
A	B	S
0	0	0
0	1	S^{-1}
1	0	S^{-1}
1	1	1



Outline

- **Part 1:**
  - Synchronous and asynchronous systems
  - Control structure
  - **AXI bus**
  - Insertion methodology
- Part 2:
  - Automatic insertion for low power structure
  - Results
- Conclusions and perspectives

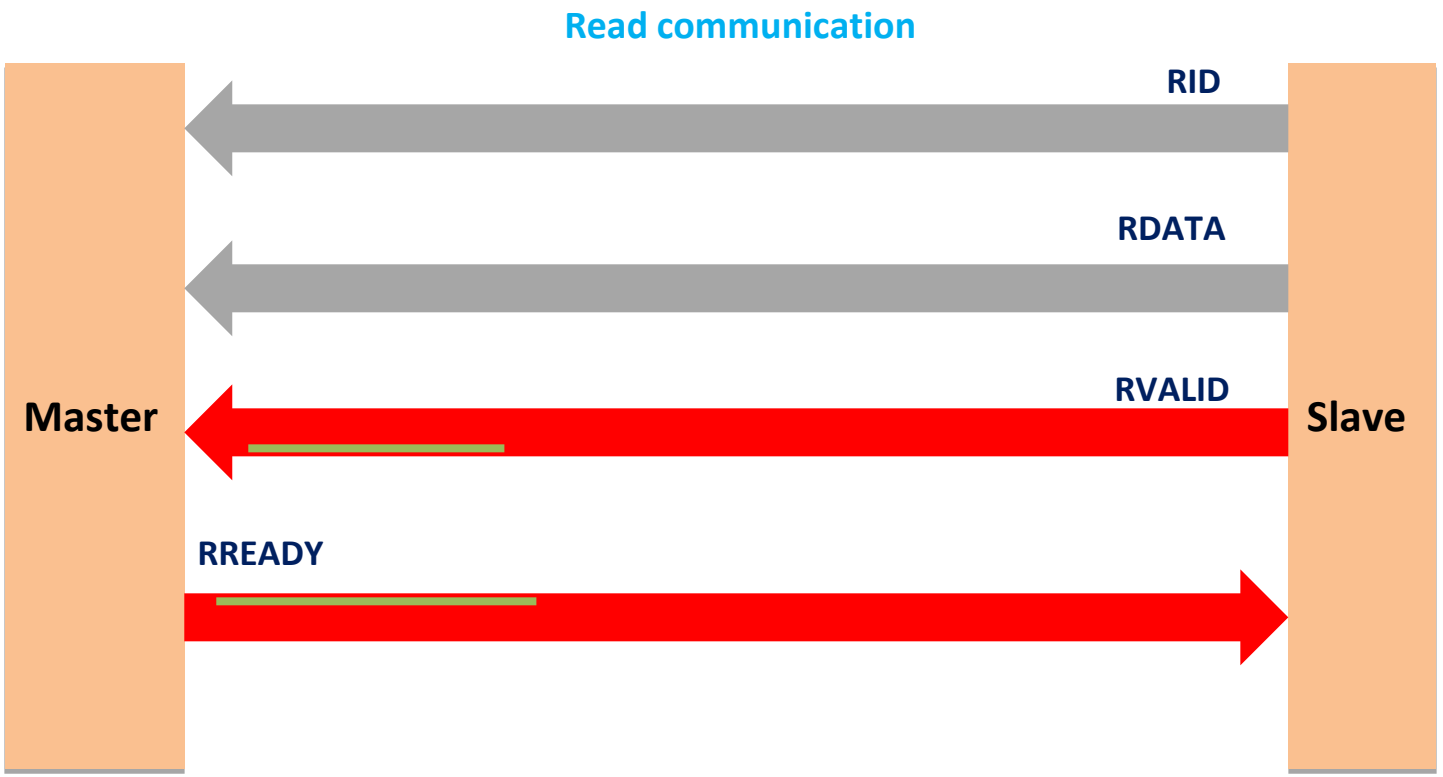
AXI bus presentation



1. Separate address, control and data
2. Separate read and write data channel
3. Support multiple outstanding transactions

Write/Read communication(1/2)

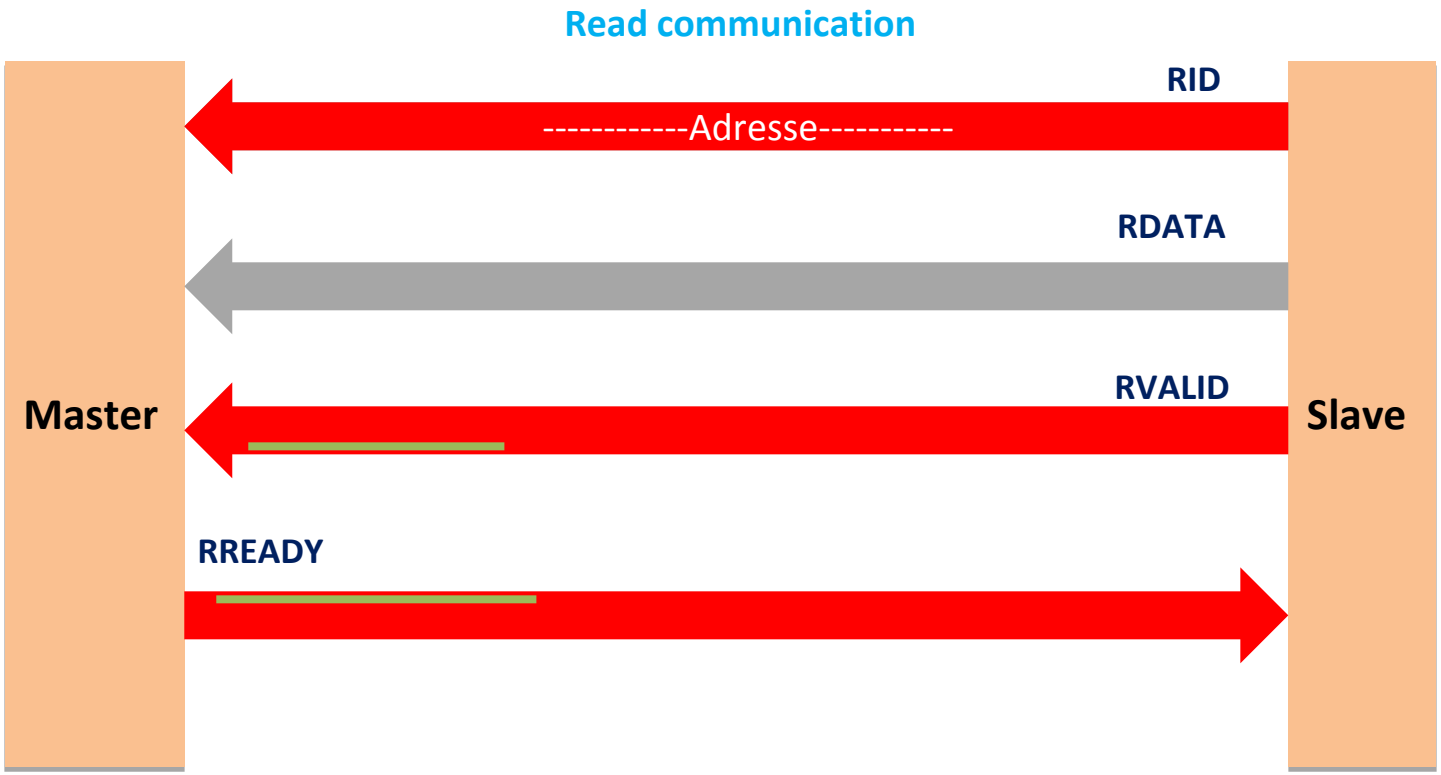
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(1/2)

