

Day 5

Hardware and Software Integration

MIAT-C3X FPGA Development Board

浯陽科技有限公司

姚敦凱

965402013@cc.ncu.edu.tw



CSIE MIAT Lab.
National Central University

Outline

- Part I
 - High Efficiency Communication
- Part II
 - Software and Hardware Integration

Part I

HIGH EFFICIENCY COMMUNICATION

Design Consideration

MIAT-STM32F103 communicate with MIAT-C3X

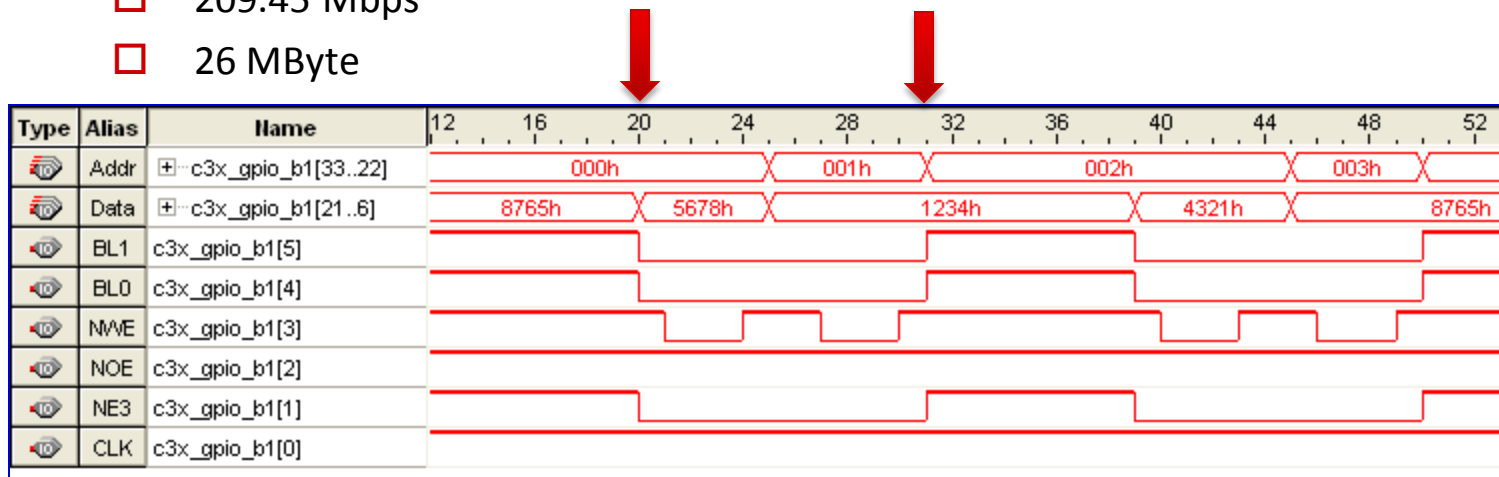
Protocols

- ☐ UART
- ☐ I²C
- ☐ SPI
- ☐ SDIO
- ☐ GPIO

