SoC Lab

January 4, 2024

Final Project Presentation

Workload Optimized SoC

Group 6

Name : 劉冠亨、張力元、謝言鼎

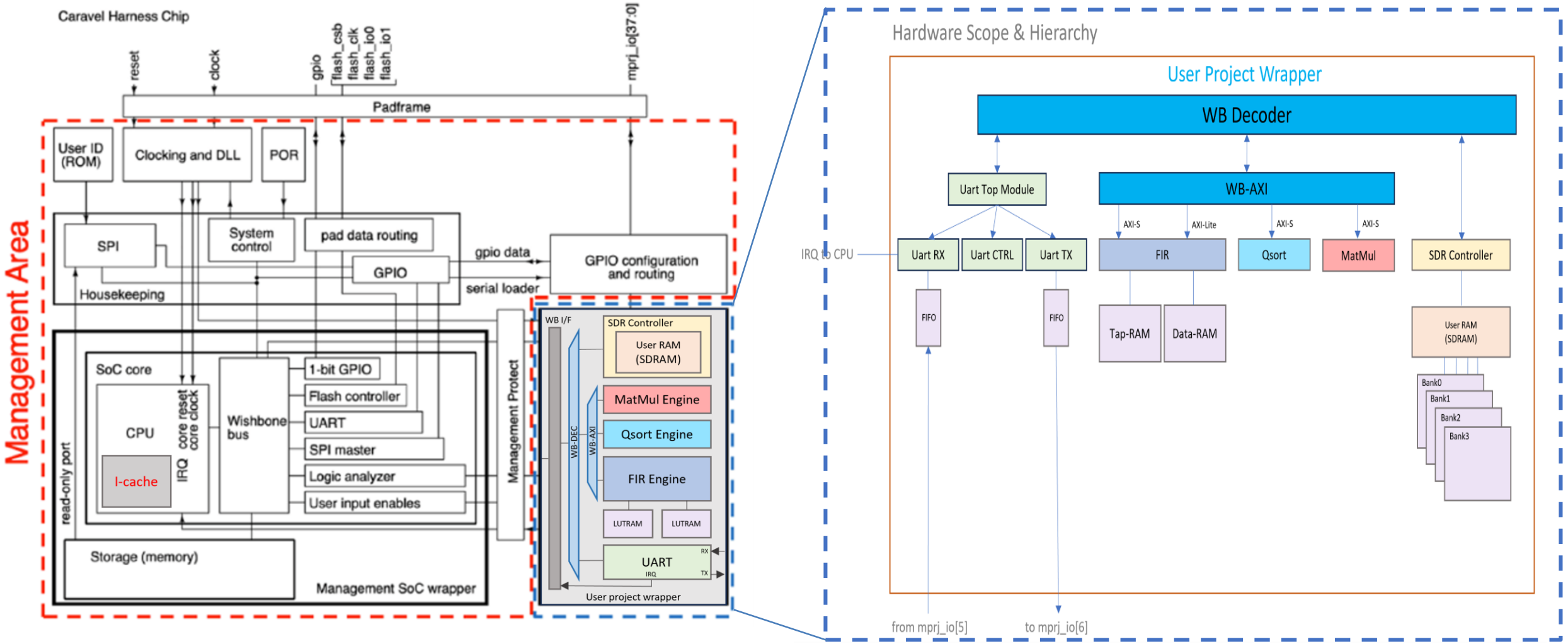
Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Block Diagram



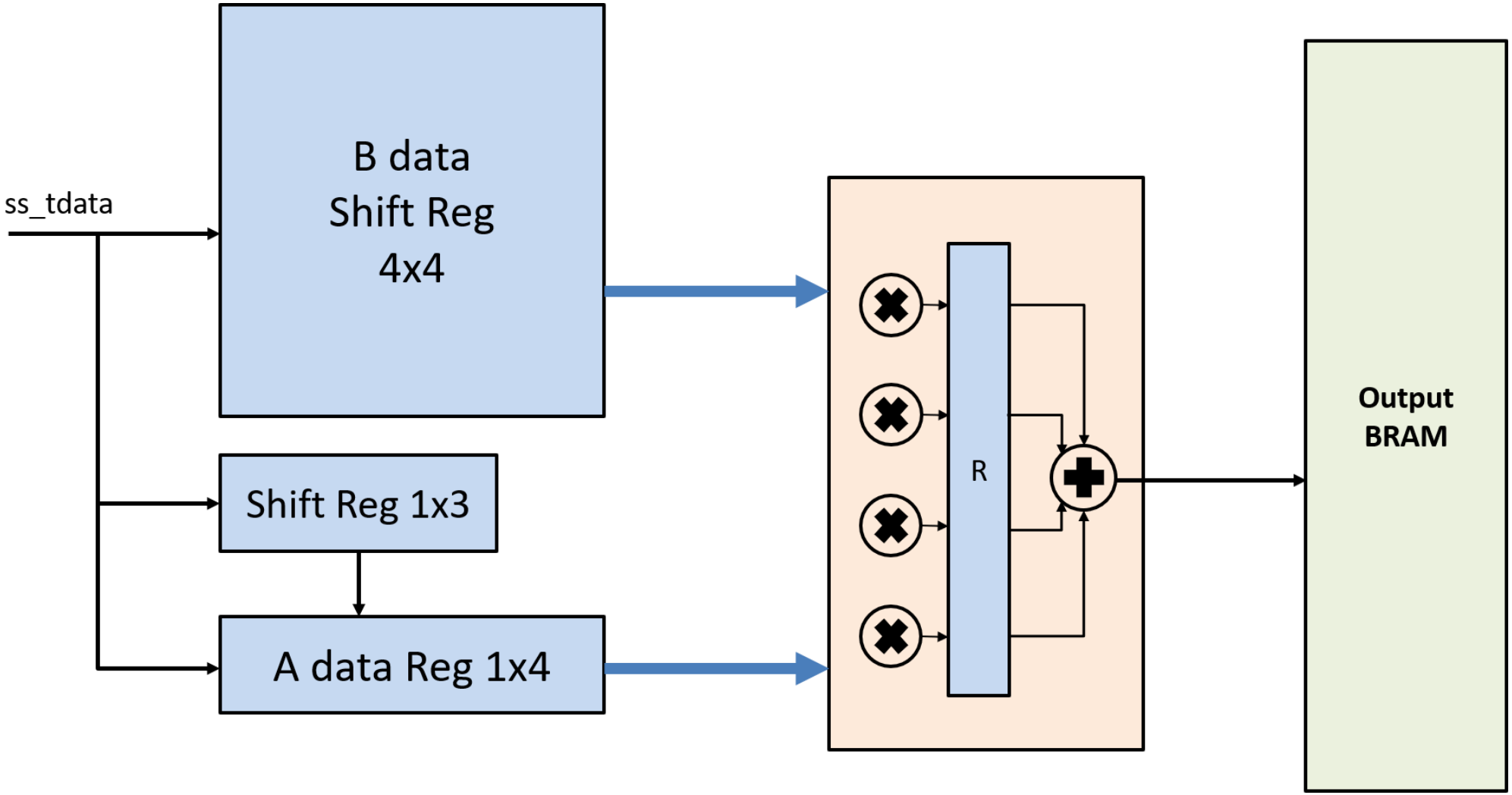
Configuration Register Address Map

Module	Base Address	Offset	Description
UART	0x3000_0000	0x00	Uart rx port
		0x04	Uart tx port
		0x08	Status of UART IP
		0x0C	Baudrate config
MatMul	0x3000_0100	0x00	Status of MatMul Engine
		0x80	Matrix A input
		0x84	Matrix B input
		0x8C	Matrix R output
Qsort	0x3000_0200	0x00	Status of Qsort Engine
		0x80	Input data
		0x84	Output data
FIR	0x3000_0300	0x00	Status of FIR engine
		0x10 – 0x13	Data length
		0x40 – 0x7F	Tap coefficients
		0x80 – 0x83	Input x[n]
		0x84 – 0x87	Output y[n]
SDRAM	0x3800_0000		Execution memory that stores firmware code