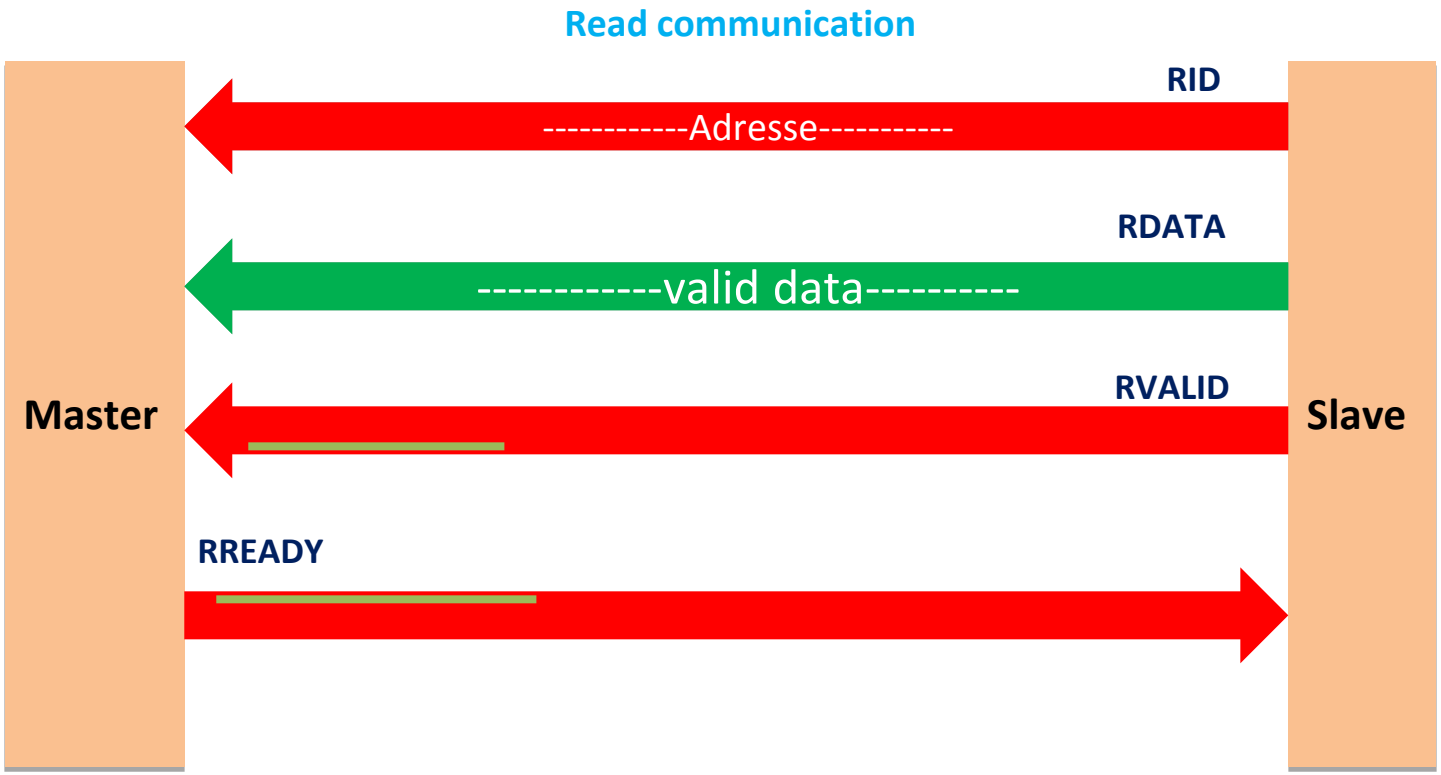
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(1/2)

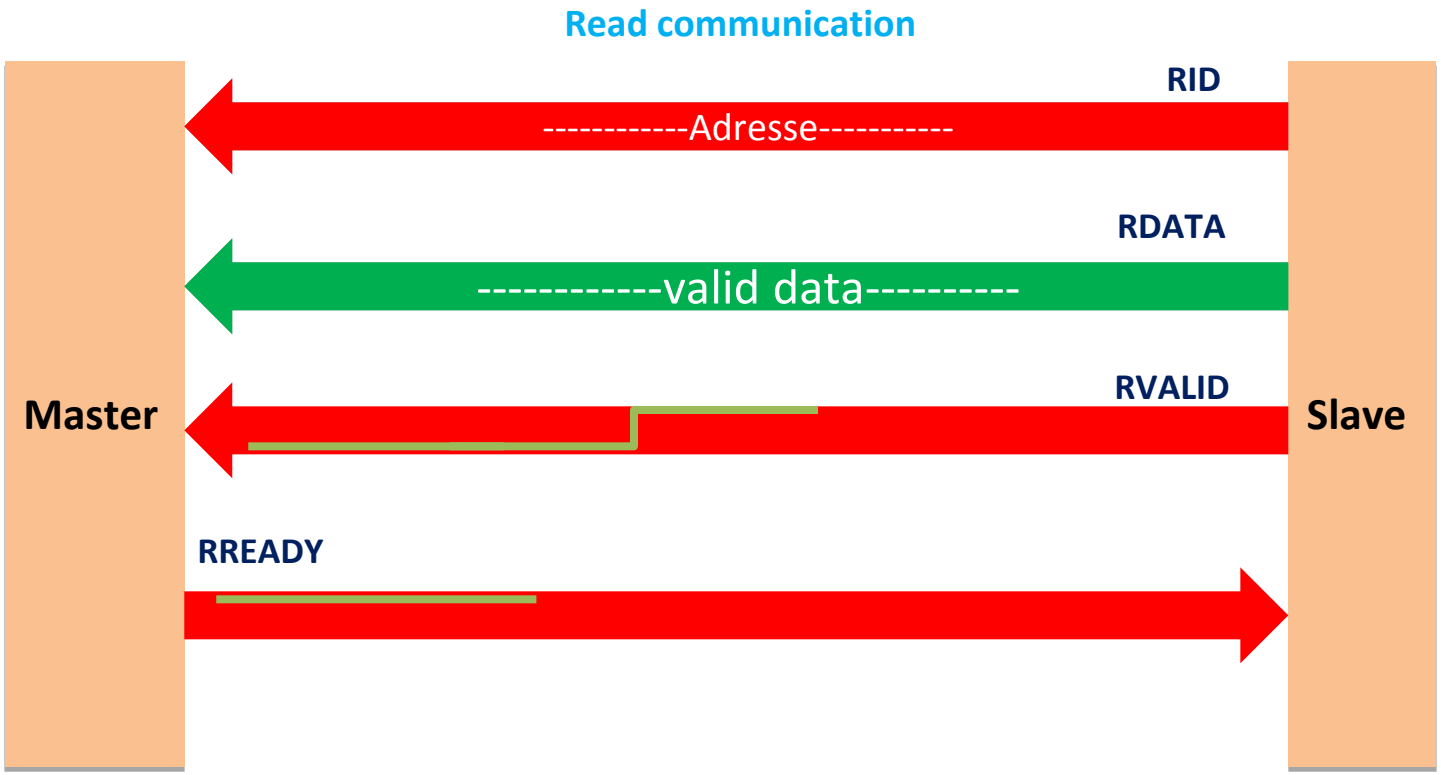
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(1/2)

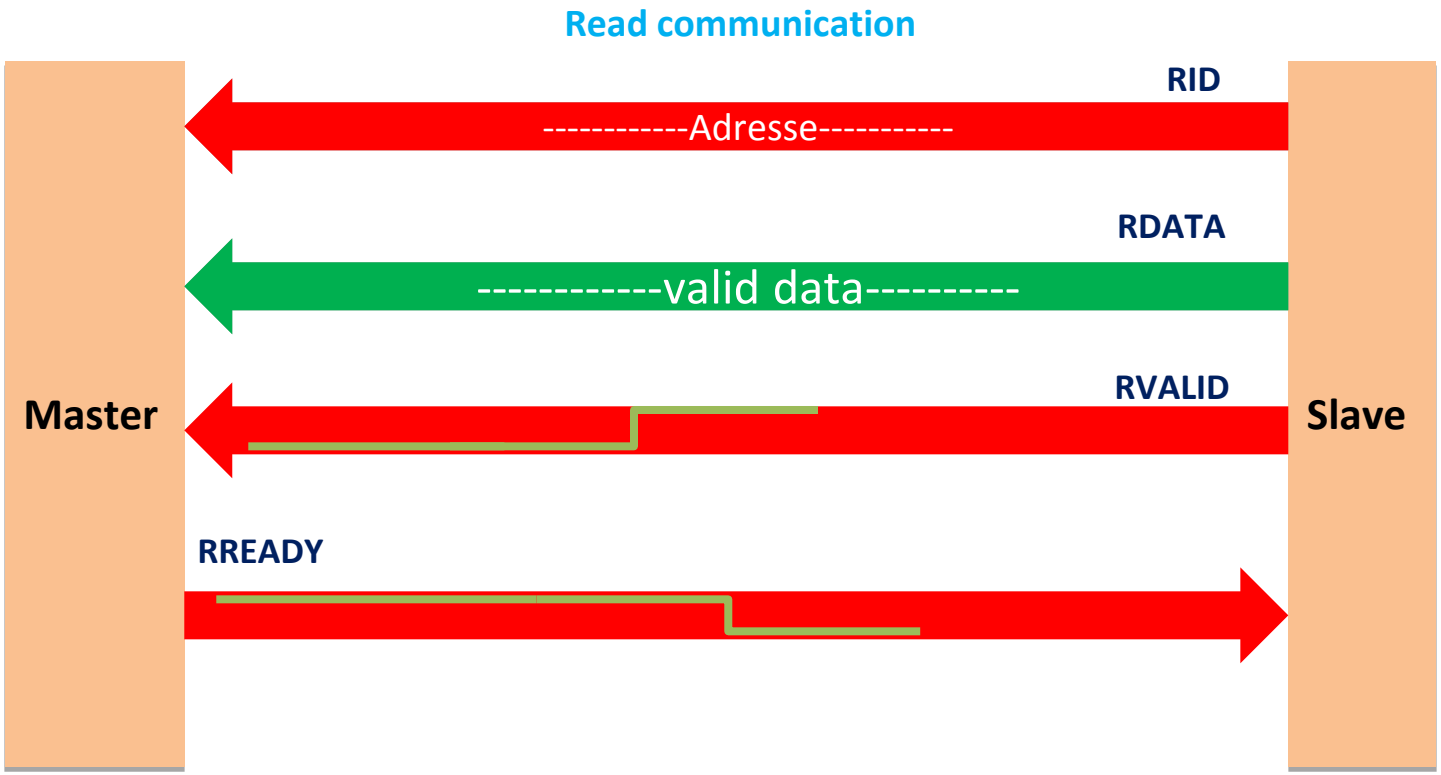
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(1/2)