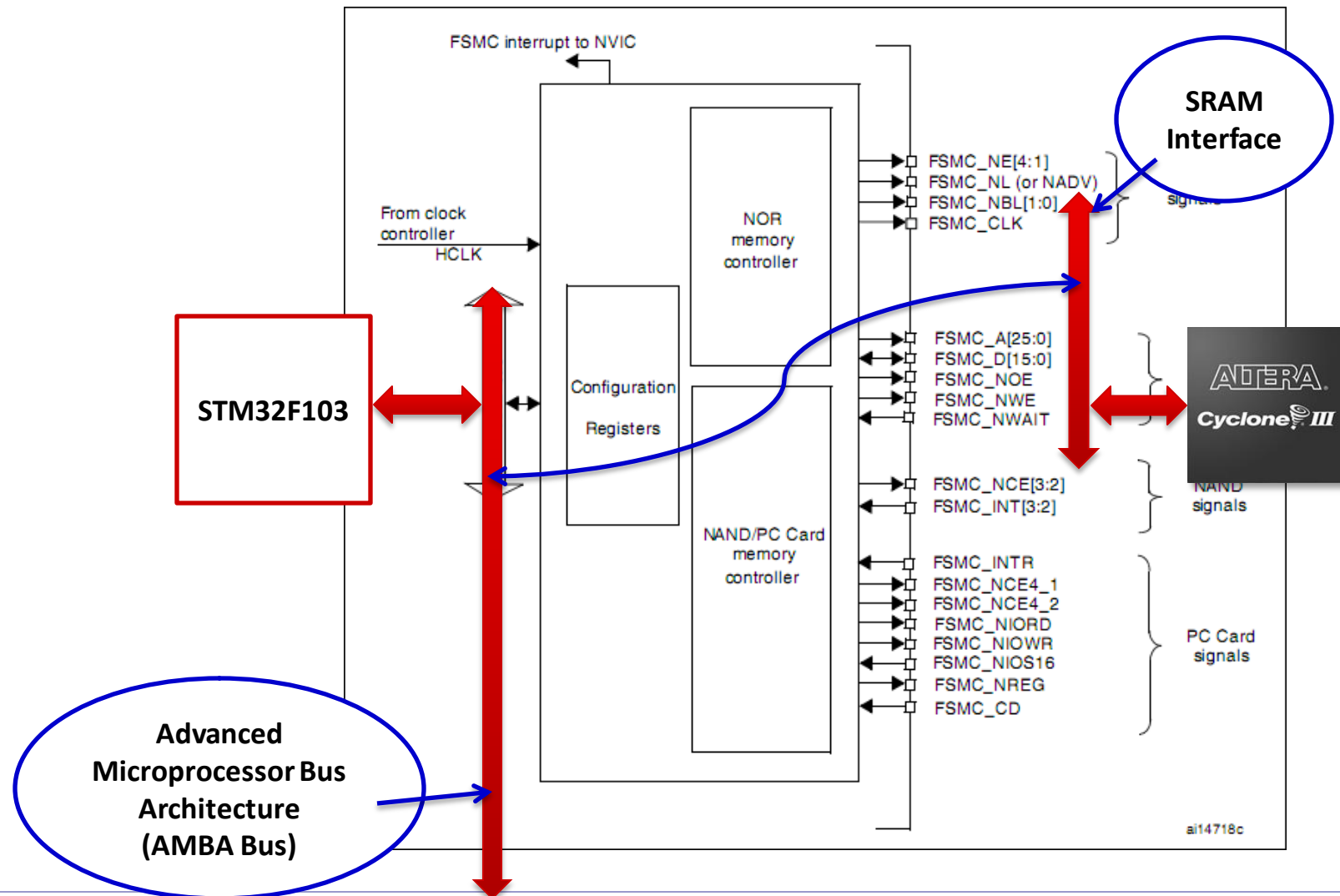


FSMC Interface

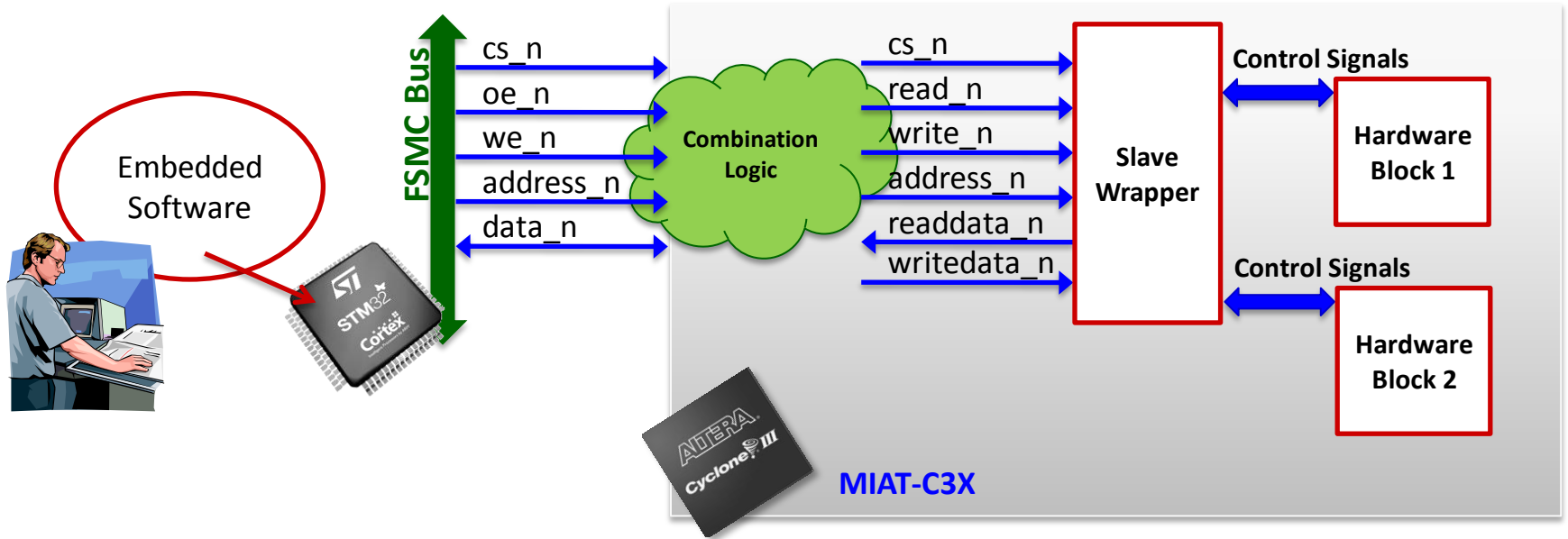
- ☐ (72Mhz*32-bit) / 11-Cycle
- ☐ 209.45 Mbps
- ☐ 26 MByte



FSMC Interface



System Diagram



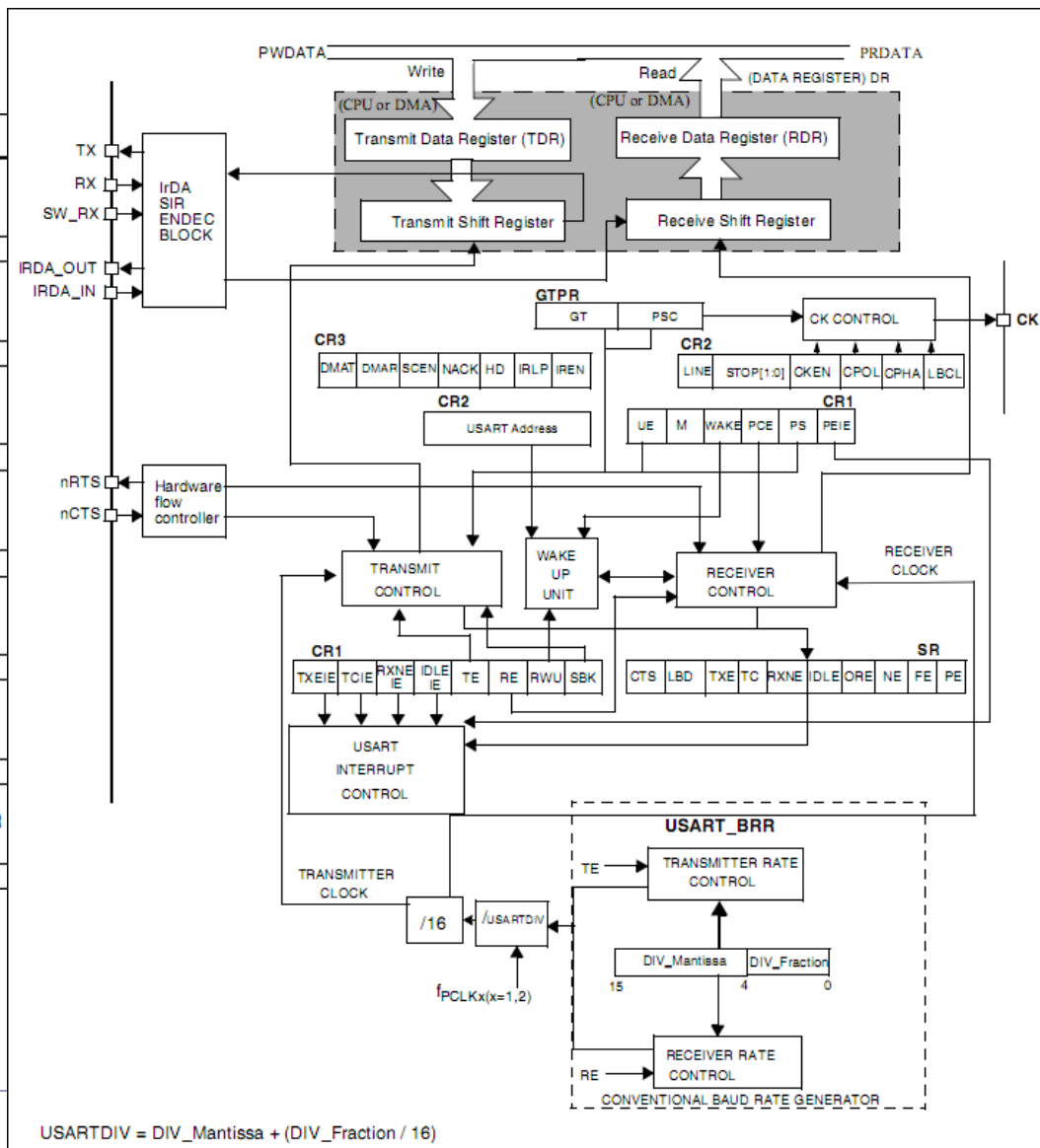
Memory Mapping

Offset	Register
0x00	USART_SR Reset value
0x04	USART_DR Reset value
0x08	USART_BRR Reset value
0x0C	USART_CR1 Reset value
0x10	USART_CR2 Reset value
0x14	USART_CR3 Reset value
0x18	USART_GTPR Reset value