Outline

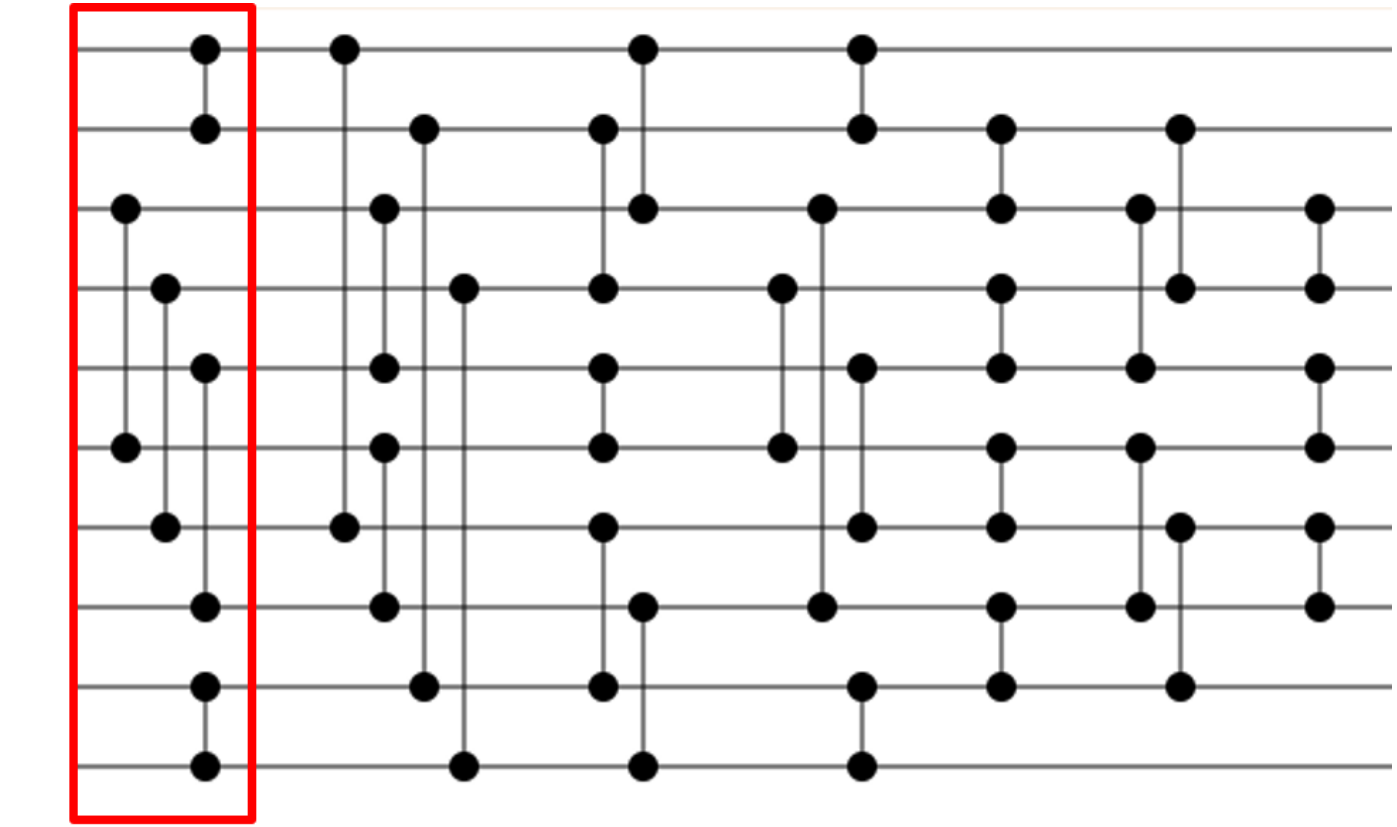
- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Matrix Multiplication Engine