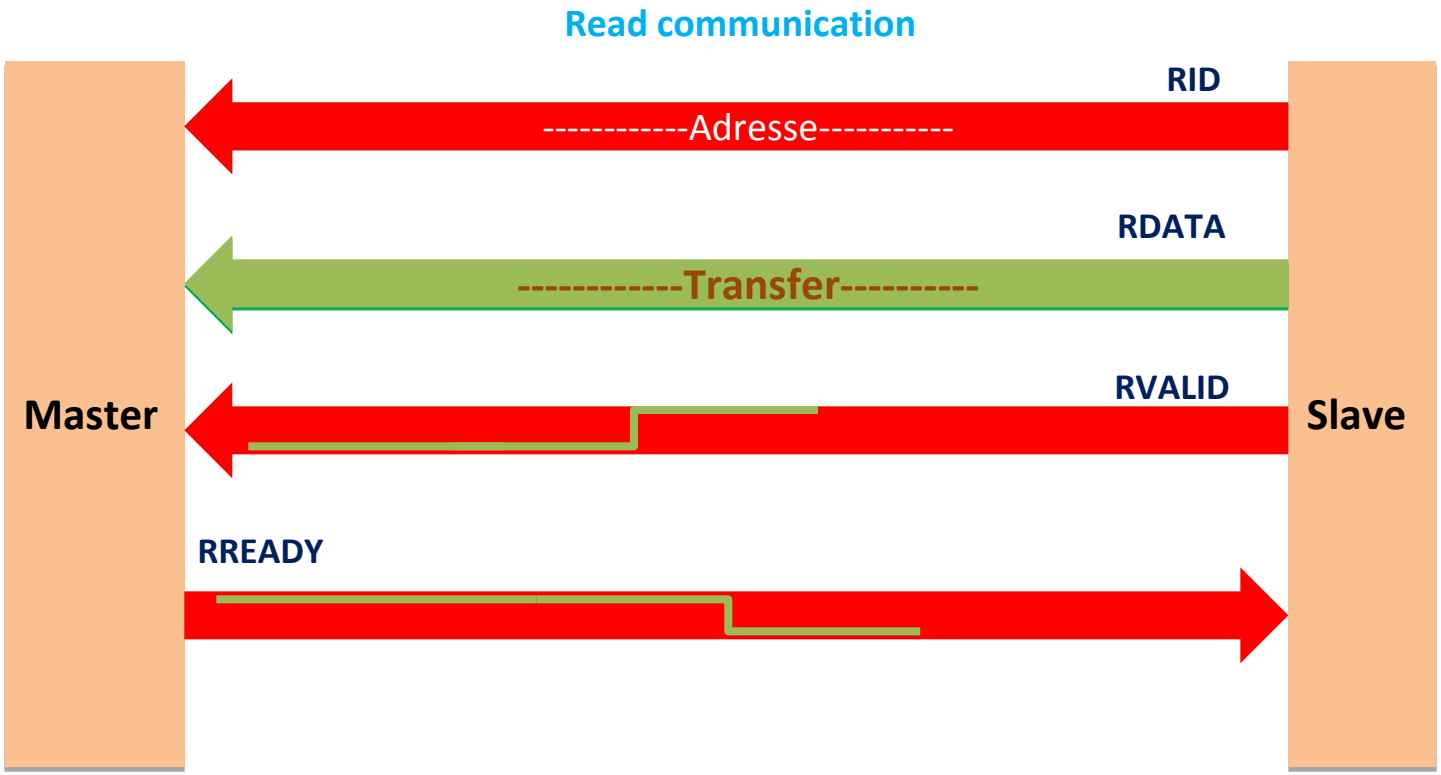
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(1/2)