Write register

Read register

Slave Wrapper



Lab 1. Register Access

- MIAT-C3X Slave Wrapper address base

- 0x68000000

Offset	Register Name
0x0	r_sw_reg_0
0x1	r_sw_reg_1
0x2	r_sw_reg_2
0x3	r_sw_reg_3

- Try to read and write them

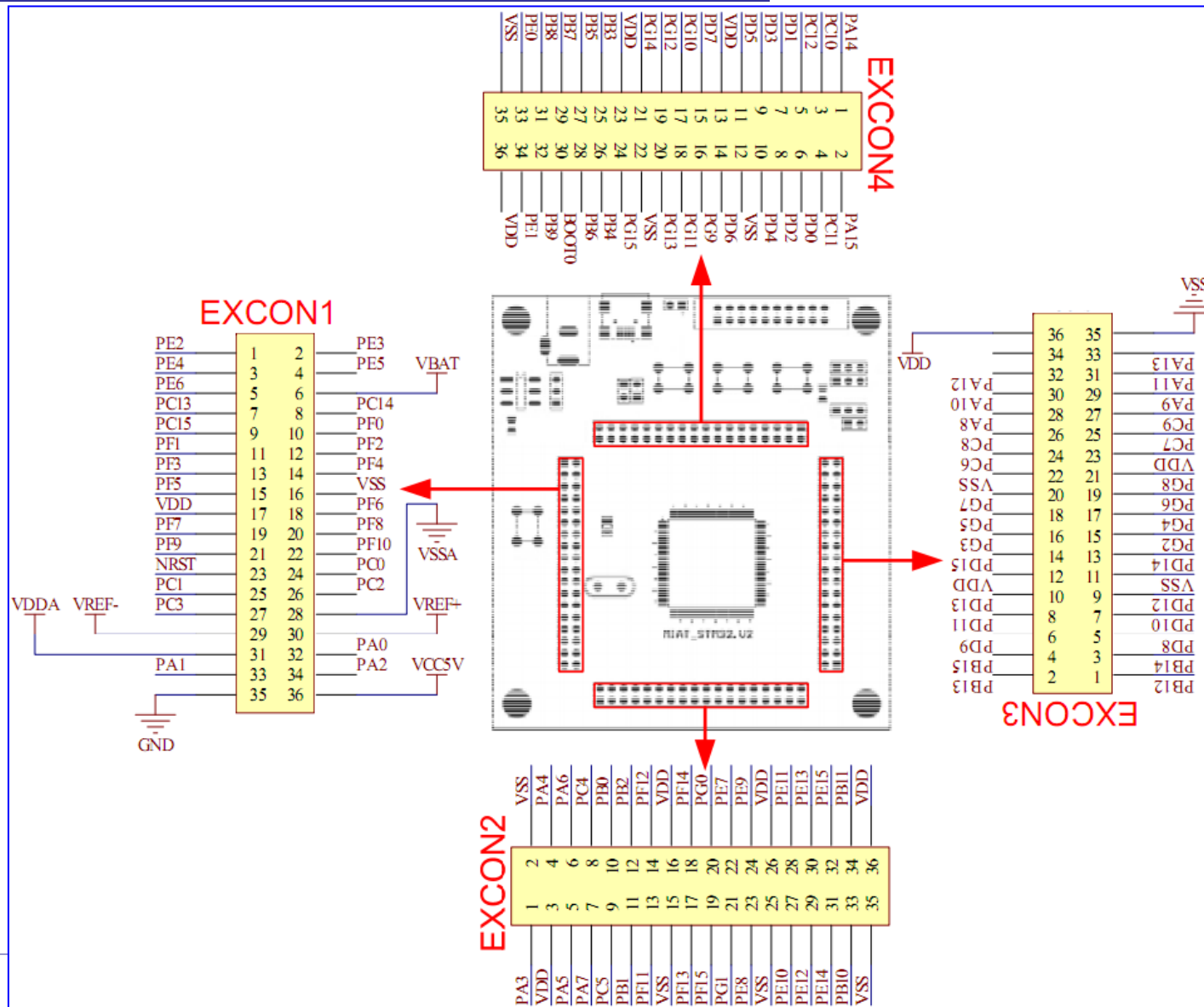
Lab 1. Connect 1 of 2

HDL Name	Pin Mapping	Description	MIAT STM32	MIAT IOB
c3x_clock50m	E16	miatc3x onboard		
c3x_dipsw0	E1	miatc3x onboard		
c3x_gpio_b1[0]			PA9	19. UART1_TX
c3x_gpio_b1[1]	D1	miatc3x-JP1 2.D1	PG10	
c3x_gpio_b1[2]	F2	miatc3x-JP1 3.F2	PD4	
c3x_gpio_b1[3]	F1	miatc3x-JP1 4.F1	PD5	
c3x_gpio_b1[4]	G2	miatc3x-JP1 5.G2	PE0	
c3x_gpio_b1[5]	G1	miatc3x-JP1 6.G1	PE1	
c3x_gpio_b1[6]	G5	miatc3x-JP1 7.G5	PD14	
c3x_gpio_b1[7]	B1	miatc3x-JP1 8.B1	PD15	
c3x_gpio_b1[8]	F3	miatc3x-JP1 9.F3	PD0	
c3x_gpio_b1[9]	J2	miatc3x-JP1 10.J2	PD1	
c3x_gpio_b1[10]	J1	miatc3x-JP1 13.J1	PE7	
c3x_gpio_b1[11]	K2	miatc3x-JP1 14.K2	PE8	
c3x_gpio_b1[12]	K1	miatc3x-JP1 15.K1	PE9	
c3x_gpio_b1[13]	L2	miatc3x-JP1 16.L2	PE10	
c3x_gpio_b1[14]	L1	miatc3x-JP1 17.L1	PE11	
c3x_gpio_b1[15]	N2	miatc3x-JP1 18.N2	PE12	

Lab 1. Connect 2 of 2

c3x_gpio_b1[16]	N1	miatc3x-JP1 19.N1	PE13	
c3x_gpio_b1[17]	P2	miatc3x-JP1 20.P2	PE14	
c3x_gpio_b1[18]	P1	miatc3x-JP1 21.P1	PE15	
c3x_gpio_b1[19]	K5	miatc3x-JP1 22.K5	PD8	
c3x_gpio_b1[20]	L4	miatc3x-JP1 23.L4	PD9	
c3x_gpio_b1[21]	R1	miatc3x-JP1 24.R1	PD10	
c3x_gpio_b1[22]	L3	miatc3x-JP1 25.L3	PF0	
c3x_gpio_b1[23]	N9	miatc3x-JP1 26.N9	PF1	
c3x_gpio_b1[24]	R10	miatc3x-JP1 27.R10	PF2	
c3x_gpio_b1[25]	T10	miatc3x-JP1 28.T10	PF3	
c3x_gpio_b1[26]	R11	miatc3x-JP1 31.R11	PF4	
c3x_gpio_b1[27]	P9	miatc3x-JP1 32.P9	PF5	
c3x_gpio_b1[28]	M10	miatc3x-JP1 33.M10	PF12	
c3x_gpio_b1[29]	N11	miatc3x-JP1 34.N11	PF13	
c3x_gpio_b1[30]	P11	miatc3x-JP1 35.P11	PF14	
c3x_gpio_b1[31]	N12	miatc3x-JP1 36.N12	PF15	
c3x_gpio_b1[32]	F14	miatc3x-JP1 37.F14	PG0	
c3x_gpio_b1[33]	A6	miatc3x-JP1 38.A6	PG1	