Sorting Engine

- Use the optimal sorting network
- 5 comparators, 7 cycles

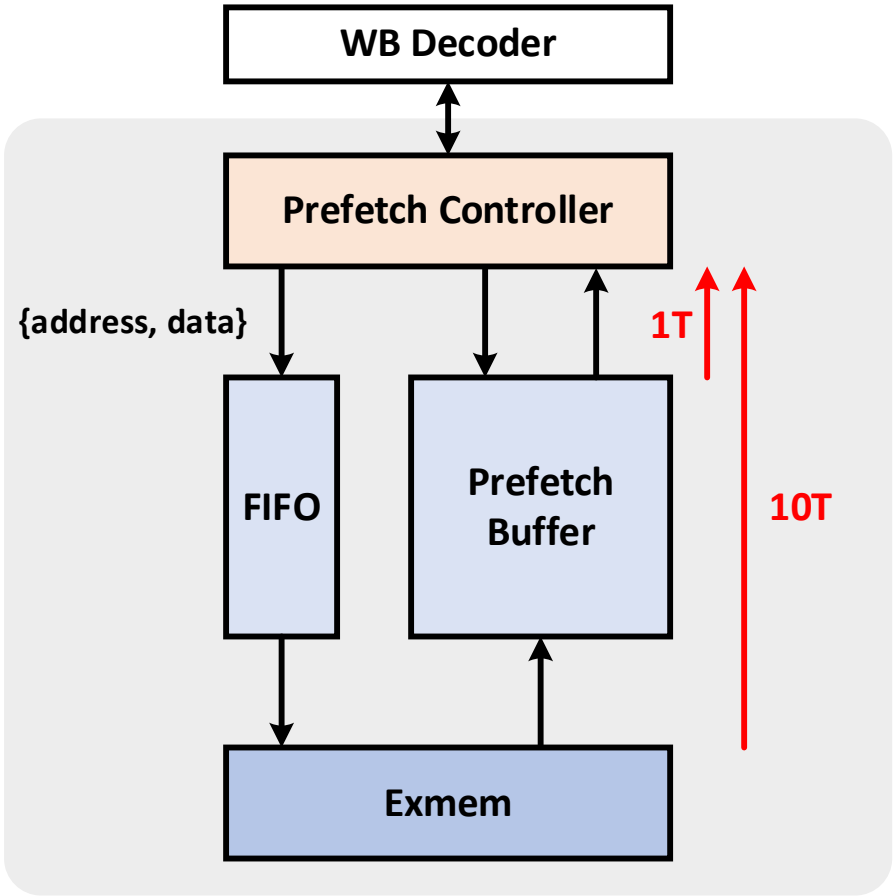


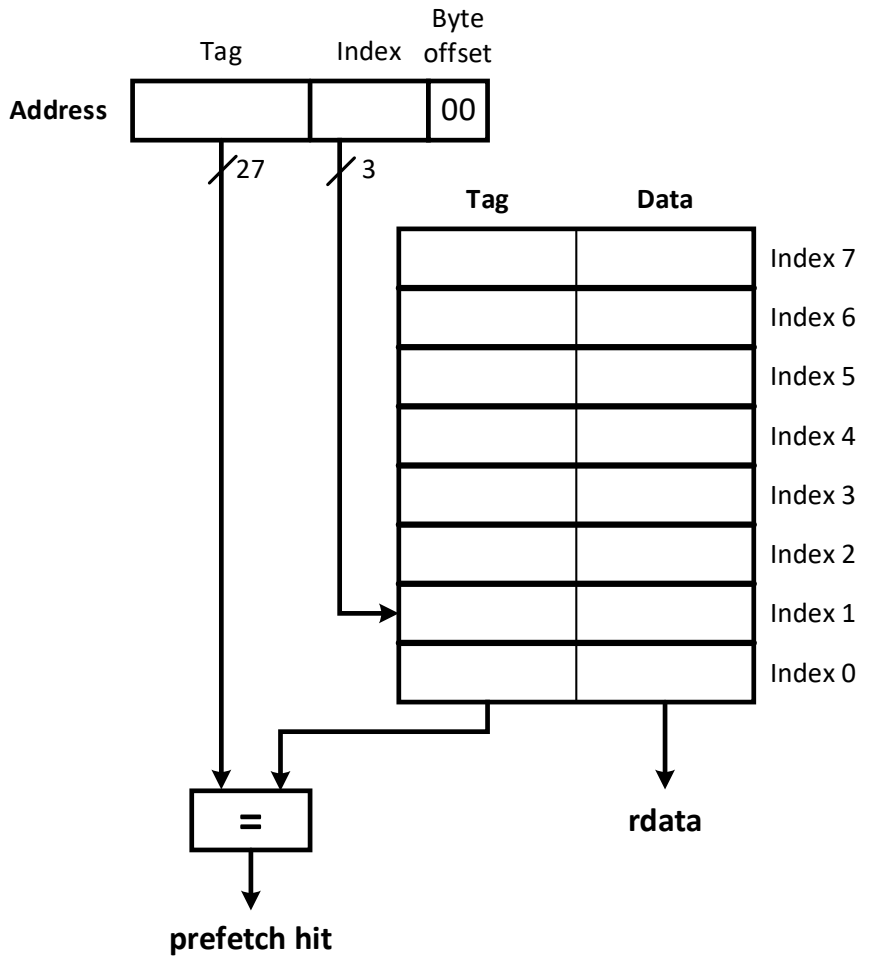
Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Pipelined Execution Memory with Prefetch Controller

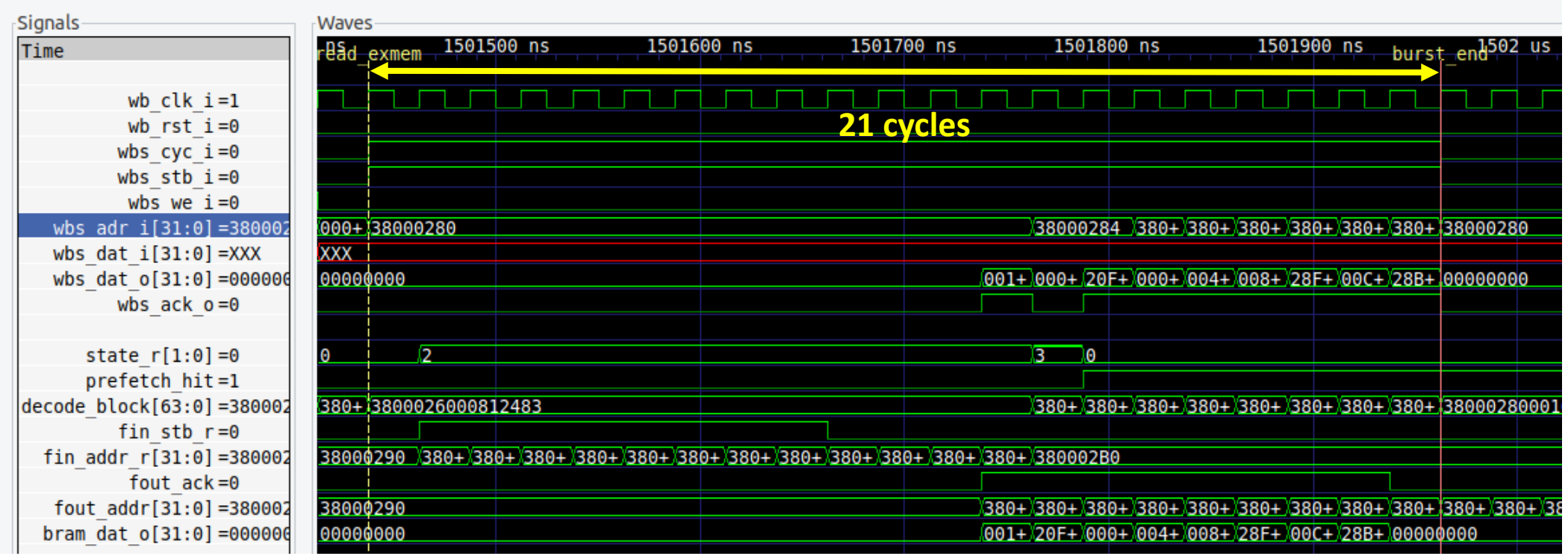
- Use index to choose the corresponding block in the prefetch buffer
- In the case of a prefetch miss (i.e., non-matching tags), initiate a burst of 8 pipelined requests into the FIFO