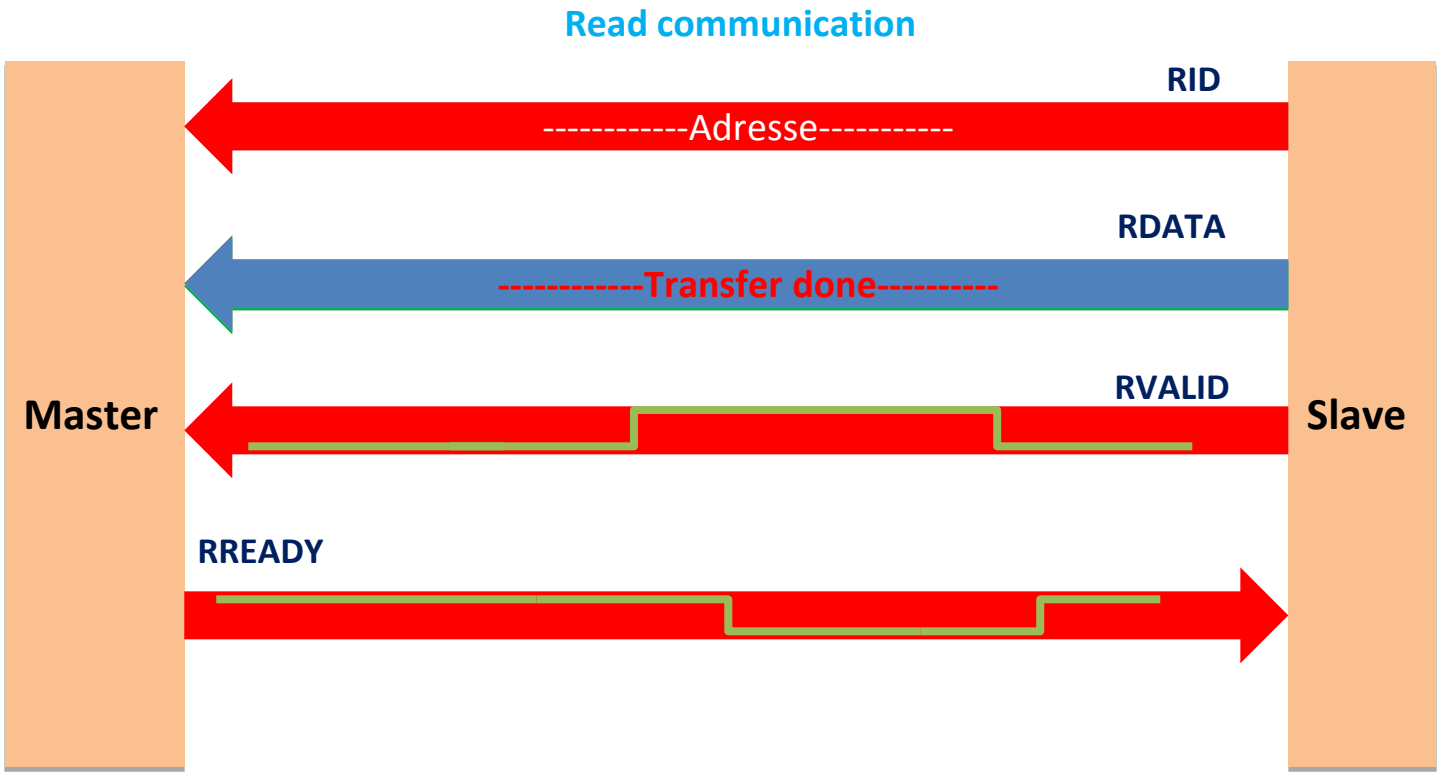
Operation: Master-Slave



Rvalid : Request signal
Rready : Acknowledgement signal

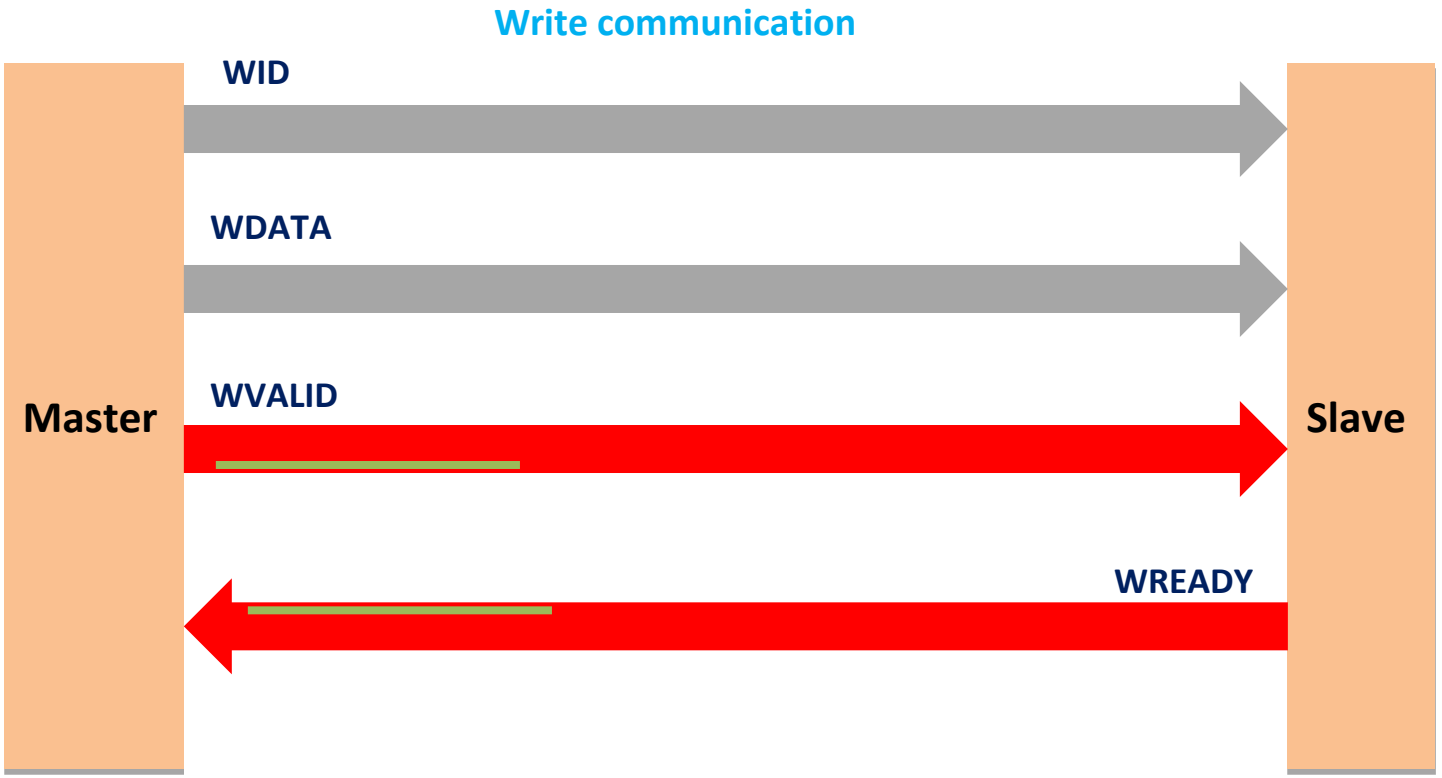
Write/Read communication(1/2)

Operation: Master-Slave



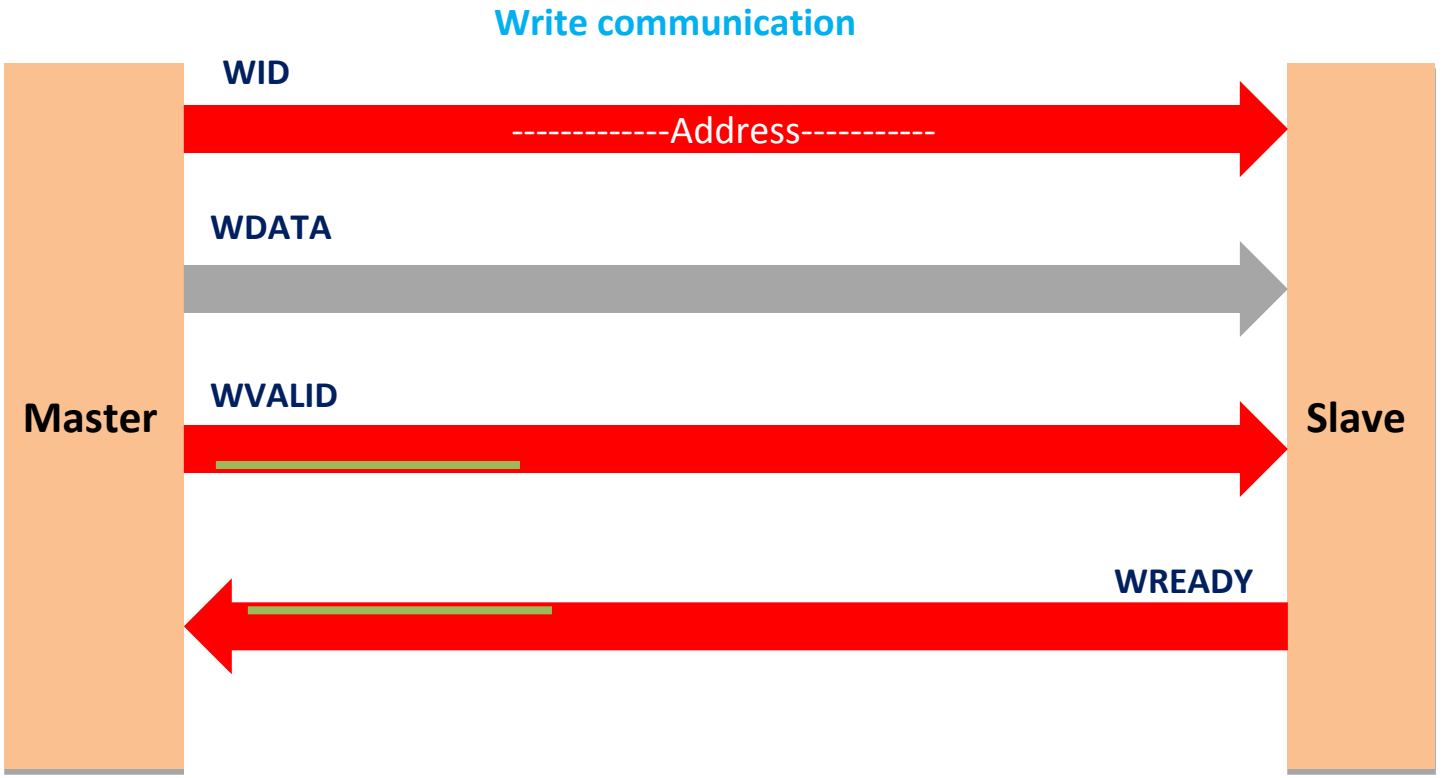
Rvalid : Request signal
Rready : Acknowledgement signal

Write/Read communication(2/2)



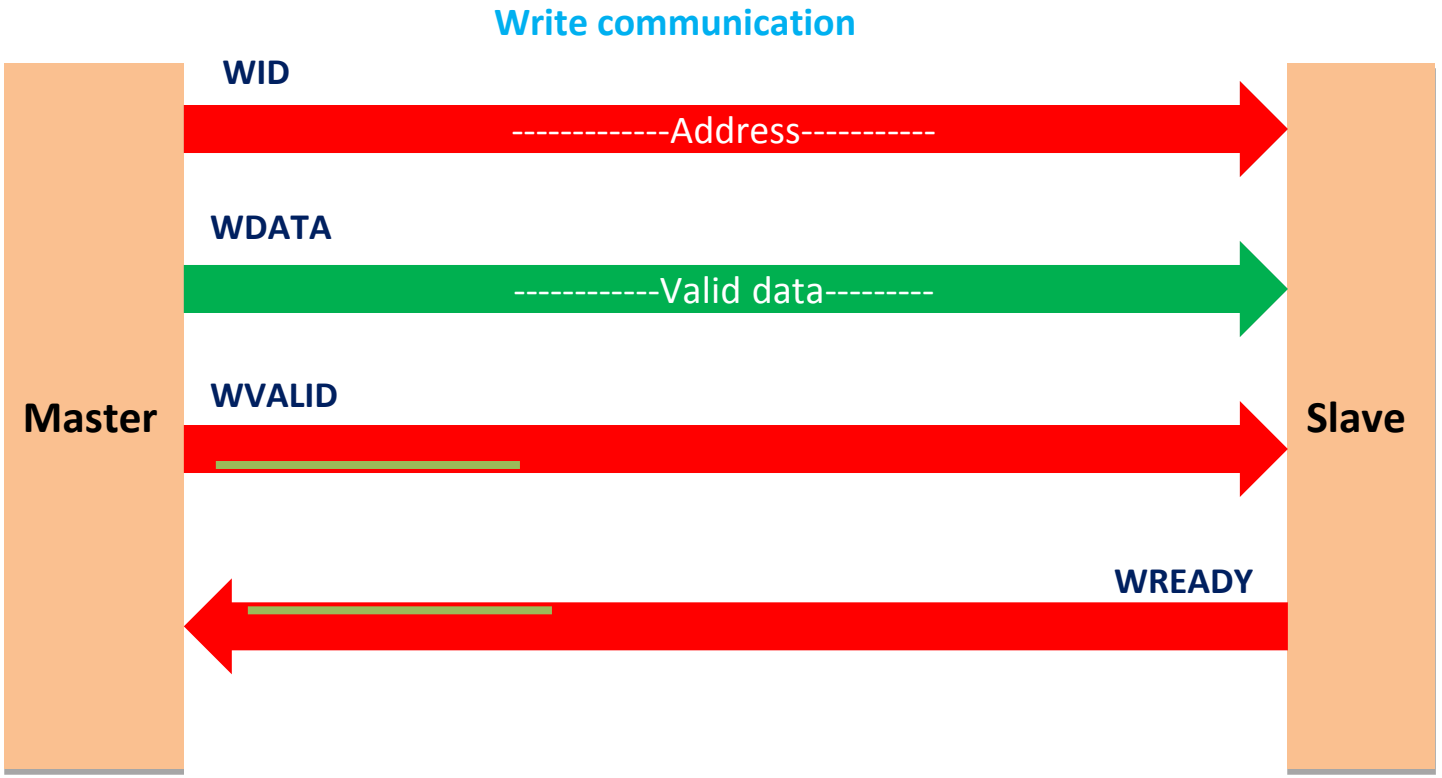
Wvalid	: Request signal
wready	: Acknowledgement signal