Lab 1. MIAT-STM32 Connector

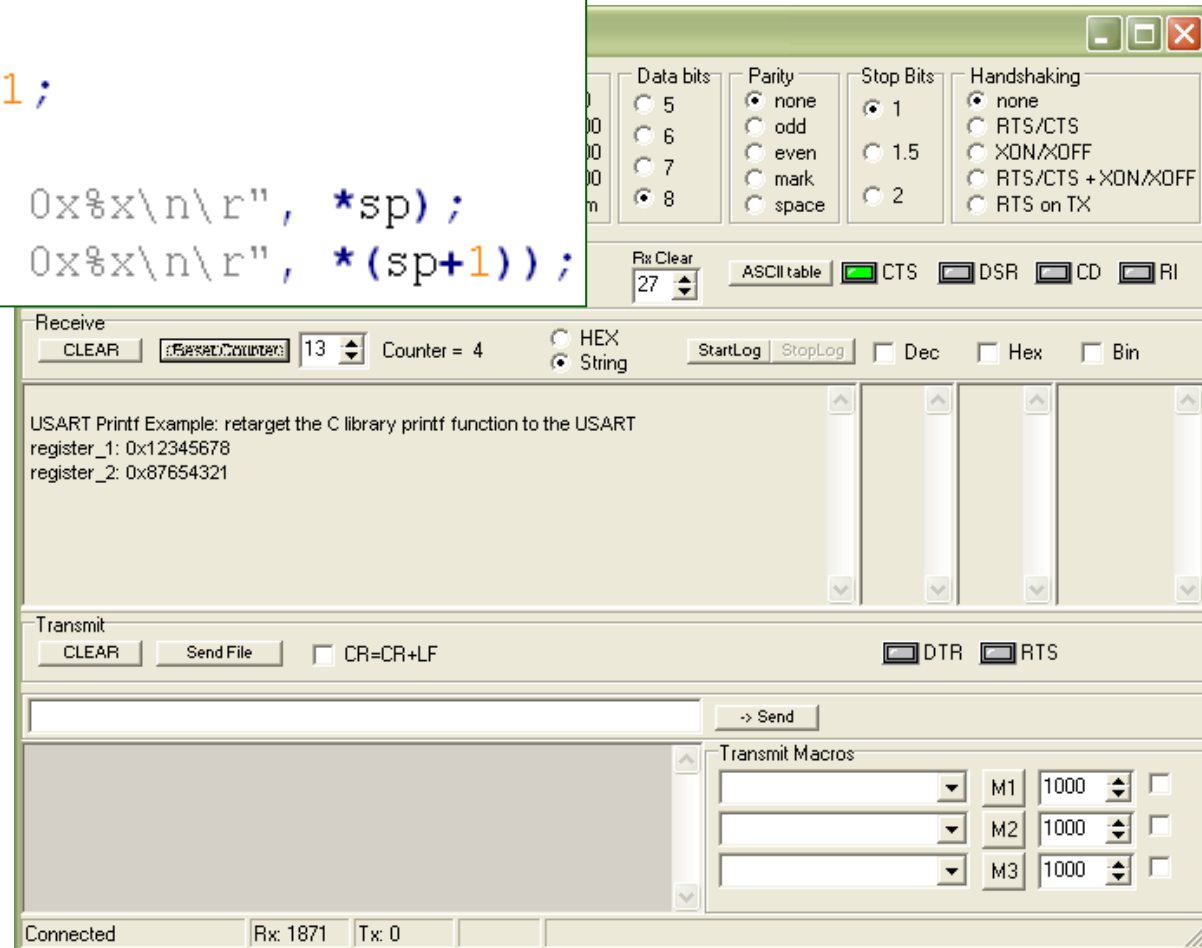


Lab 1. Software Flow

- Use pointer to operate

```
sp = (u32 *)0x68000000;
*sp = 0x12345678;
*(sp+1) = 0x87654321;

printf("register_1: 0x%x\n\r", *sp);
printf("register_2: 0x%x\n\r", *(sp+1));
```



The screenshot shows a serial terminal window with the following settings and output:

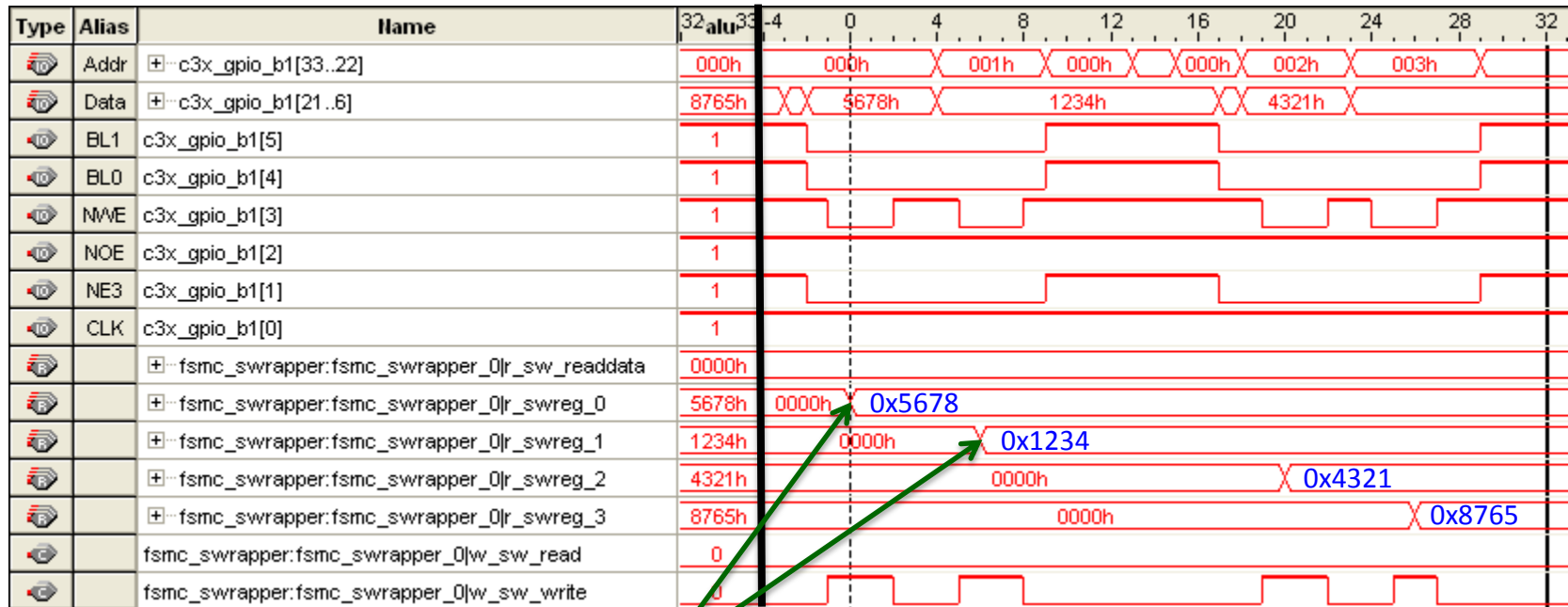
- Settings:**
 - Data bits: 8
 - Parity: none
 - Stop Bits: 1
 - Handshaking: none
 - Rx Clear: 27
 - ASCII table: ☒
 - CTS: ☒
 - DSR: ☐
 - CD: ☐
 - RI: ☐
- Receive Section:**
 - CLEAR button
 - Serial Port: 13
 - Counter = 4
 - HEX / String radio buttons (String is selected)
 - StartLog / StopLog buttons
 - Dec / Hex / Bin checkboxes (all unchecked)
- Transmit Section:**
 - CLEAR button
 - Send File button
 - CR=CR+LF checkbox (unchecked)
 - DTR / RTS checkboxes (both unchecked)
 - Send button
 - Transmit Macros section with three rows (M1, M2, M3) each set to 1000.
- Output Log:**

USART Printf Example: retarget the C library printf function to the USART
register_1: 0x12345678
register_2: 0x87654321
- Status Bar:**