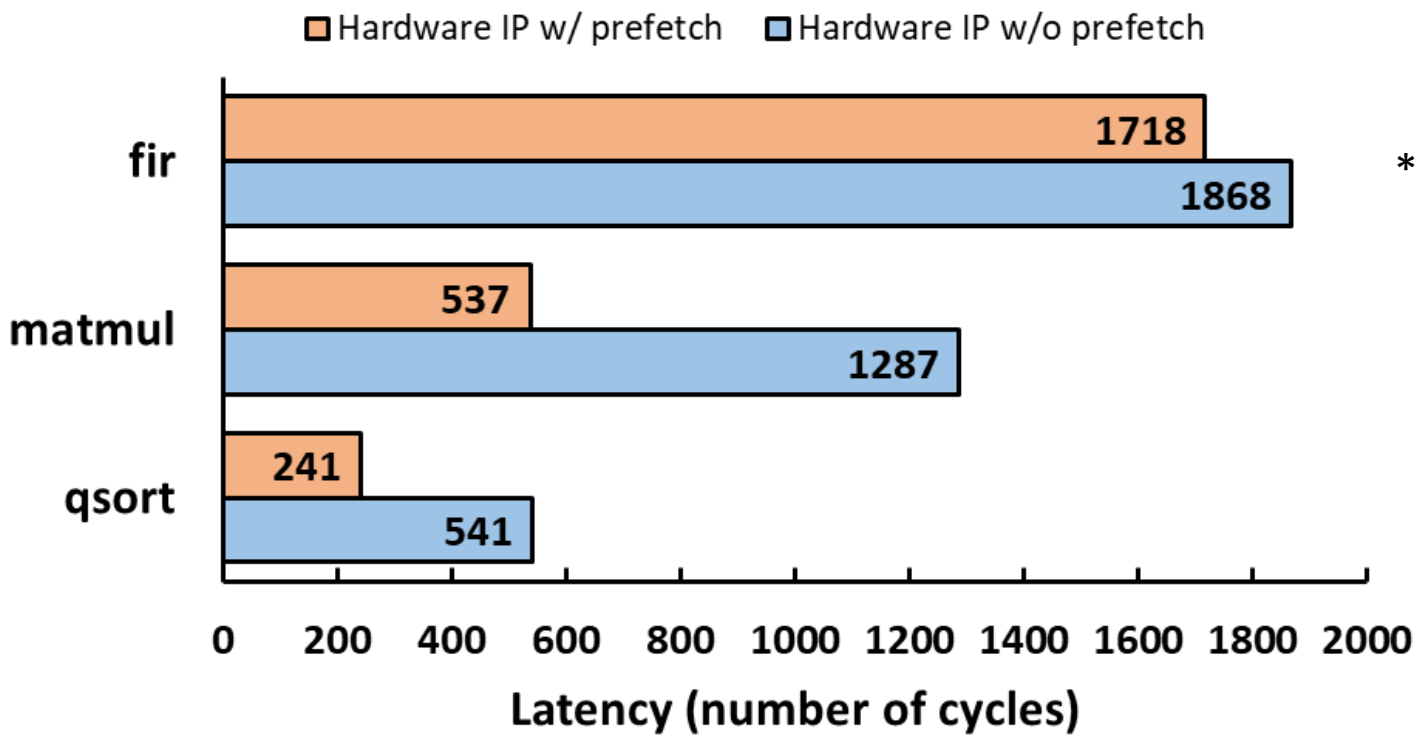
Pipelined Execution Memory with Prefetch Controller

- Based on observations, the CPU usually reads 8 instructions consecutively from the execution memory
- With prefetching, the latency of reading 8 instructions can be reduced from 80 cycles to 21 cycles



Pipelined Execution Memory with Prefetch Controller

- With a pipelined execution memory and prefetch controller, the overall latency decreases from 3696 cycles to 2496 cycles, resulting in a reduction of 1200 cycles



