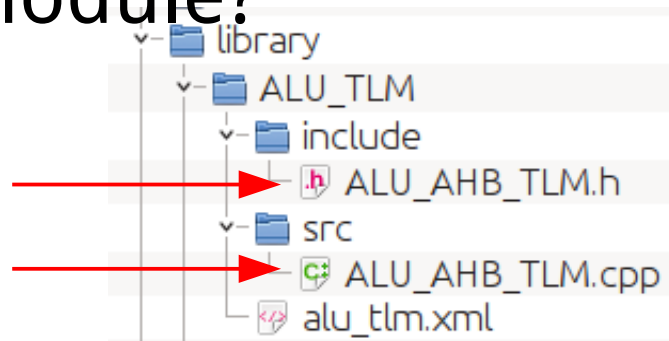


# **Write a AHB Slave Module for PA Using SystemC**

d01943032 Yu Sheng Lin

# Preface

- You have learned how to add a SystemC module to PA few weeks ago
- But how to write a custom SystemC AHB slave module?



- If you are so ~~free~~ earnest that you have read the example thoroughly, then you can leave now XD

# Write a AHB Slave

```
#include "systemc.h"  
#include "AMBA/AMBA.h"
```

← New, but trivial

↖ This module has an AHB slave port,  
1-bit addressing space and 32-bit  
data width

```
SC_MODULE(MyFirstAMBASlave) {  
public:  
    AMBA::AHBLiteTarget_inouts slave_port<1, 32> p_AHB;  
    SC_HAS_PROCESS(MyFirstAMBASlave);  
    MyFirstAMBASlave(const sc_module_name name);  
};
```

↑  
MyFirstAMBASlave.h

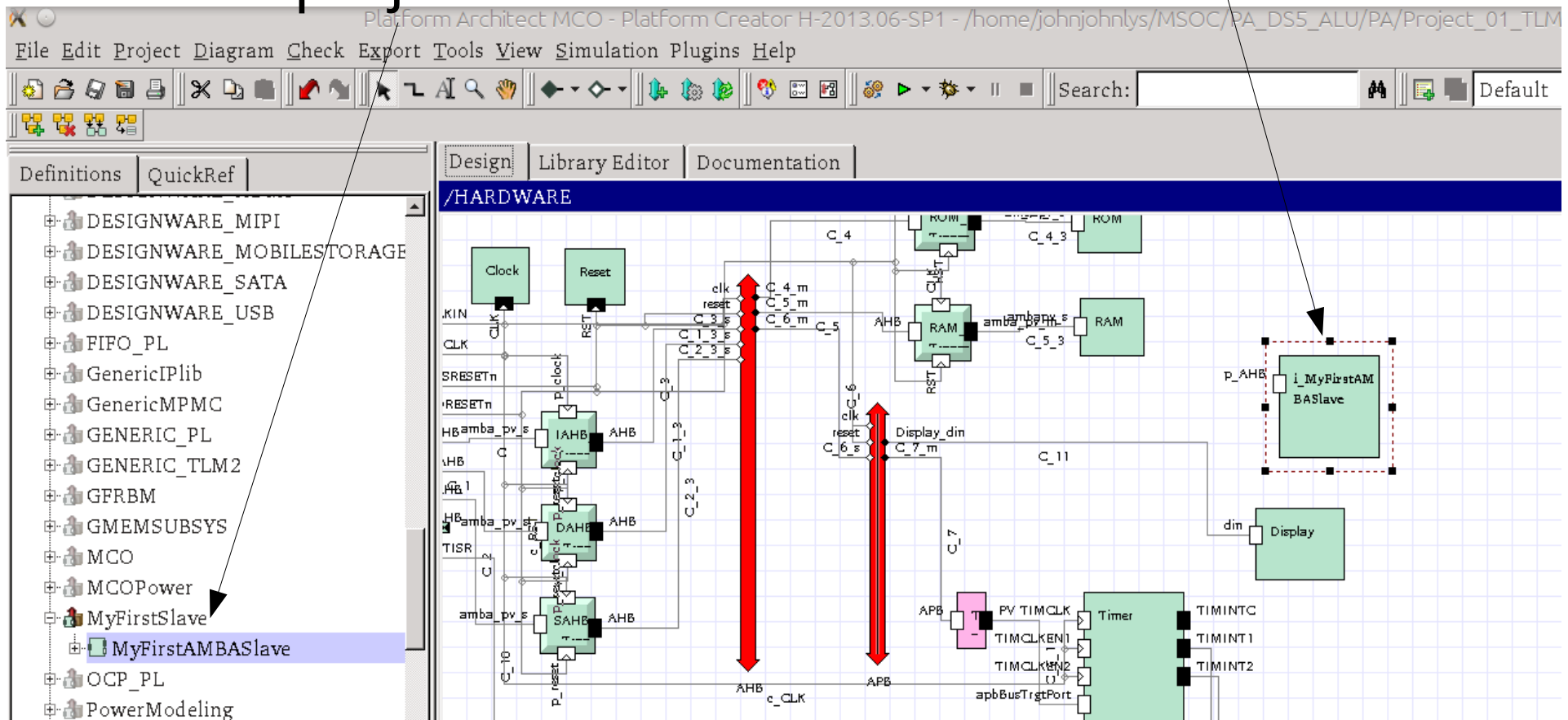
↗ MyFirstAMBASlave.cpp

```
#include "MyFirstAMBASlave.h"
```

```
MyFirstAMBASlave::MyFirstAMBASlave(  
    const sc_module_name name  
) : sc_module(name)  
{  
}
```