Connected | Rx: 1871 | Tx: 0

Lab 1. Access Hardware

□ Hardware Waveform

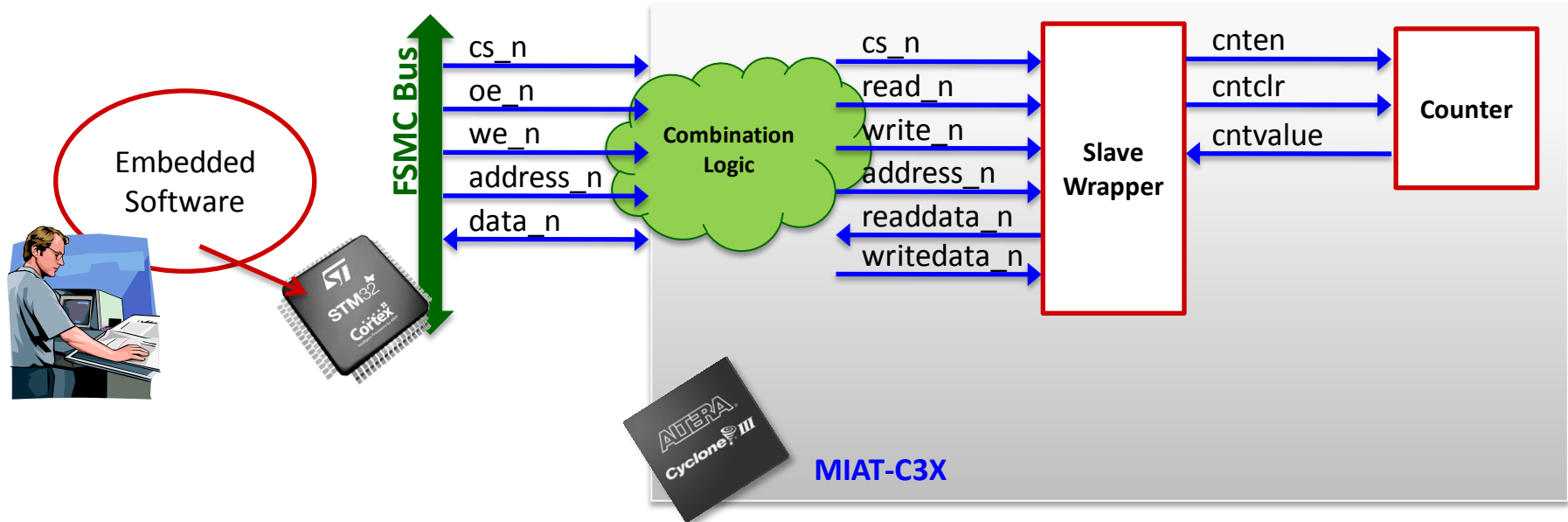


Carefully, ARM's operation is "little endian"

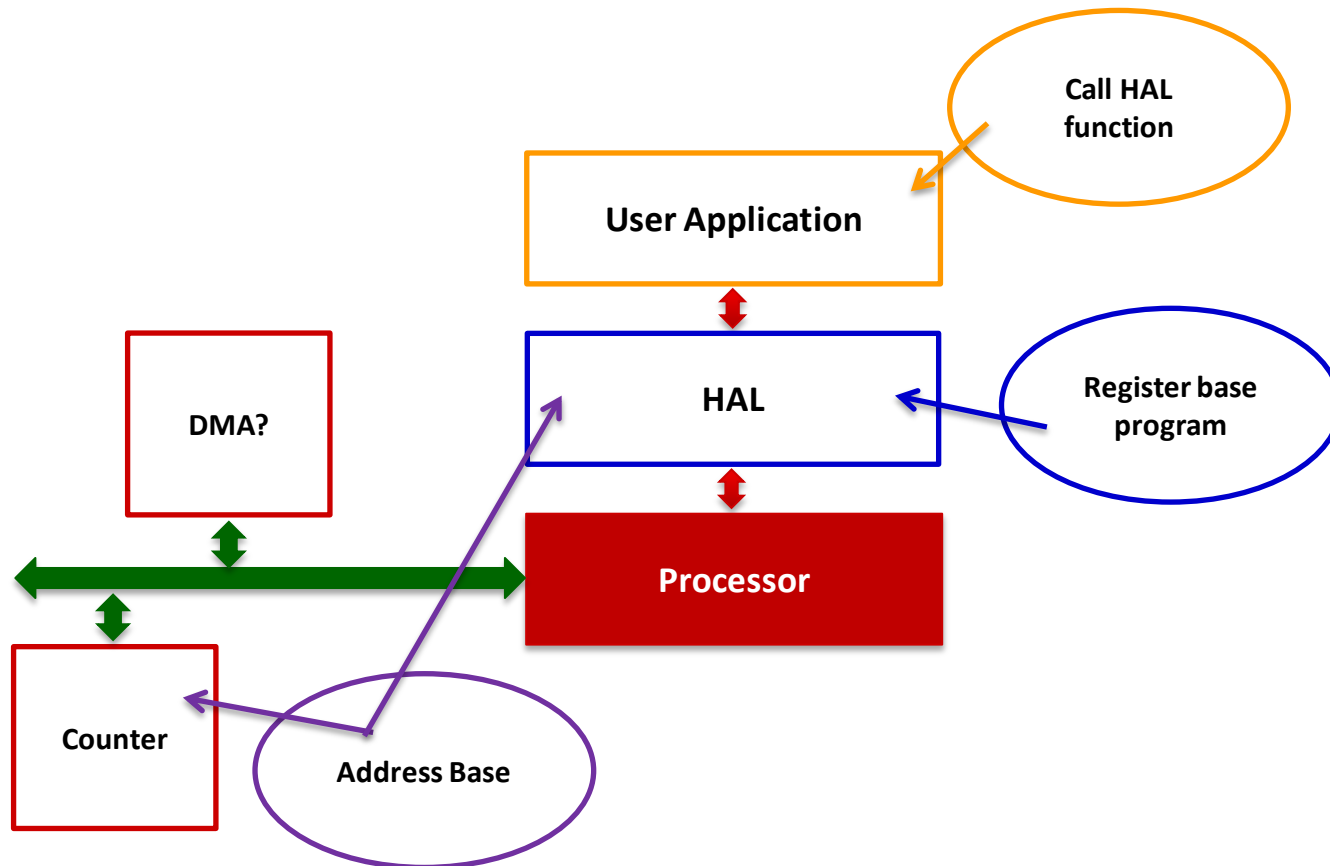
Part II

SOFTWARE & HARDWARE INTEGRATION

Lab 2. Counter



Lab 2. Software Architecture



Lab 2. Register Table

MIAT-C3X Counter Slave Wrapper address base

0x68000000

Offset	Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	CTRL	en															clr
	CTRH																
0x4	DRL	Low 16-bit Value															
	DRH	High 16-bit Value															

Lab 2. Software Flow

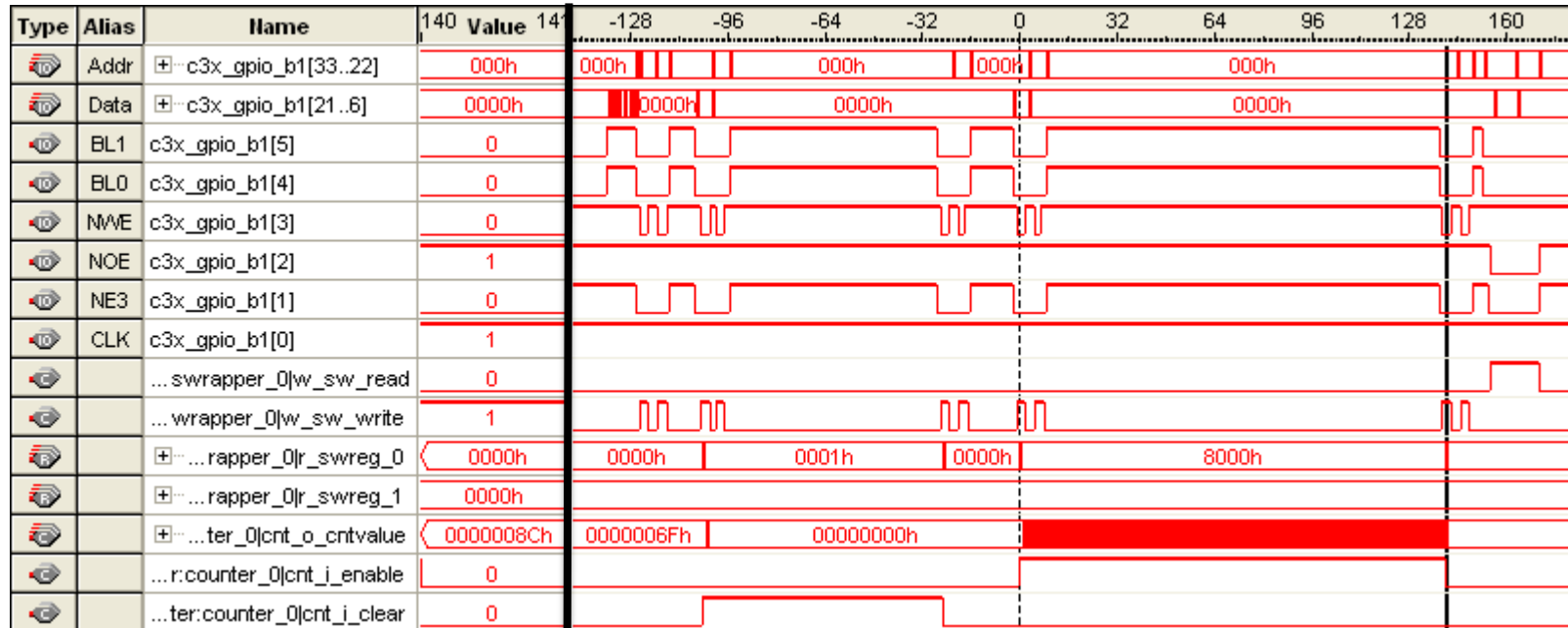
□ Call HAL Function

```
miatc3x_counter_disable();  
miatc3x_counter_clear();  
miatc3x_counter_enable();  
for(i=0;i<10;i++);  
miatc3x_counter_disable();  
  
printf("During Time: 0x%x", miatc3x_counter_get());
```