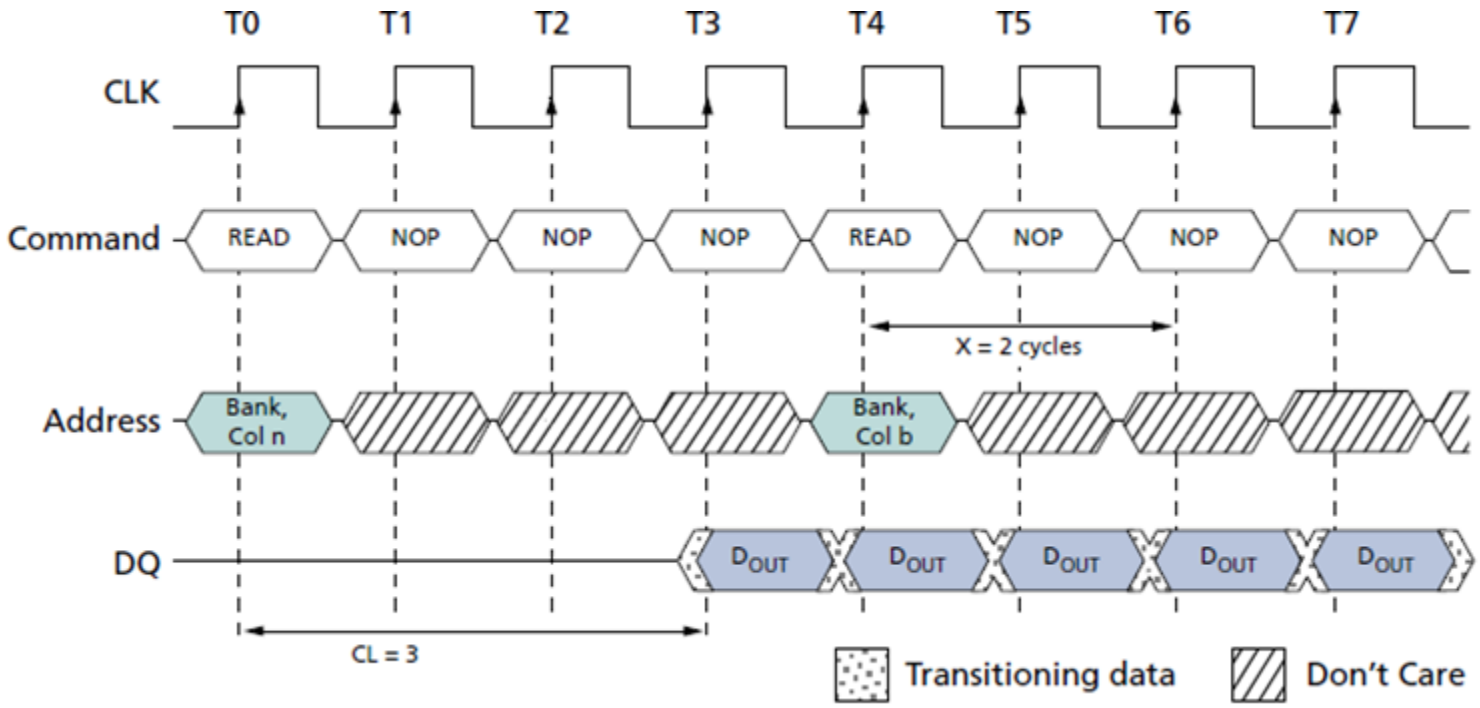
*FIR data length = 64

Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

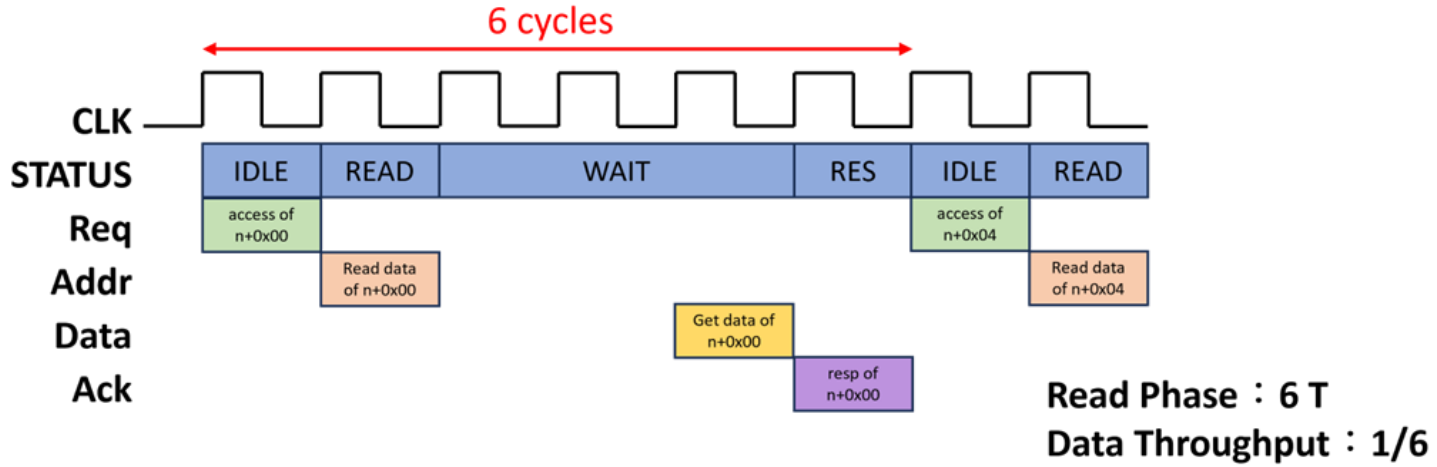
SDRAM Device with Read-burst Mode

- Burst length = 8



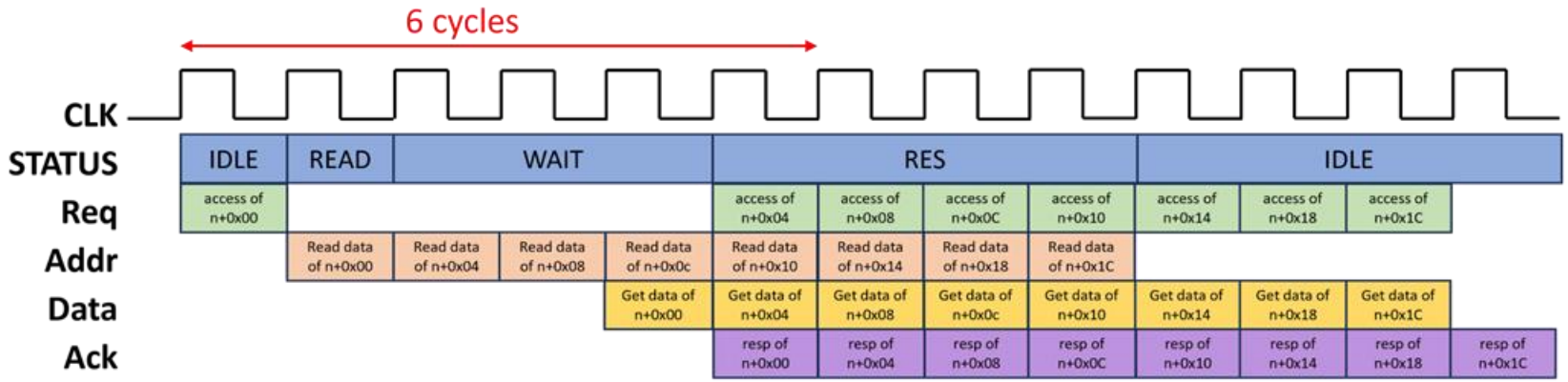
SDRAM with Prefetch Controller

- Origin SDRAM access diagram



Read Phase：6 T
Data Throughput：1/6

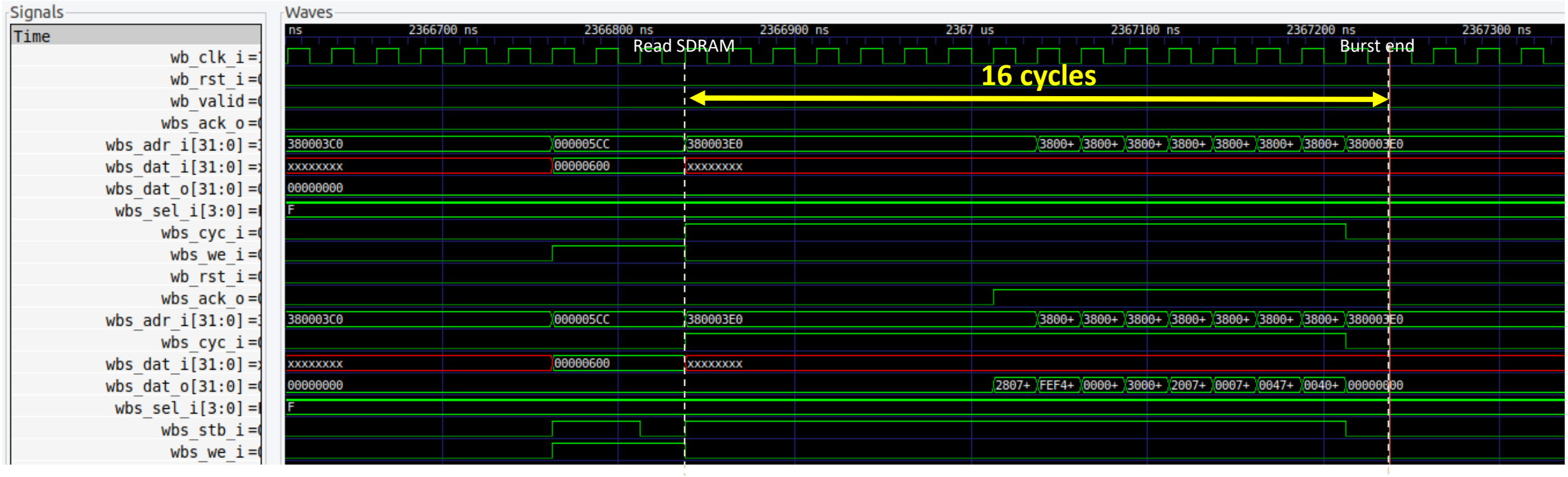
- Optimized SDRAM access diagram (with burst mode SDRAM device)