Write/Read communication(2/2)



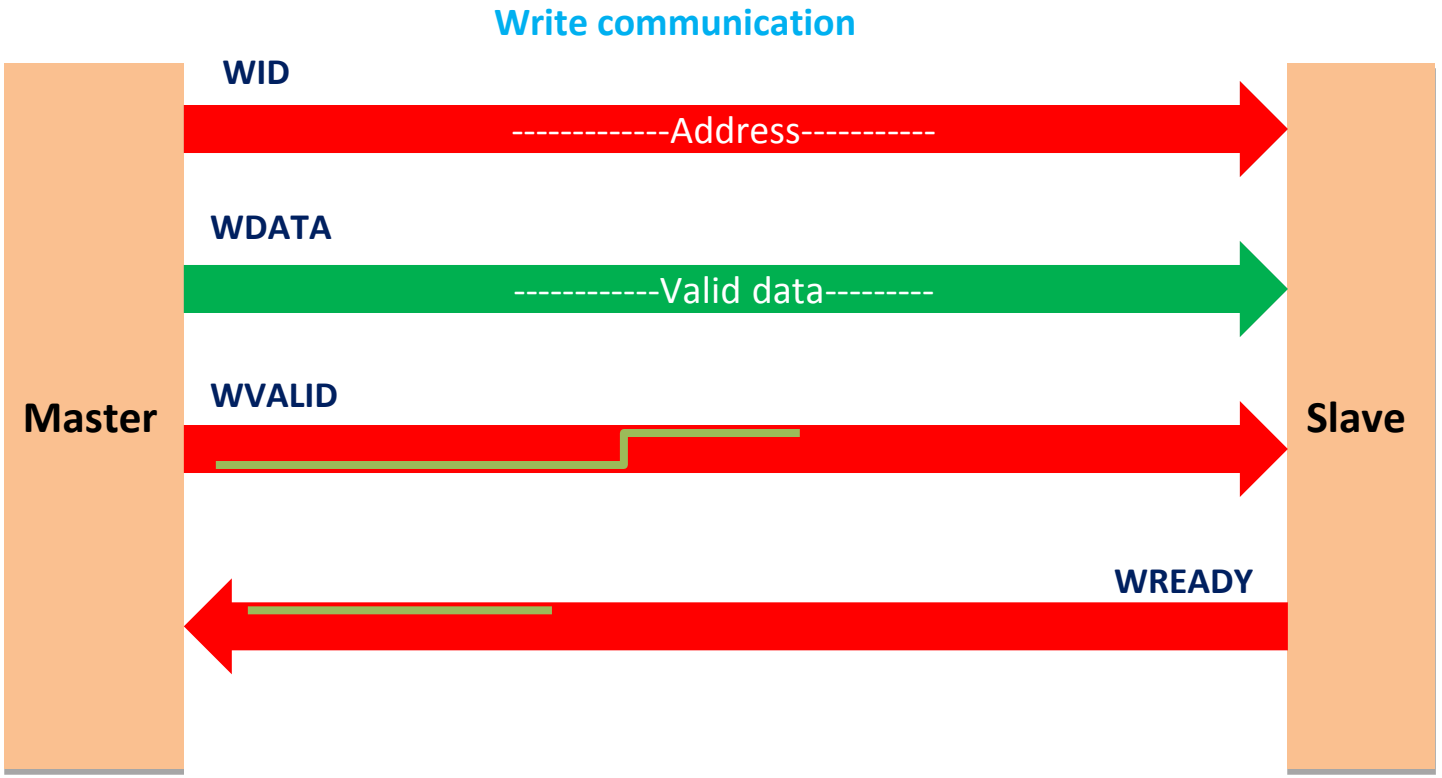
Wvalid : Request signal
wready : Acknowledgement signal

Write/Read communication(2/2)



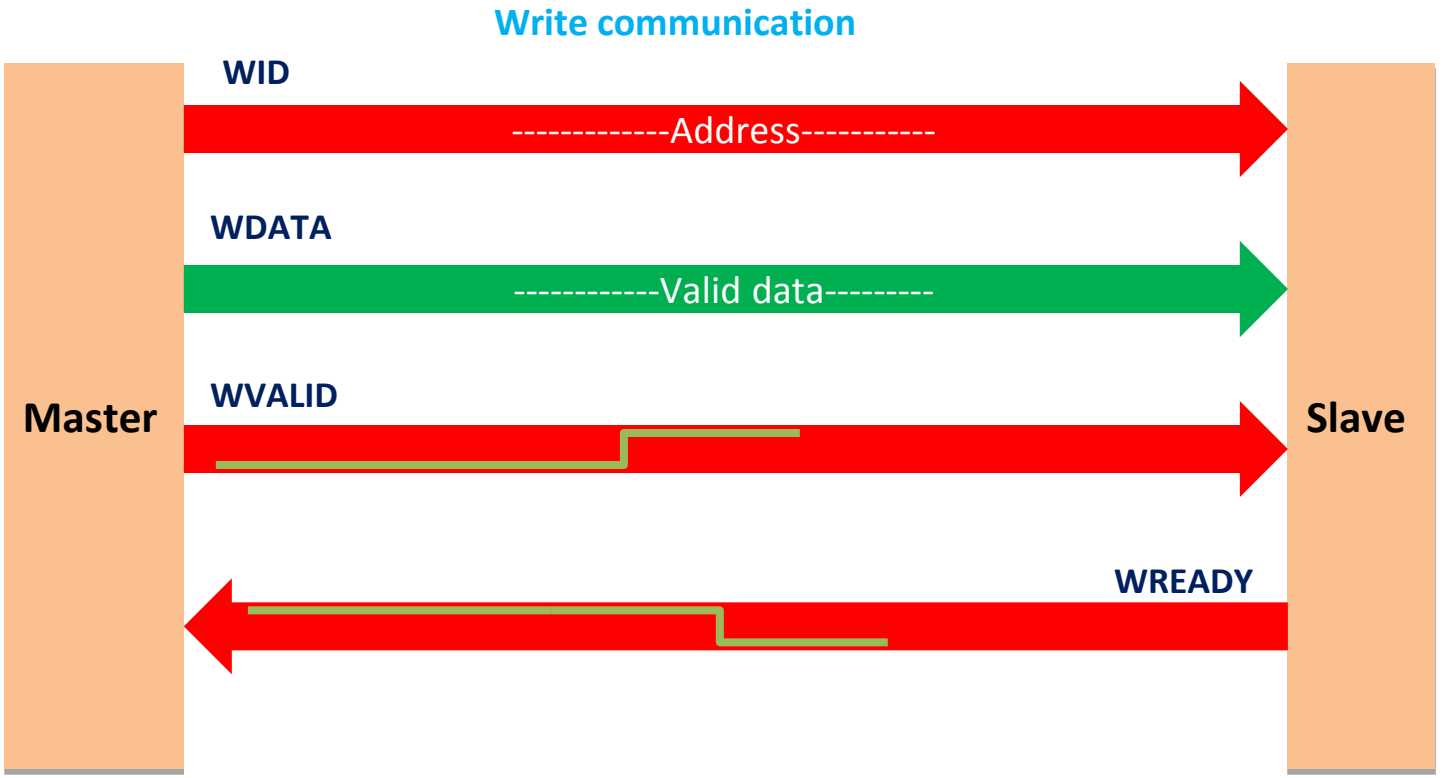
Wvalid : Request signal
wready : Acknowledgement signal

Write/Read communication(2/2)



