

Advanced Computer Systems Architecture

Chip-Multiprocessors: Applications and Architectures

CSE 561M

Prof. Patrick Crowley

Plan for Today

- Questions
- Today's discussion

Objective

- Create simple receive and transmit microblocks
- Discuss simple_rpt project, which brings together packet receive, processing, and transmit
- Driving “real” traffic

MSF Operation

- MSF breaks incoming packets into *mpackets*, which are fixed-length segments (configurable as 64, 128 or 256 bytes)
- A given mpacket can be marked as the start of packet (SOP), end of packet (EOP), or both (if the packet fits within one mpacket).
- mpackets are stored in a buffer (RBUF and TBUF) at the MSF
- The microengines are responsible for re-assembling mpackets into packets

Configuring the MSF

- The MSF is configurable, allowing it to connect to different interface types
 - Ethernet, ATM, SONET, etc.
- All configuration takes place via software-controlled CSRs

Preparing the 2800 MSF Receive Control CSR

3	0	RX_EN_S
3	1	RX_EN_C
2	9	RESERVED
2	8	
2	7	SPI4_CHECKSUM_MODE
2	6	DATA_DIP4_DIS
2	5	FLWCTL_VPAR_TYPE
2	4	DATA_VPAR_TYPE
2	3	FLWCTL_VPAR_DIS
2	2	DATA_VPAR_DIS
2	1	FLWCTL_HPAR_DIS
2	0	DATA_HPAR_DIS
1	9	RSTAT_OV_VALUE
1	8	
1	7	RSTAT_OVERRIDE
1	6	DUPLEX_MODE
1	5	RX_CWRD_SIZE
1	4	STAT_CLOCK
1	3	
1	2	RSTAT_SELECT
1	1	RX_CALENDAR_MODE
1	0	RESERVED
9		CSIX_FREELIST
8		RESERVED
7		RBUF_ELE_SIZE_2
6		
5		RBUF_ELE_SIZE_1
4		
3		RBUF_ELE_SIZE_0
2		
1		RBUF_PARTITION
0		

Preparing the 2400 MSF Receive Control CSR

3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0																	
RX_EN[3:0]				RESERVED									RX_MODE		RX_WIDTH		RX_MPHY_EN		RX_MPHY_MODE		RX_MPHY_POLL_MODE		TX_CBUS_MODE		RESERVED		TX_CBUS_WIDTH		RX_MPHY_LEVEL		RESERVED		CSIX_FREELIST		RESERVED							RBUF_ELEMENT_SIZE		RESERVED	

Writing CSRs (2400)

```
//----- Set the receive control CSR in the MSF.  
// 2XXX PRM p. 464  
immed32($rx_ctl_xfer, RX_CONTROL_VAL1)  
immed32(addr, MSF_RX_CONTROL_ADDR)  
msf[write, $rx_ctl_xfer, addr, 0, 1],  
    ctx_swap[msf_rx_ctl_sig]  
  
immed32($rx_ctl_xfer, RX_CONTROL_VAL2)  
immed32(addr, MSF_RX_CONTROL_ADDR)  
msf[write, $rx_ctl_xfer, addr, 0, 1],  
    ctx_swap[msf_rx_ctl_sig]
```

Manipulating CSRs (2800)

```
//-----  
// 2800 Bit positions within the MSF transmit control CSR  
#define RX_EN_SPHY_BITPOS          30  
#define RBUF_ELE_SIZE_0_BITPOS     2  
  
#define RBUF_ELEM_SIZE             64  
#define RX_CONTROL_VAL             ((1 << RX_EN_SPHY_BITPOS) | \  
                                     ((RBUF_ELEM_SIZE/64)-1) << RBUF_ELE_SIZE_0_BITPOS))
```

Writing CSRs (2800)

```
//----- Set the receive control CSR in the MSF.  
immed32($rx_ctl_xfer, RX_CONTROL_VAL)  
immed32(addr, MSF_RX_CONTROL_ADDR)  
msf[write, $rx_ctl_xfer, addr, 0, 1],  
    ctx_swap[msf_rx_ctl_sig]
```

Receiving from the MSF

1. Microengine places a thread on the `RX_THREAD_FREELIST` at the MSF
2. The MSF signals the thread when it has an mpacket ready in an RBUF element, and writes that packet's *receive status words* – RBUF entry, EOP, SOP and status bits (i.e., error bits) -- into the thread's transfer registers
3. Upon receiving the signal, the microengine reads the receive status from the transfer registers
4. The microengine copies the data into memory, and notifies the MSF to release the RBUF entry