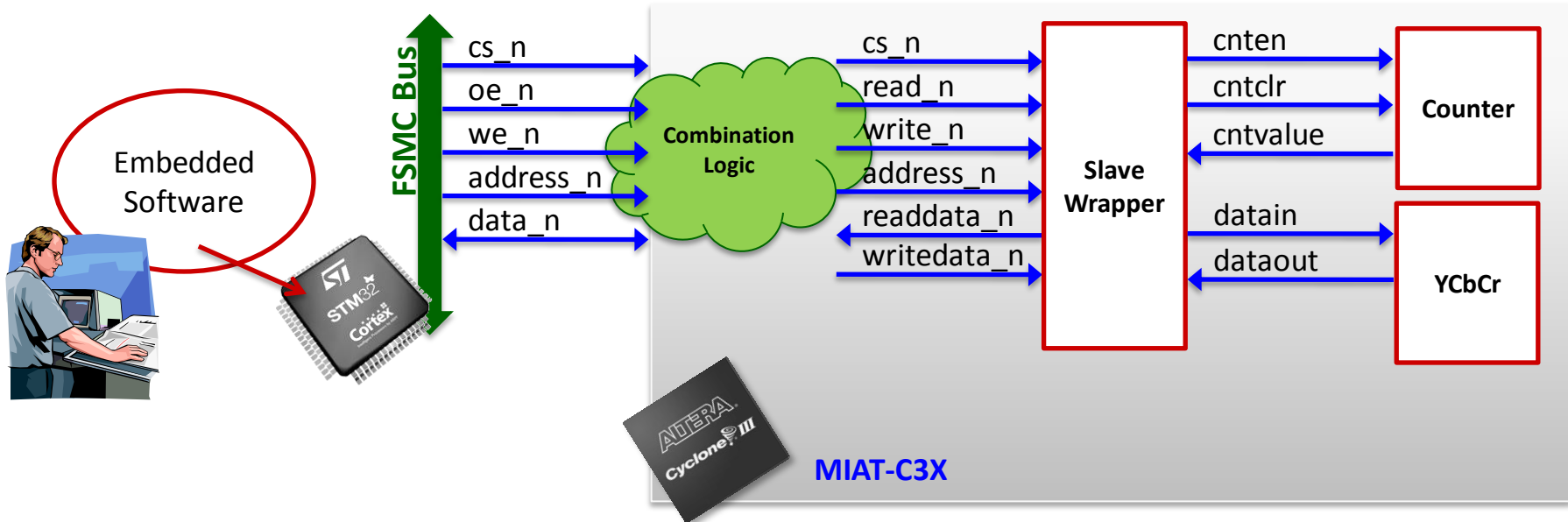
USART Printf Example: retarget the C library printf function to the USART
During Time: 0x8c

Lab 2. Access Hardware

□ Waveform



Lab 3. YCbCr Accelerator



Lab 3. Register Table

□ MIAT-C3X Counter Slave Wrapper address base
 ■ 0x68000008

Offset	Register	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	RGBL	Low 16-bit Value															
wo	RGBH	High 16-bit Value															
0x0	YUVL	Low 16-bit Value															
ro	YUVH	High 16-bit Value															

Lab 3. Compare

- ❑ ARM Cortex-M3
 - Optimized Fixed Point Compiler
- ❑ Hardware
 - Pipeline
 - Fifo buffer

USART Printf Example: retarget the C library printf function to the USART

Float point operation
> During Time: 2850 ns
Y: 104.008003
U: 71.389999
V: 193.156998

Fixed point operation
> During Time: 210 ns
Y: 105
U: 70
V: 193

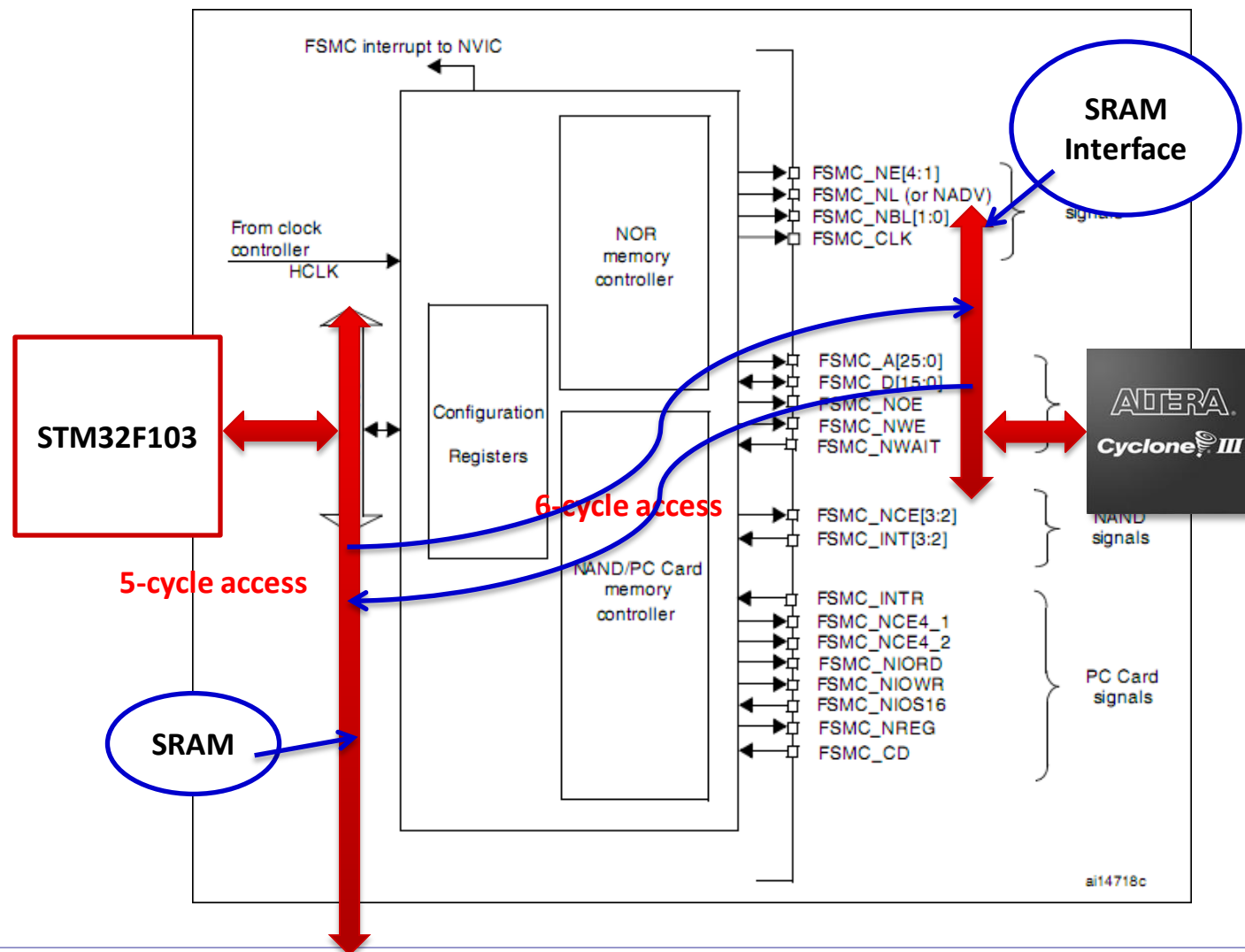
Fixed point operation
test: 6899393
> During Time: 390 ns
Y: 105
U: 70
V: 193