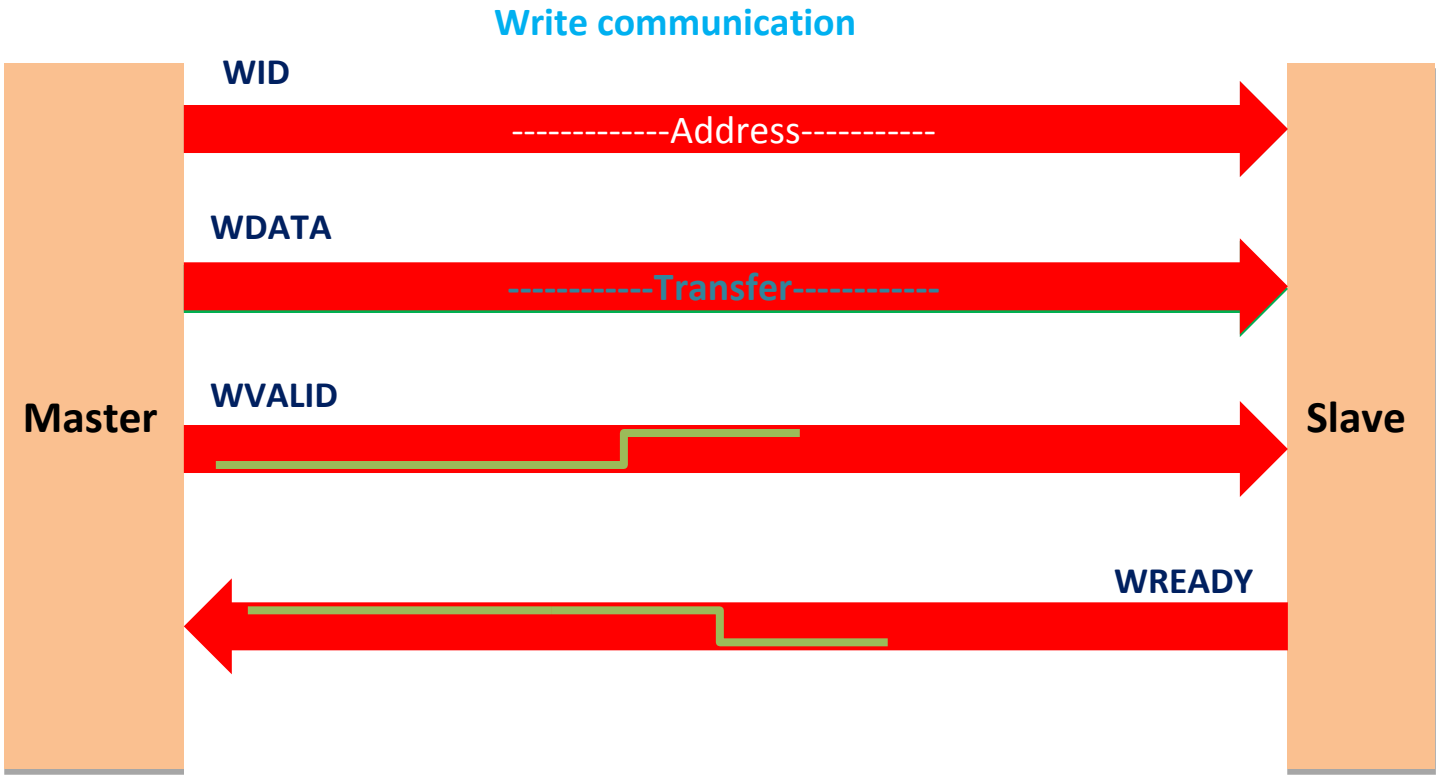
Wvalid : Request signal
wready : Acknowledgement signal

Write/Read communication(2/2)



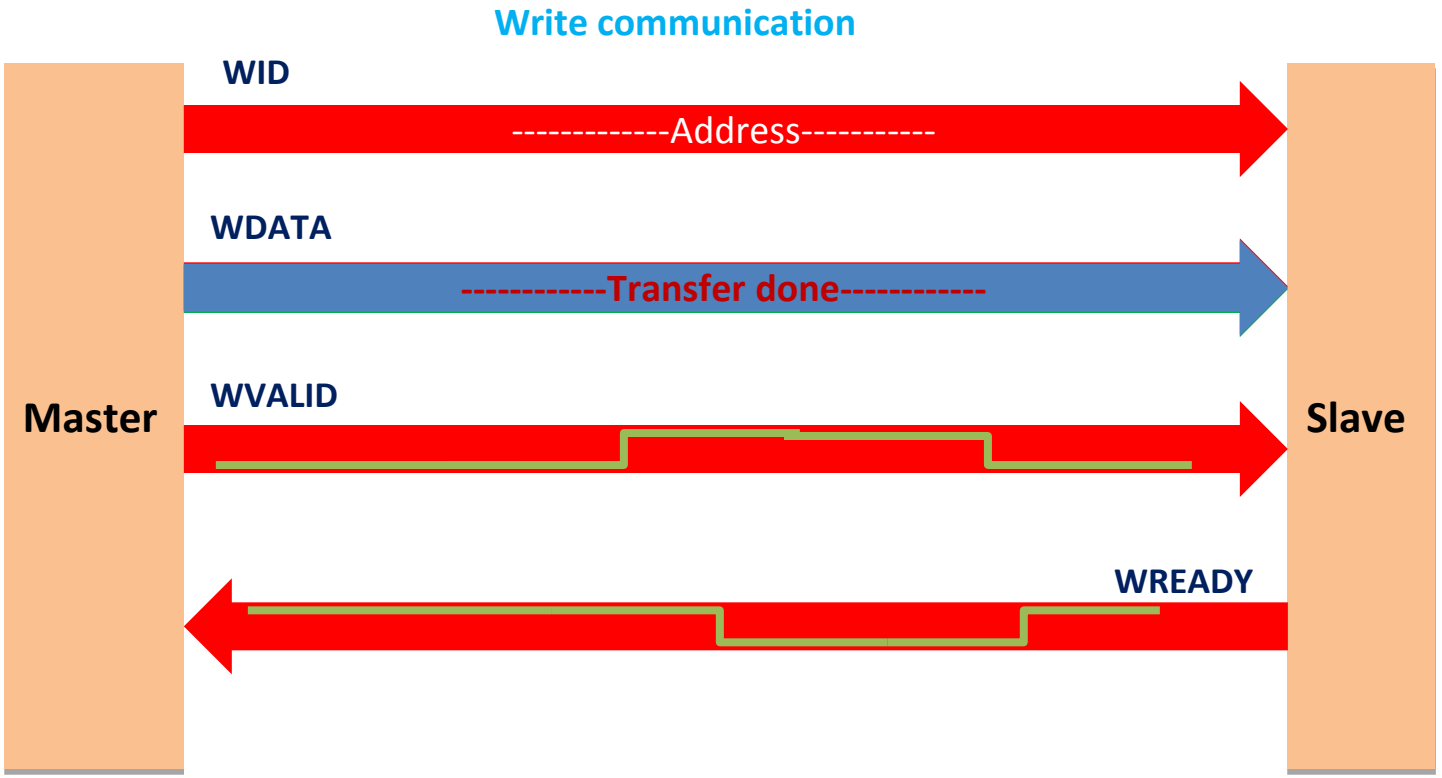
Wvalid : Request signal
wready : Acknowledgement signal

Write/Read communication(2/2)



Wvalid : Request signal
wready : Acknowledgement signal

Write/Read communication(2/2)