Read Phase：6 T
Data Throughput：8/13

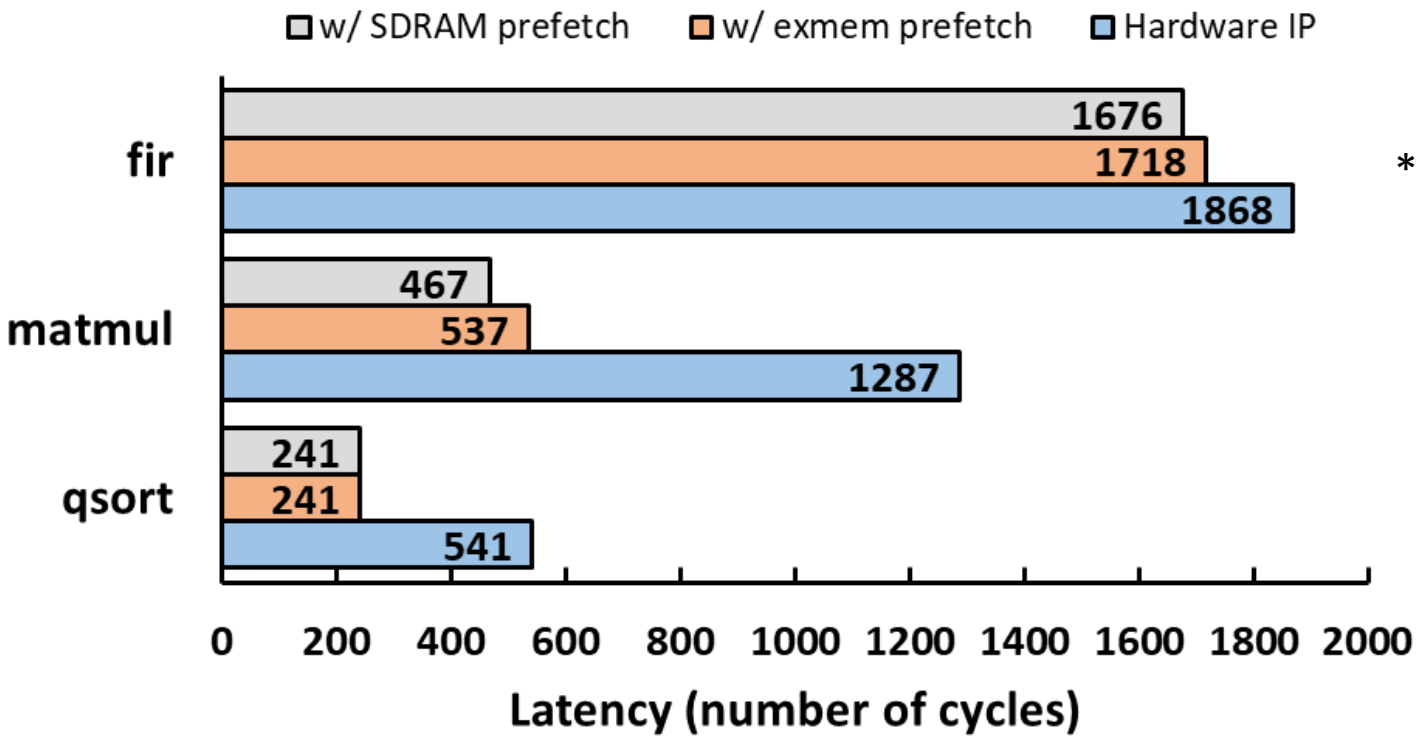
SDRAM with Prefetch Controller

- With prefetch scheme and burst mode SDRAM, the latency of reading 8 instructions can be reduced from 80 cycles to 16 cycles



SDRAM with Prefetch Controller

- With a burst mode SDRAM and prefetch controller, the overall latency decreases from 3696 cycles to 2384 cycles, resulting in a reduction of 1312 cycles



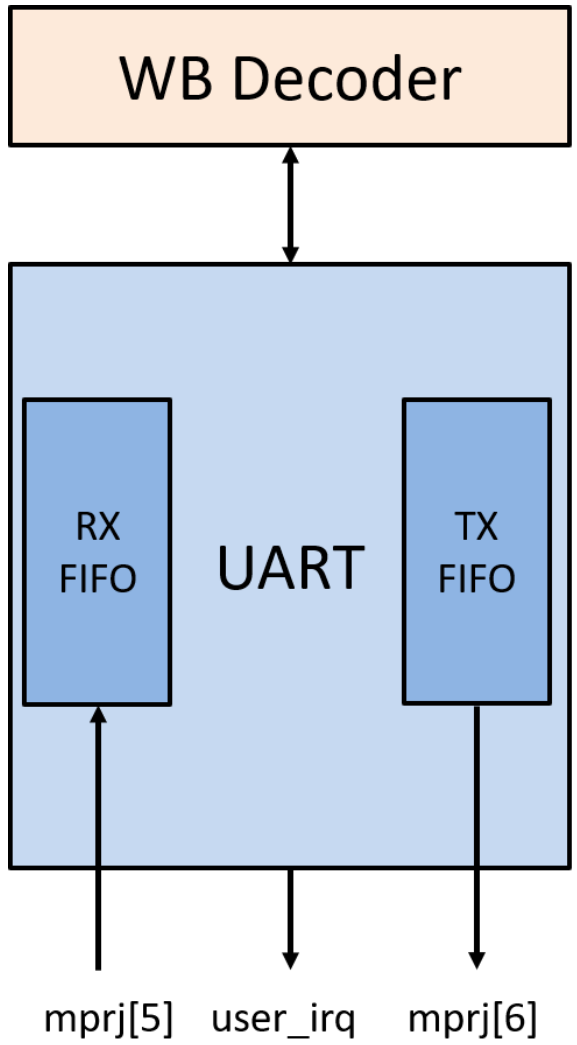
*FIR data length = 64

Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

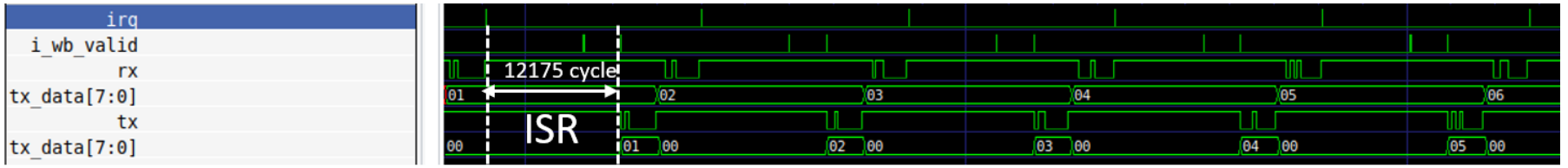
FIFO on Rx/Tx

- FIFO depth: 8
- Change baud rate to 115200
- UART sends IRQ signal to CPU when RX FIFO is full
- ISR finishes after writing TX data to TX FIFO

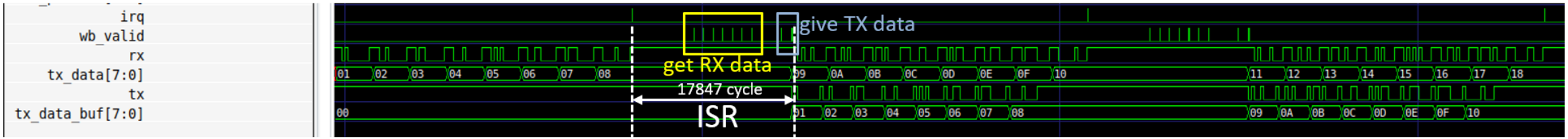


FIFO on Rx/Tx

- Baseline:
  - One interrupt per data
  - Rx/Tx no overlap
  - (Data transmission time * 2) + ISR time = 4147*2 + 12175 = 20509 cycles per data