# The Result

- Repeating previous steps, you can drag the cell in to the diagram
  - The project name does not matter



# Today's Target

- A module that just invert the data written to it
  - Write 0x0f0f0f0f
  - Read 0xf0f0f0f0
- Quite easy, huh?

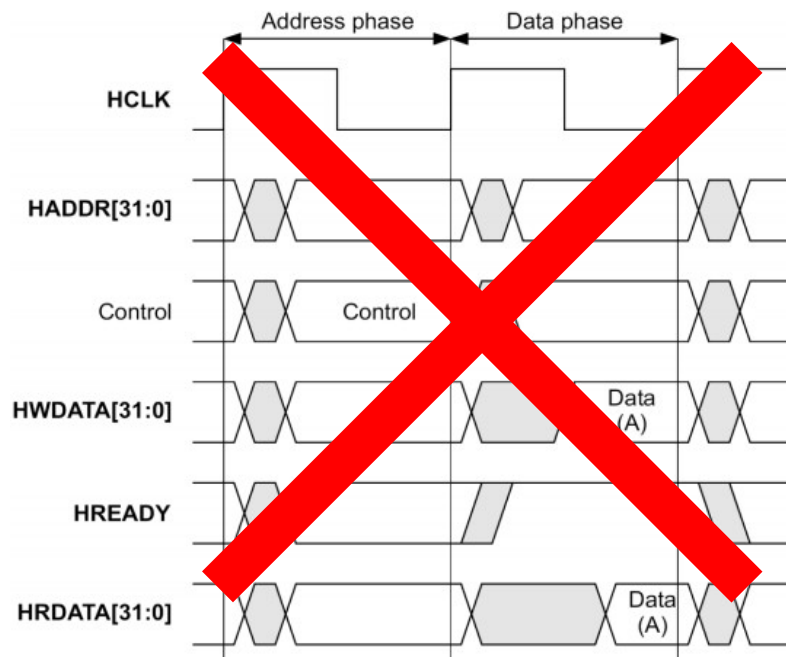
# Programming the Bus

- The AMBA model of PA doesn't require PCA knowledge of AHB
  - TLM, Event based design

```
AMBA::AHBLiteTarget_inouts slave_port<1, 32> p_AHB;
```

PCA protocol

Event based design  
(analog to javascript)



```
<button onclick="foo()">  
  Click me  
</button>
```

# The Most Important Events

```
SC_METHOD(receiveWriteData);  
sensitive << p_AHB.getReceiveWriteDataTrfEventFinder();  
dont_initialize();
```

```
SC_METHOD(sendReadData);  
sensitive << p_AHB.getSendReadDataTrfEventFinder();  
dont_initialize();
```

```
SC_METHOD(sendEoT);  
sensitive << p_AHB.getSendEotTrfEventFinder();  
dont_initialize();
```