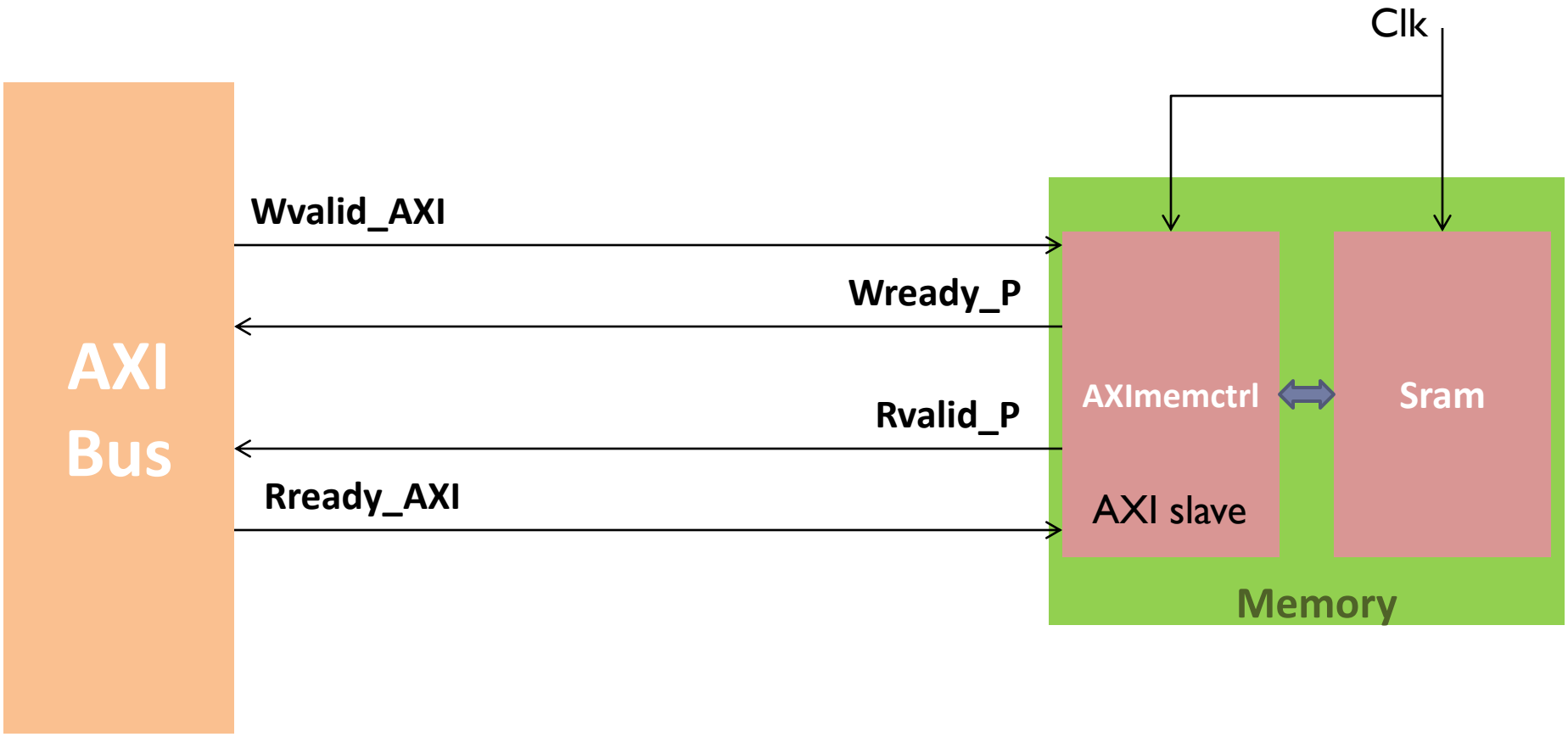


Wvalid : Request signal
wready : Acknowledgement signal

Outline

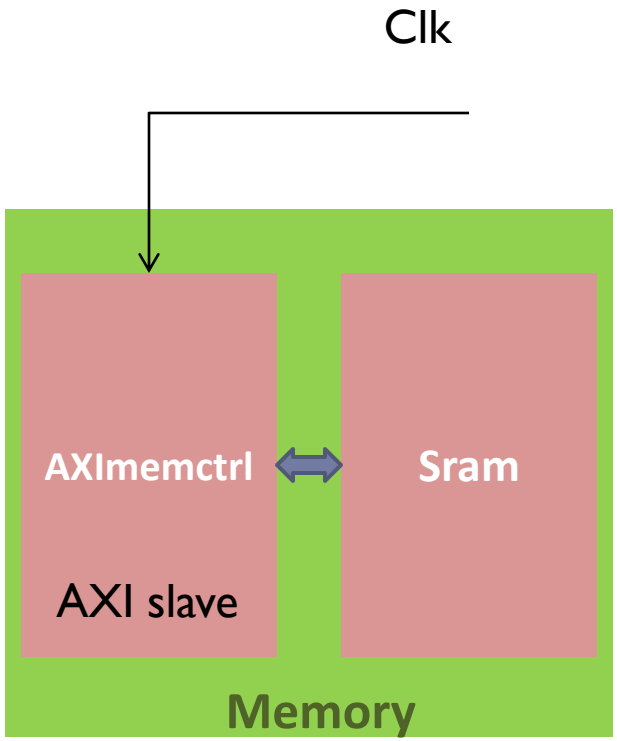
- **Part 1:**
  - Synchronous and asynchronous systems
  - Control structure
  - AXI bus
  - **Insertion methodology**
- Part 2:
  - Automatic insertion for low power structure
  - Results
- Conclusions and perspectives

Insertion methodology (1/2)

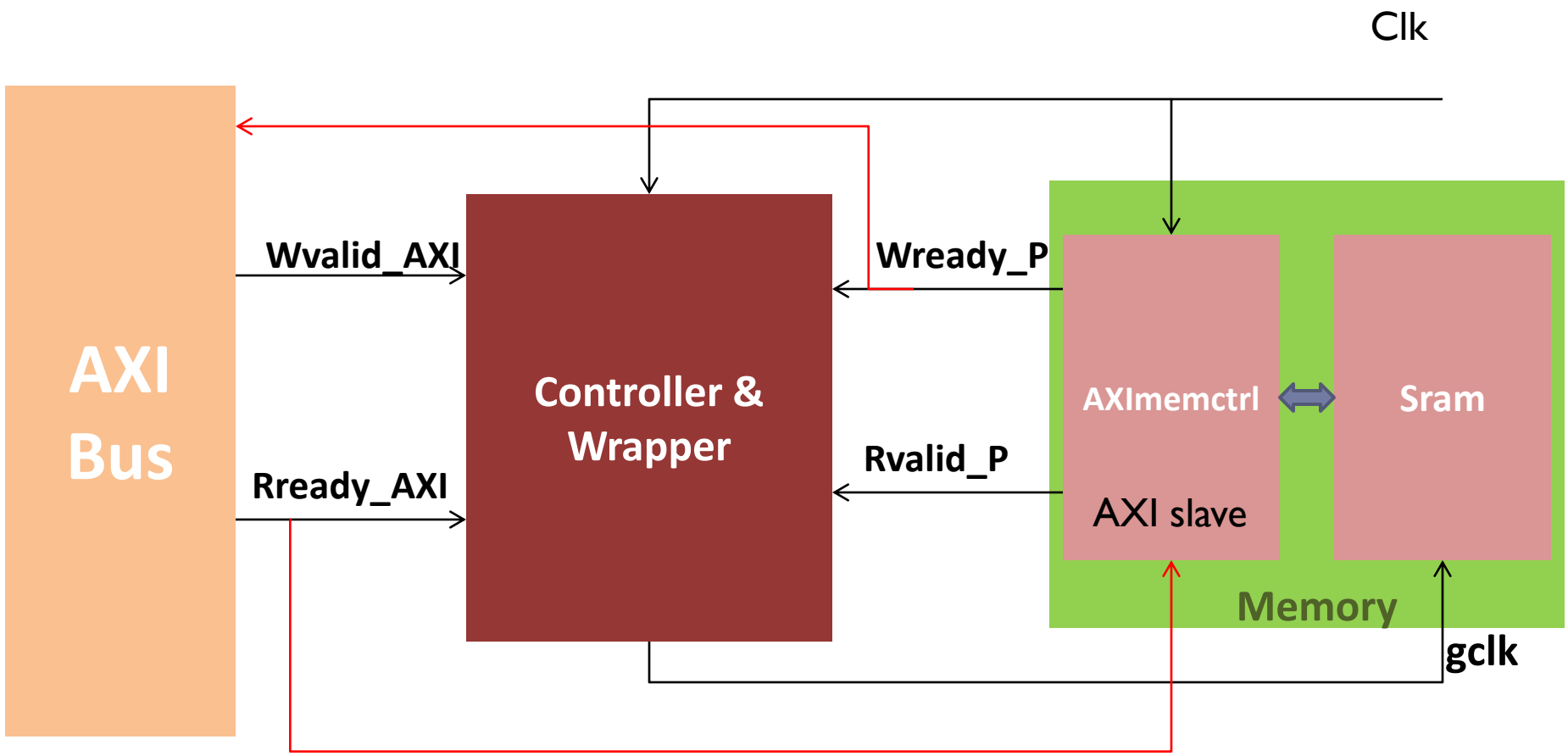


Insertion methodology (1/2)

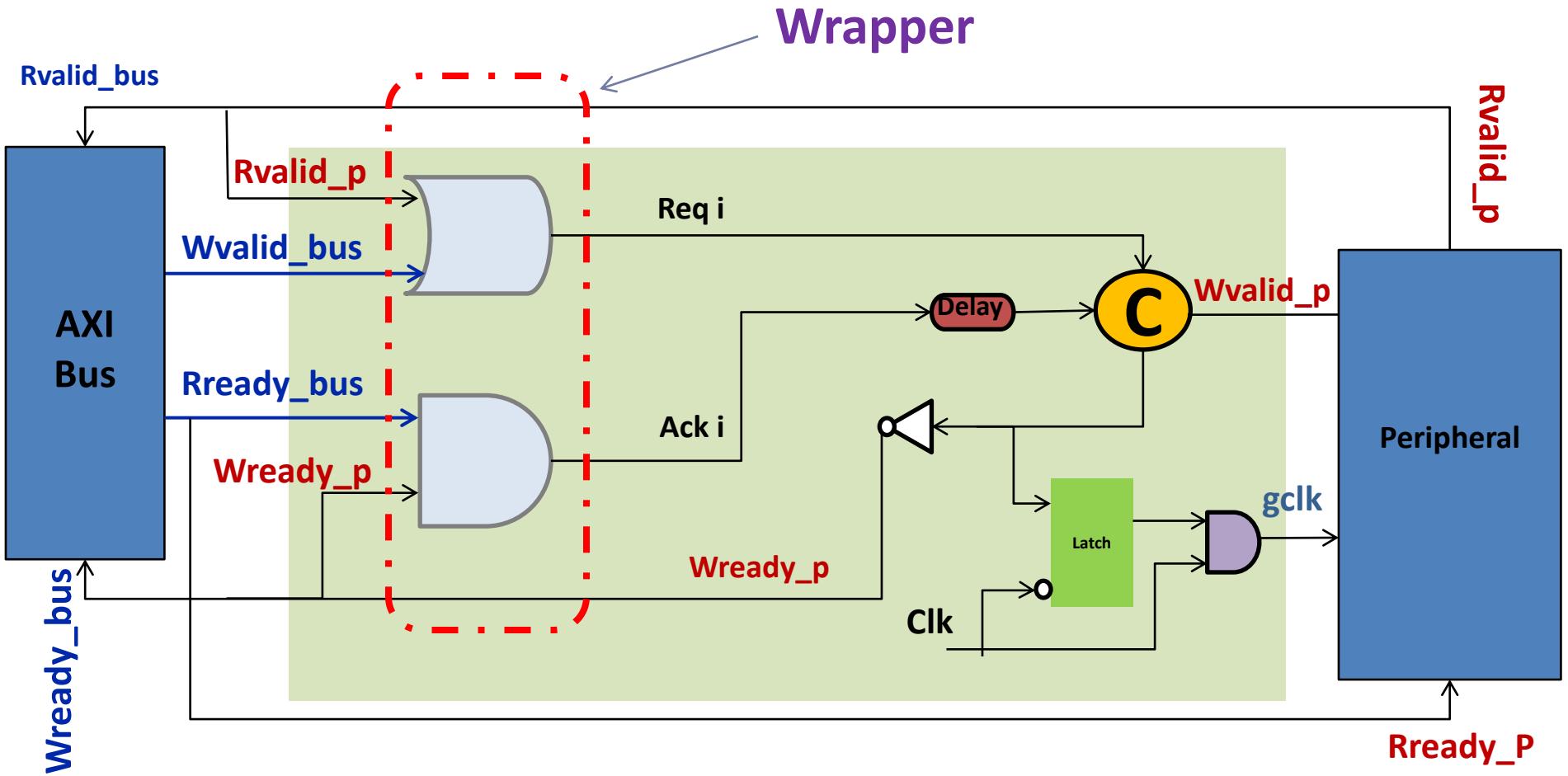
AXI Bus



Insertion methodology (1/2)