Floating Point Operation

Floating Point Operation

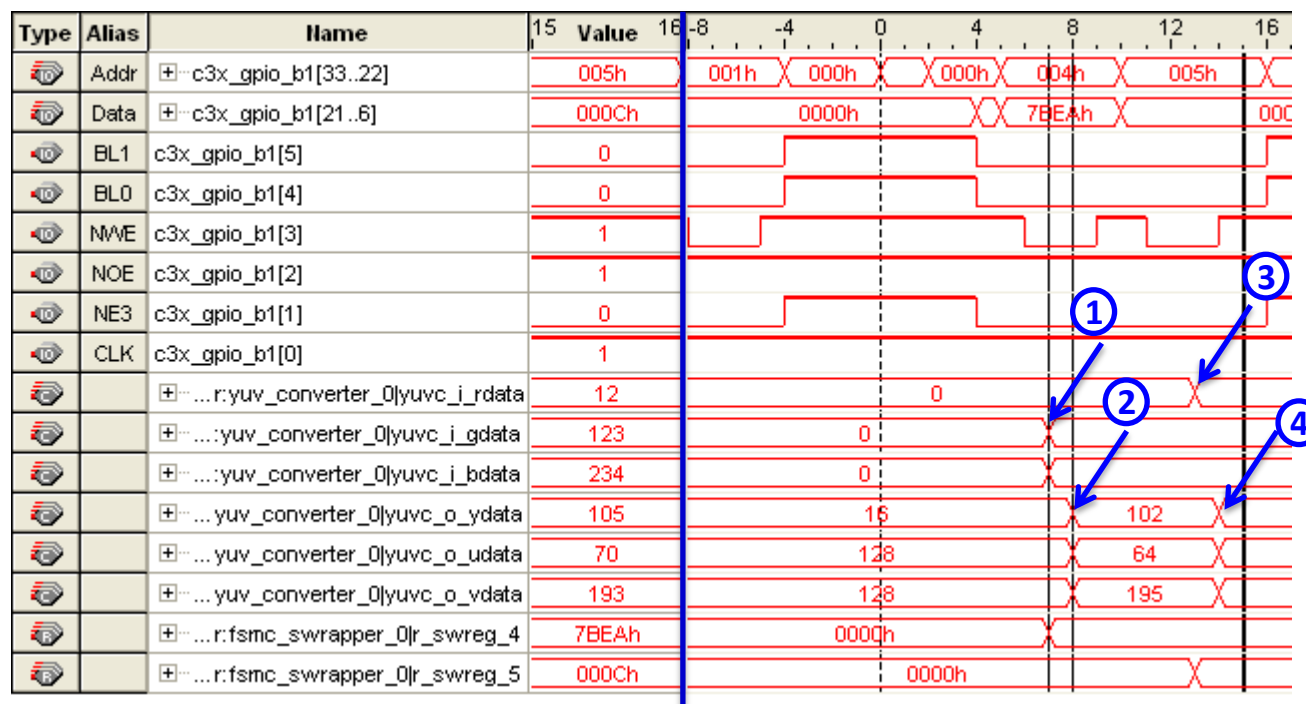
Floating Point Operation

Lab 3. Disadvantage

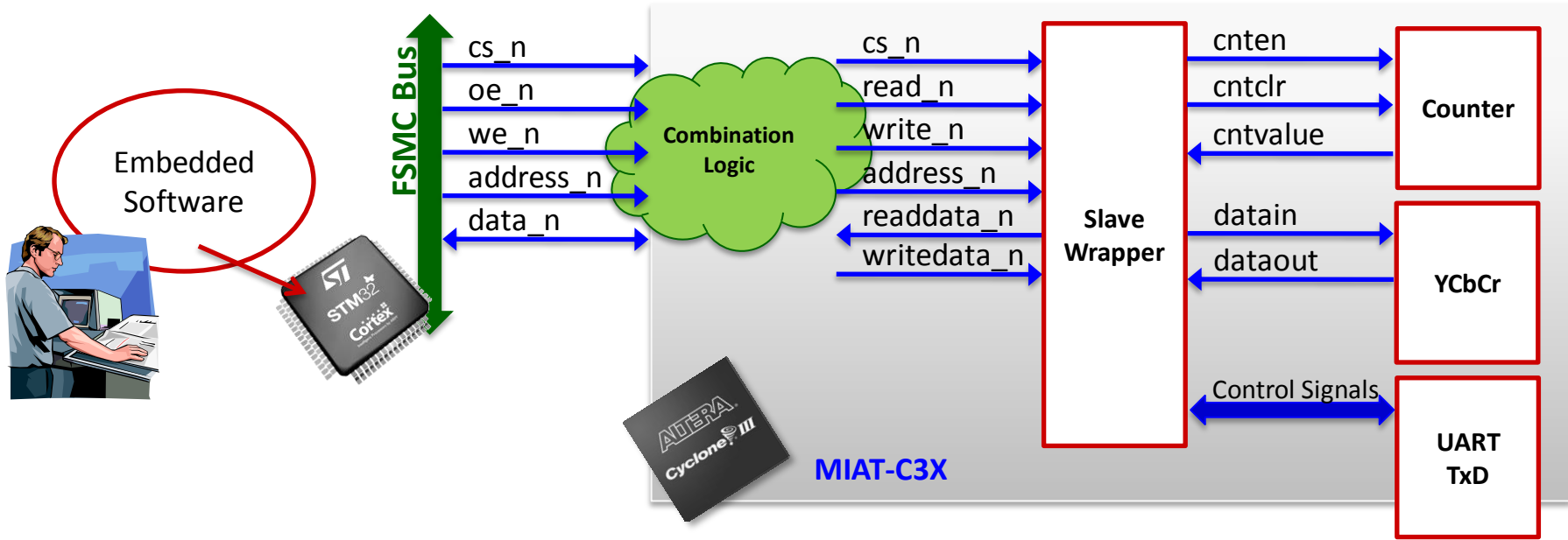


Lab 3. Hardware Operation

□ Waveform



Lab 4. UART Transmitter



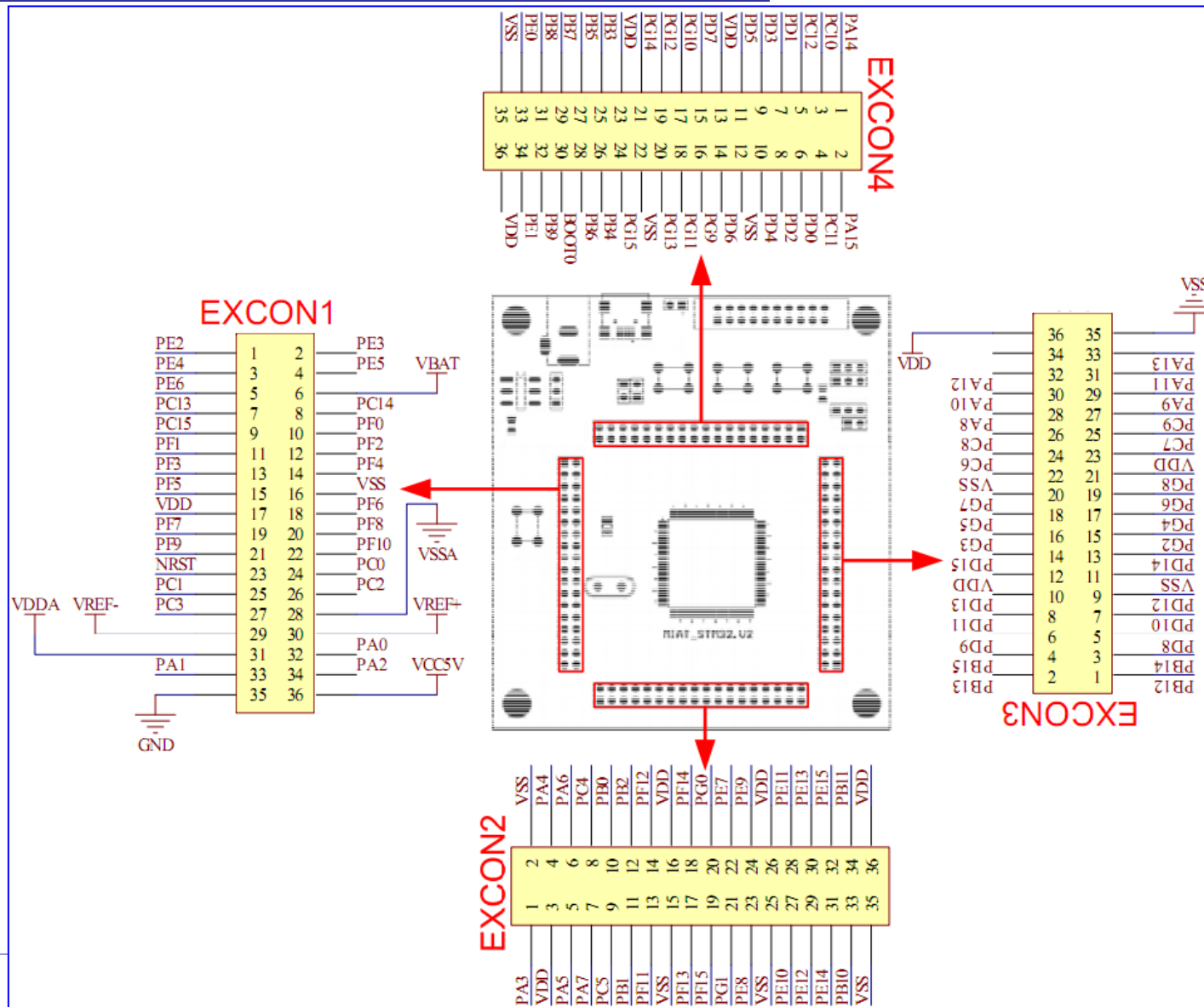
Lab 4. Connect 1 of 2

HDL Name	Pin Mapping	Description	MIAT STM32	MIAT IOB
c3x_clock50m	E16	miatc3x onboard		
c3x_dipsw0	E1	miatc3x onboard		
c3x_gpio_b1[0]	C2	miatc3x-JP1 1.C2		19. UART1_TX
c3x_gpio_b1[1]	D1	miatc3x-JP1 2.D1	PG10	
c3x_gpio_b1[2]	F2	miatc3x-JP1 3.F2	PD4	
c3x_gpio_b1[3]	F1	miatc3x-JP1 4.F1	PD5	
c3x_gpio_b1[4]	G2	miatc3x-JP1 5.G2	PE0	
c3x_gpio_b1[5]	G1	miatc3x-JP1 6.G1	PE1	
c3x_gpio_b1[6]	G5	miatc3x-JP1 7.G5	PD14	
c3x_gpio_b1[7]	B1	miatc3x-JP1 8.B1	PD15	
c3x_gpio_b1[8]	F3	miatc3x-JP1 9.F3	PD0	
c3x_gpio_b1[9]	J2	miatc3x-JP1 10.J2	PD1	
c3x_gpio_b1[10]	J1	miatc3x-JP1 13.J1	PE7	
c3x_gpio_b1[11]	K2	miatc3x-JP1 14.K2	PE8	
c3x_gpio_b1[12]	K1	miatc3x-JP1 15.K1	PE9	
c3x_gpio_b1[13]	L2	miatc3x-JP1 16.L2	PE10	
c3x_gpio_b1[14]	L1	miatc3x-JP1 17.L1	PE11	
c3x_gpio_b1[15]	N2	miatc3x-JP1 18.N2	PE12	

Lab 4. Connect 2 of 2

c3x_gpio_b1[16]	N1	miatc3x-JP1 19.N1	PE13	
c3x_gpio_b1[17]	P2	miatc3x-JP1 20.P2	PE14	
c3x_gpio_b1[18]	P1	miatc3x-JP1 21.P1	PE15	
c3x_gpio_b1[19]	K5	miatc3x-JP1 22.K5	PD8	
c3x_gpio_b1[20]	L4	miatc3x-JP1 23.L4	PD9	
c3x_gpio_b1[21]	R1	miatc3x-JP1 24.R1	PD10	