→ Master want to write data

→ Master want to read data

EoT = End of transfer

TODO: Add these lines to constructor, define the functions in .h and implement it in .cpp

MY CODE DOESN'T WORK

I HAVE NO IDEA WHY

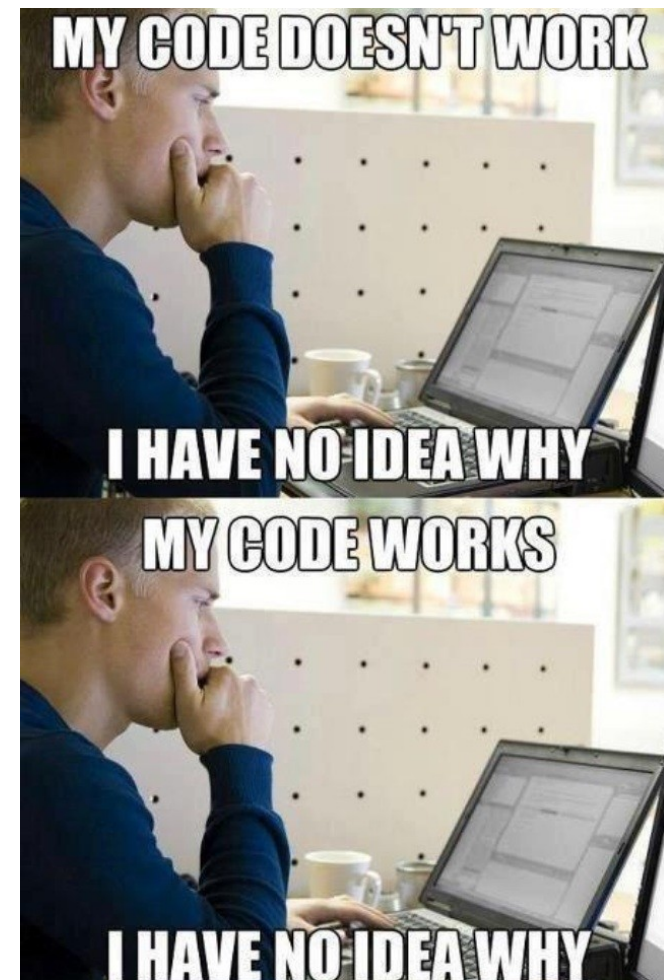
MY CODE WORKS

I HAVE NO IDEA WHY

# EoT Event

- Just write it and don't ask me why

```
void MyFirstAMBASlave::sendEoT()  
{  
    p_AHB.sendEotTrf();  
}
```





# Write Event

- Our Slave have 1-bit address

```
void MyFirstAMBASlave::receiveWriteData()
{
    p_AHB.getWriteDataTrf();
    unsigned addr = p_AHB.WriteDataTrf->getAddrTrf()->getAddress();
    unsigned data = p_AHB.WriteDataTrf->getWriteData();

    loaclBuffer[(addr>>2)&1] = ~data;
}
```

← getAddress returns the raw address from CPU  
and it is aligned to double word, so we must  
mask and shift it