Insertion methodology (2/2)



Identification of signals that adapt with my network

Outline

- Part 1:
  - Synchronous and asynchronous systems
  - Control structure
  - AXI bus
  - Insertion methodology
- **Part 2:**
  - **Automatic insertion for low power structure**
  - **Results**
- Conclusions and perspectives

Automatic insertion for low power structure(1/3)

An innovative EDA tool used for the implementation flow

STAR – RTL DESIG BUILDER
(DEFACTO Technologies)

- Tcl based APIs
- Design Editing
- Design Exploration
- RTL Generation

STAR tools : created by DEFACTO (Moirans, France)

Automatic insertion for low power structure(2/3)

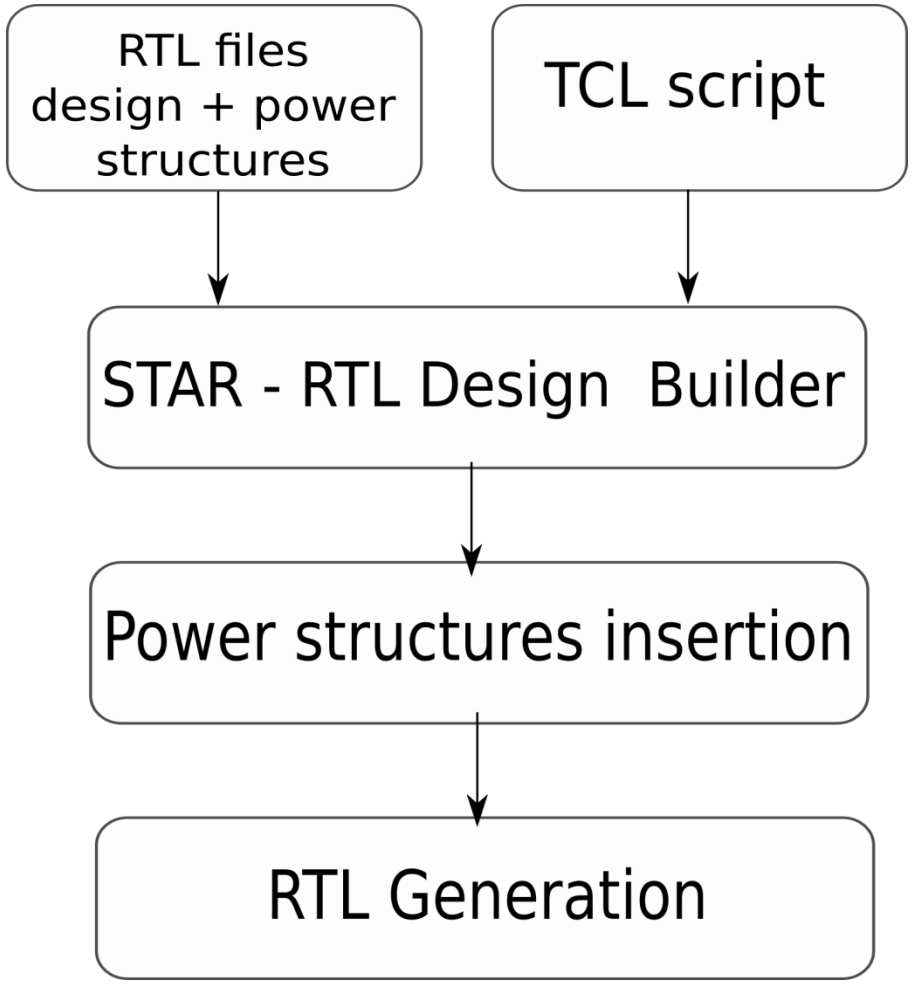
The automated implementation procedure

- Inputs :
  - Name of AXI slave (AXIMemCtrl)
  - Control signals (WVALID, RVALID, RREADY, WREADY)
  - Control structures (generic)
  - wrappers (specific to busses and NoCs)

Automatic insertion for low power structure(2/3)

The automated implementation procedure

- Algorithm:
  - Traverse hierarchy & detect memories
  - Insert power structures at the same level of memories
  - Add needed connectivity
  - Generate modified RTL (codes contain low power structures)



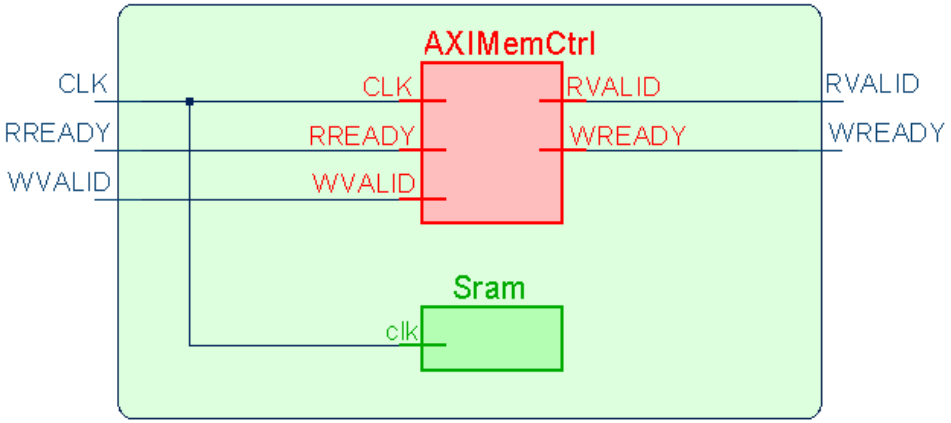
Automatic insertion for low power structure(3/3)

The automated implementation procedure

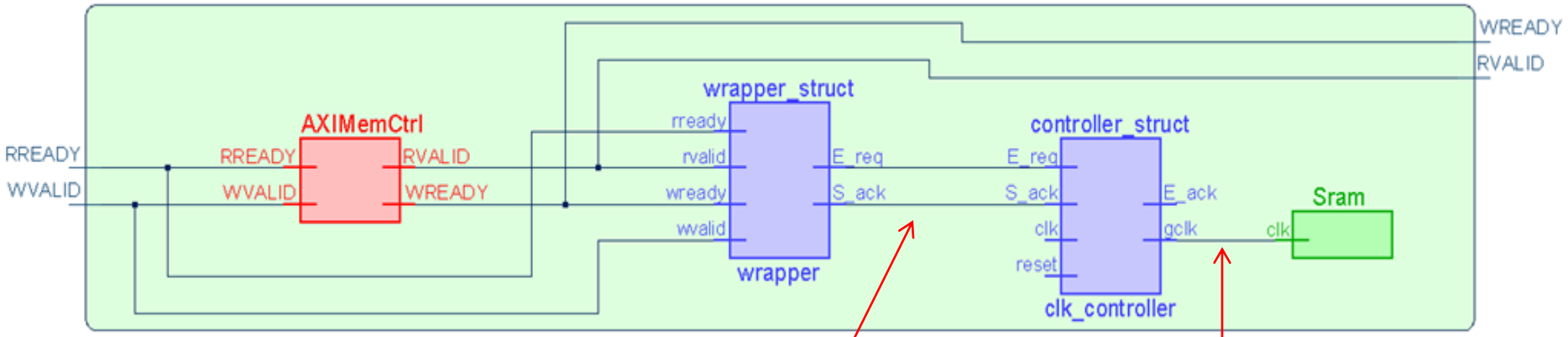
- Tcl script is developed (load design, load low power structures …)
- Use of Tcl APIs of STAR Tools to :
  - Insert controller
  - Insert wrappers based on control signals
  - Remove unneeded connections
  - Insert connections between the AXI bus slave, wrapper, controller and memory (verification with graphic interface of STAR and RTL code generation)

Results

Before insertion



After insertion



Generation of Req and Ack

Outline

- Part 1:
  - Synchronous and asynchronous systems
  - Control structure
  - AXI bus
  - Insertion methodology
- Part 2:
  - Automatic insertion for low power structure
  - Results
- **Conclusions and perspectives**

Conclusions and perspectives

- Design of low power structure for clock management
- Automatic insertion in systems using **AXI bus**
- Insertion based only on **control signals**
- Implementation methodology is validated and the generated **RTL** is as expected
- Further work will allow **to enhance** the methodology and the insertion algorithm
- Generalize the approach for different types of buses and **not only** for AXI bus (next step)

Thank you !!

Questions??