Registering a Thread on an MSF Free List

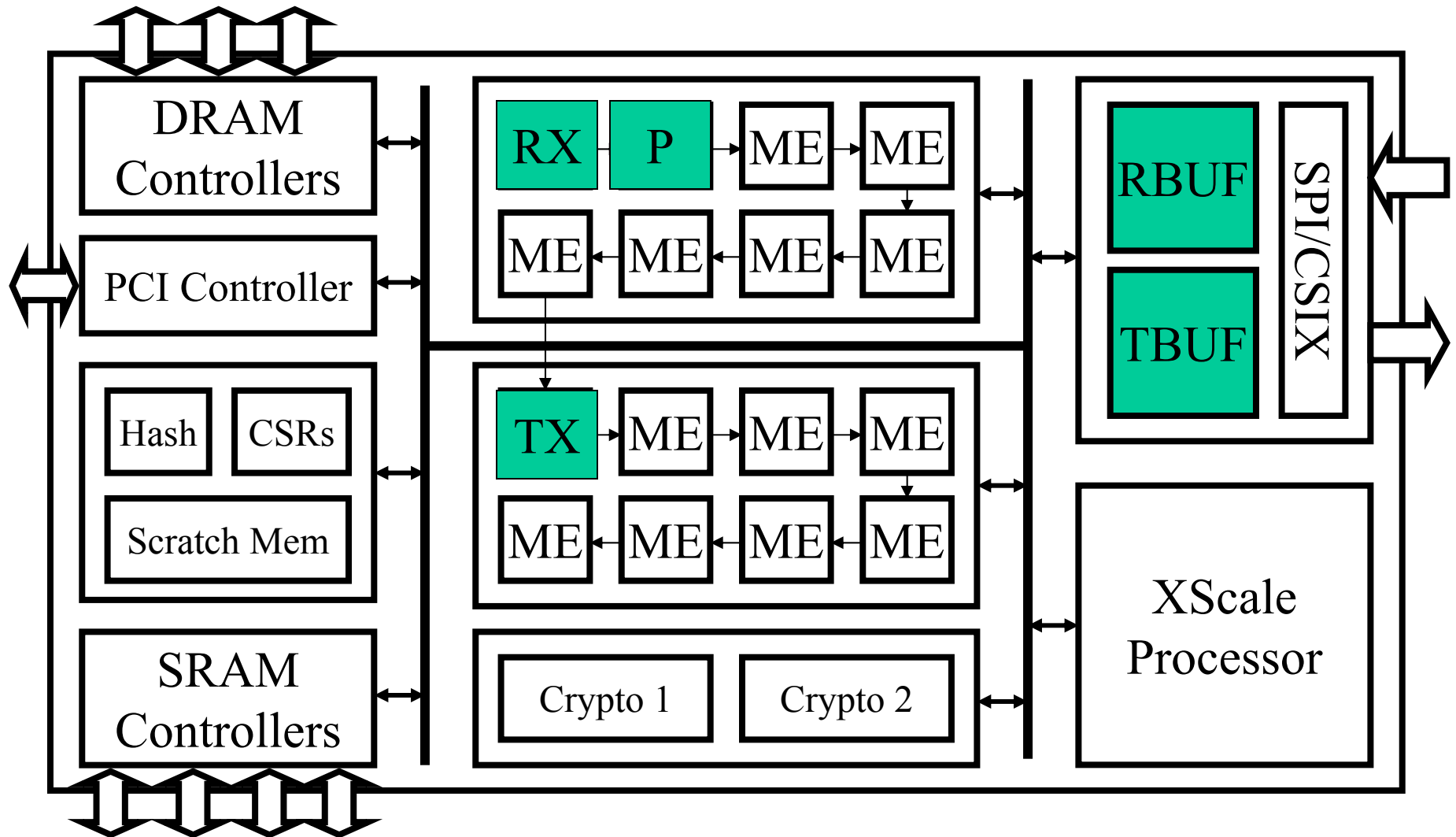
3	3	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	0	9	8	7	6	5	4	3	2	1	0
RESERVED														SIG_NO				ME_CLUS	RESERVED	ME_NO		THD		XFER_REG						

- `RX_THREAD_FREELIST_#` MSF CSR must be programmed
 - Only 16 bits, so can use a `fast_wr` command

Transmitting through the MSF

1. Find a free TBUF
2. Move mpacket into that TBUF
3. Write (and validate) control words for that TBUF (eop, sop, valid)

Sample RPT Project Setup



Sample Project

1. Discuss `simple_rpt`

Simulating with Packets

- You must create MSF devices and streams, and map traffic
- In simulation, you must also issue this command on the IDE's command line
 - `ps_start_packet_receive() ;`
- See Development Tools User Guide for details

Assignment

- Thursday
 - J&K: Read Ch. 6
- Tuesday
 - J&K: Read Ch. 7
 - **Commentary:** Download the sample `simple_rpt` project and implement a different process step (currently it counts packets seen). In your commentary, describe your processing step and include your code.