TODO: Add these lines to .cpp and declare  
variables if necessary

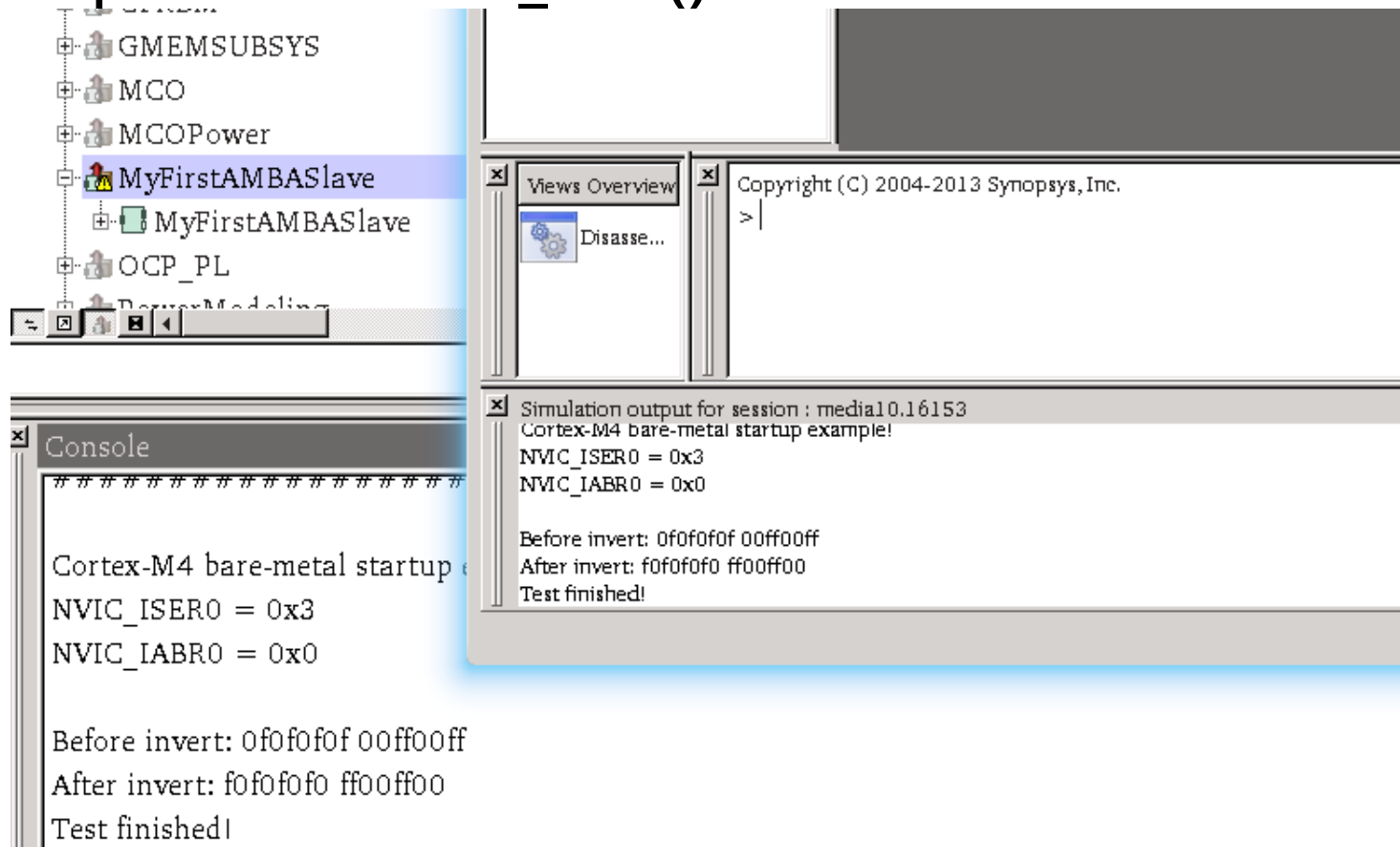
# Read Event

```
void MyFirstAMBASlave::sendReadData()
{
    p_AHB.getReadDataTrf();
    unsigned addr = p_AHB.ReadDataTrf->getAddrTrf()->getAddress();
    unsigned data;
    // TODO: what should the data be?
    p_AHB.ReadDataTrf->setReadData(data);
    p_AHB.sendReadDataTrf();
}
```

TODO: Add these lines to .cpp and assign appropriate value to data

# The C++ Code

- Please look into alu\_test.c and defines.h
  - Assign an appropriate address for MyFirstSlave
  - Explain the ALU\_test() function to the TA



# Caveats and Hints

- Importing SystemC
  - Project → Reload SystemC Modules
  - Right click on the module and click reload
- VPA SOP
  - Disconnect (in VPA) → Stop (in PA) → Connect (in VPA) → Load Image (in VPA)
  - These steps ensure load the newest image
- axf file
  - DS-5-Workspace/startup\_Cortex-M4/startup\_Cortex-M4.axf
  - Do not use the wrong files!