c3x_gpio_b1[22]	L3	miatc3x-JP1 25.L3	PF0	
c3x_gpio_b1[23]	N9	miatc3x-JP1 26.N9	PF1	
c3x_gpio_b1[24]	R10	miatc3x-JP1 27.R10	PF2	
c3x_gpio_b1[25]	T10	miatc3x-JP1 28.T10	PF3	
c3x_gpio_b1[26]	R11	miatc3x-JP1 31.R11	PF4	
c3x_gpio_b1[27]	P9	miatc3x-JP1 32.P9	PF5	
c3x_gpio_b1[28]	M10	miatc3x-JP1 33.M10	PF12	
c3x_gpio_b1[29]	N11	miatc3x-JP1 34.N11	PF13	
c3x_gpio_b1[30]	P11	miatc3x-JP1 35.P11	PF14	
c3x_gpio_b1[31]	N12	miatc3x-JP1 36.N12	PF15	
c3x_gpio_b1[32]	F14	miatc3x-JP1 37.F14	PG0	
c3x_gpio_b1[33]	A6	miatc3x-JP1 38.A6	PG1	

Lab 1. MIAT-STM32 Connector



Lab 4. Replace STM32F103 UART

```
PUTCHAR_PROTOTYPE
```

```
{  
    /* Write a character to the USART */  
    //USART_SendData(USARTx, (u8) ch);  
  
    /* Loop until the end of transmission */  
    //while(USART_GetFlagStatus(USARTx, USART_FLAG_TXE) == RESET)  
    //{  
    //}  
    //}  
  
    miatc3x_stx_senddata(ch);  
  
    return ch;  
}
```

USART Printf Example: retarget the C library printf function to the USART

Float point operation

> During Time: 2870 ns

Y: 104.008003

U: 71.389999

V: 193.156998

Fixed point operation

> During Time: 230 ns

Y: 105

U: 70

V: 193

Fixed point operation

> During Time: 380 ns

Y: 105

U: 70

V: 193

Note

☐ 講義與範例

- Day3 & Day5
- <http://www.csie.ncu.edu.tw/~tkyao/tahwa201007/>

☐ 軟體

- Altera QuartusII
- Altera ModelSim
- Keil RVMDK