- FIFO:
  - One interrupt per 8 data
  - Rx/Tx overlap
  - ((Data transmission time * 8) + ISR time)/8 = (4147*8 + 17847)/8 = 6378 cycles per data



3x time improvement

Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Compilation Flag

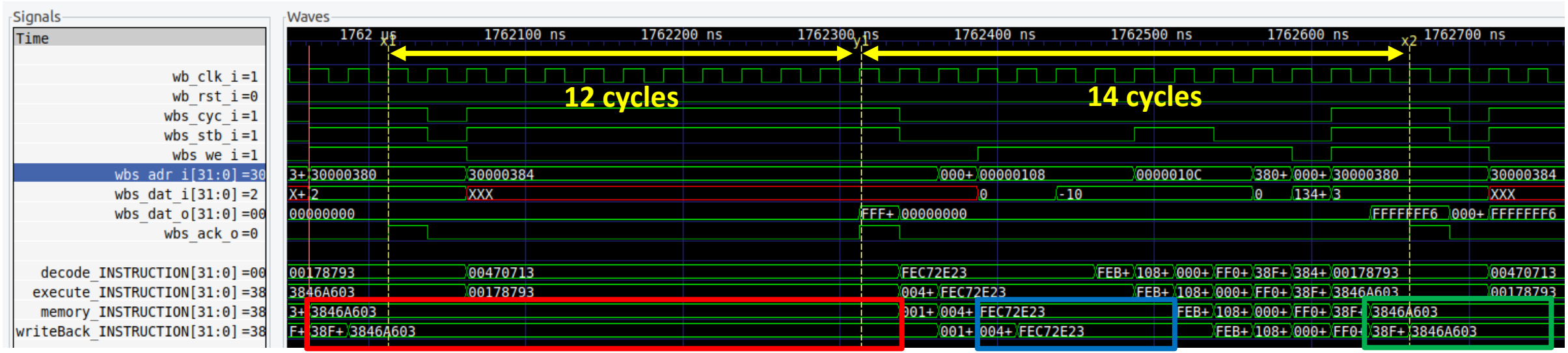
- Compile with -O3 flag
  - Retain loops in the instruction cache

fir.c

```
for(int i = 1; i <= SEQ_LEN; i = i + 1) {
    FIR_X = i;
    outputsignal[i-1] = FIR_Y;
}
```

counter_la_all.out

```
3800008c: 38f6a023          sw  a5,896(a3) # 30000380 <_esram_rom+0x1ffffcd8>
38000090: 3846a603          lw  a2,900(a3)
38000094: 00178793          addi  a5,a5,1
38000098: 00470713          addi  a4,a4,4
3800009c: fec72e23          sw  a2,-4(a4)
380000a0: feb796e3          bne a5,a1,3800008c <fir+0x88>
```



Revised FIR Firmware

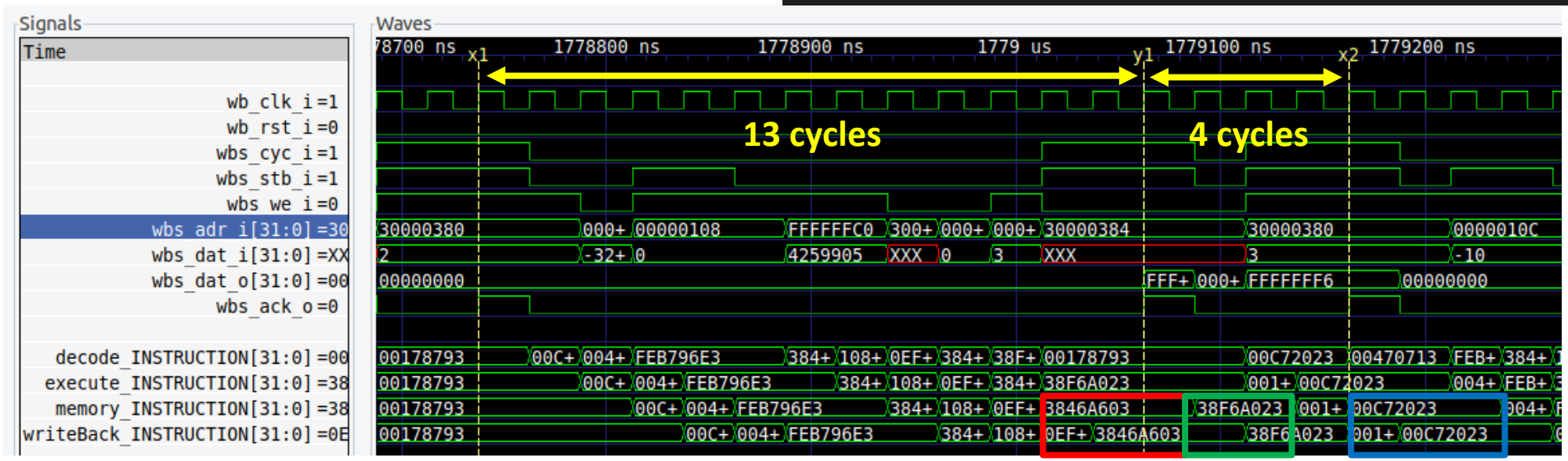
- Execute instructions in the following order
  - Read y from FIR engine to CPU (lw)
  - Write x to FIR engine (sw)
  - Store y into user memory (sw)

```
int fir_output;
FIR_X = 1;
for(int i = 2; i <= SEQ_LEN; i = i + 1) {
    fir_output = FIR_Y;
    FIR_X = i;
    outputsignal[i-2] = fir_output;
}
outputsignal[SEQ_LEN-1] = FIR_Y;
```

fir.c

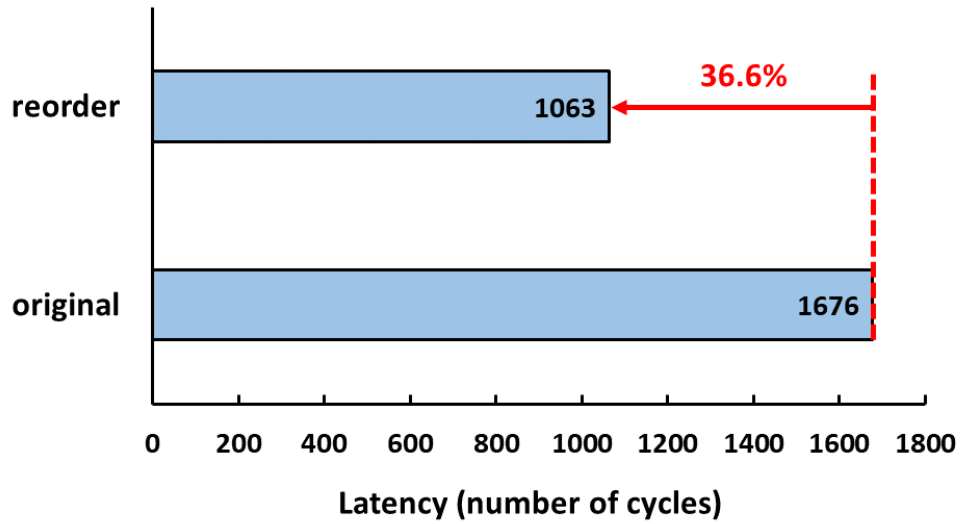
counter_la_all.out

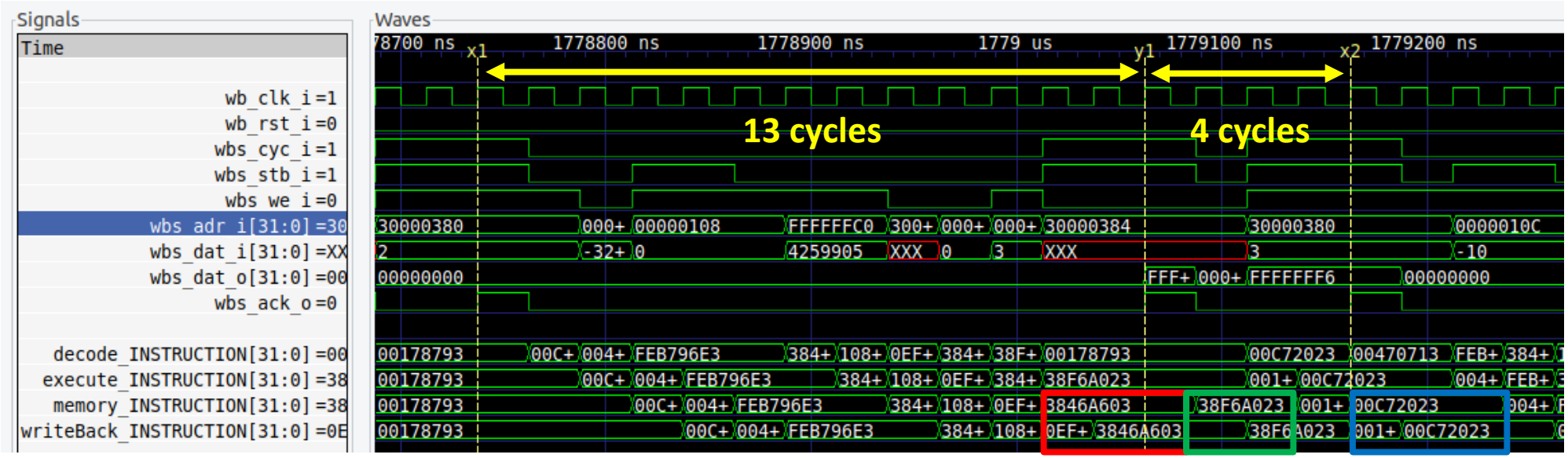
```
38000098: 3846a603          lw  a2,900(a3) # 30000384 <_esram_rom+0x1ffffcdc>
3800009c: 38f6a023          sw  a5,896(a3)
380000a0: 00178793          addi  a5,a5,1
380000a4: 00c72023          sw  a2,0(a4)
380000a8: 00470713          addi  a4,a4,4
380000ac: feb796e3          bne a5,a1,38000098 <fir+0x94>
380000b0: 3846a783          lw  a5,900(a3)
```



Revised FIR Firmware

- Execute instructions in the following order
  - Read y from FIR engine to CPU (lw)
  - Write x to FIR engine (sw)
  - Store y into user memory (sw)





Outline

- System Overview
- Hardware Accelerator
- Pipelined Execution Memory with Prefetch Controller
- SDRAM with Prefetch Controller
- UART with I/O FIFO
- Firmware Optimization
- FPGA Implementation

Timing and utilization report

- Setup time and hold time slack (40MHz)

Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	1.339 ns	Worst Hold Slack (WHS):	0.032 ns	Worst Pulse Width Slack (WPWS):	11.250 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	22179	Total Number of Endpoints:	22179	Total Number of Endpoints:	8542

All user specified timing constraints are met.

- Utilization report including FF, LUT and BRAM

Baseline (Lab6)

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs	5364	0	0	53200	10.08
LUT as Logic	5176	0	0	53200	9.73
LUT as Memory	188	0	0	17400	1.08
LUT as Distributed RAM	18	0			
LUT as Shift Register	170	0			
Slice Registers	6195	0	0	106400	5.82
Register as Flip Flop	6195	0	0	106400	5.82
Register as Latch	0	0	0	106400	0.00
F7 Muxes	169	0	0	26600	0.64
F8 Muxes	47	0	0	13300	0.35

Site Type	Used	Fixed	Prohibited	Available	Util%
Block RAM Tile	8	0	0	140	5.71
RAMB36/FIFO*	5	0	0	140	3.57
RAMB36E1 only	5				
RAMB18	6	0	0	280	2.14
RAMB18E1 only	6				

Our work

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs	8743	0	0	53200	16.43
LUT as Logic	8491	0	0	53200	15.96
LUT as Memory	252	0	0	17400	1.45
LUT as Distributed RAM	82	0			
LUT as Shift Register	170	0			
Slice Registers	9702	0	0	106400	9.12
Register as Flip Flop	9350	0	0	106400	8.79
Register as Latch	352	0	0	106400	0.33
F7 Muxes	204	0	0	26600	0.77
F8 Muxes	74	0	0	13300	0.56

Site Type	Used	Fixed	Prohibited	Available	Util%
Block RAM Tile	14	0	0	140	10.00
RAMB36/FIFO*	11	0	0	140	7.86
RAMB36E1 only	11				
RAMB18	6	0	0	280	2.14
RAMB18E1 only	6				

Jupyter notebook

- FPGA prototyping

```
In [21]: asyncio.run(async_main())
Start Caravel Soc
matrix multiplication start
0x3e
0x44
0x4a
0x50
matrix multiplication done, time: 0.0008633136749267578 seconds
quick sort start
0x28
0x37d
0x9ed
0xa6d
0xca1
0x10ab
0x120e
0x1631
0x1787
0x2371
quick sort done, time: 0.0005624294281005859 seconds
fir start
0x0
0xfff6
0xffe3
0xffe7
0x23
0x9e
0x151
0x2dc
0x393
0x44a
fir done, time: 0.0011889934539794922 seconds
uart start
uart done, time: 4.863739013671875e-05 seconds
Waitting for interrupt
hello
```

Quality of Result (QoS)

- Workload latency

*FIR data length = 64

Number of cycles	Baseline	Our work	Improvement
Qsort	26185	241	108.6X
MatMul	74458	467	159.4X
FIR	148821	1063	140.0X
Total	249464	1771	140.8X

- UART latency (512 characters)
  - Number of cycles = 408,367
  - Baud rate = 115,200
  - QoR metric = 408,367 * 25ns – 512 * (1 / baud rate) = 5.764 ms

	Latency (cycles*period)	Metric (ms)	Improvement
UART	1313138*25ns	28.384	4.92X
UART w/ FIFO	408367*25ns	5.764	