Justin VanWinkle
Portfolio 1

Part 1:
Config 1: (460 * 360)pixels * 4 bits * 3 colors / (1byte/8bits) = 248,400 bytes
Config 2: (720 * 480)pixels * 4 bits * 3 colors / (1byte/8bits) = 518,400 bytes

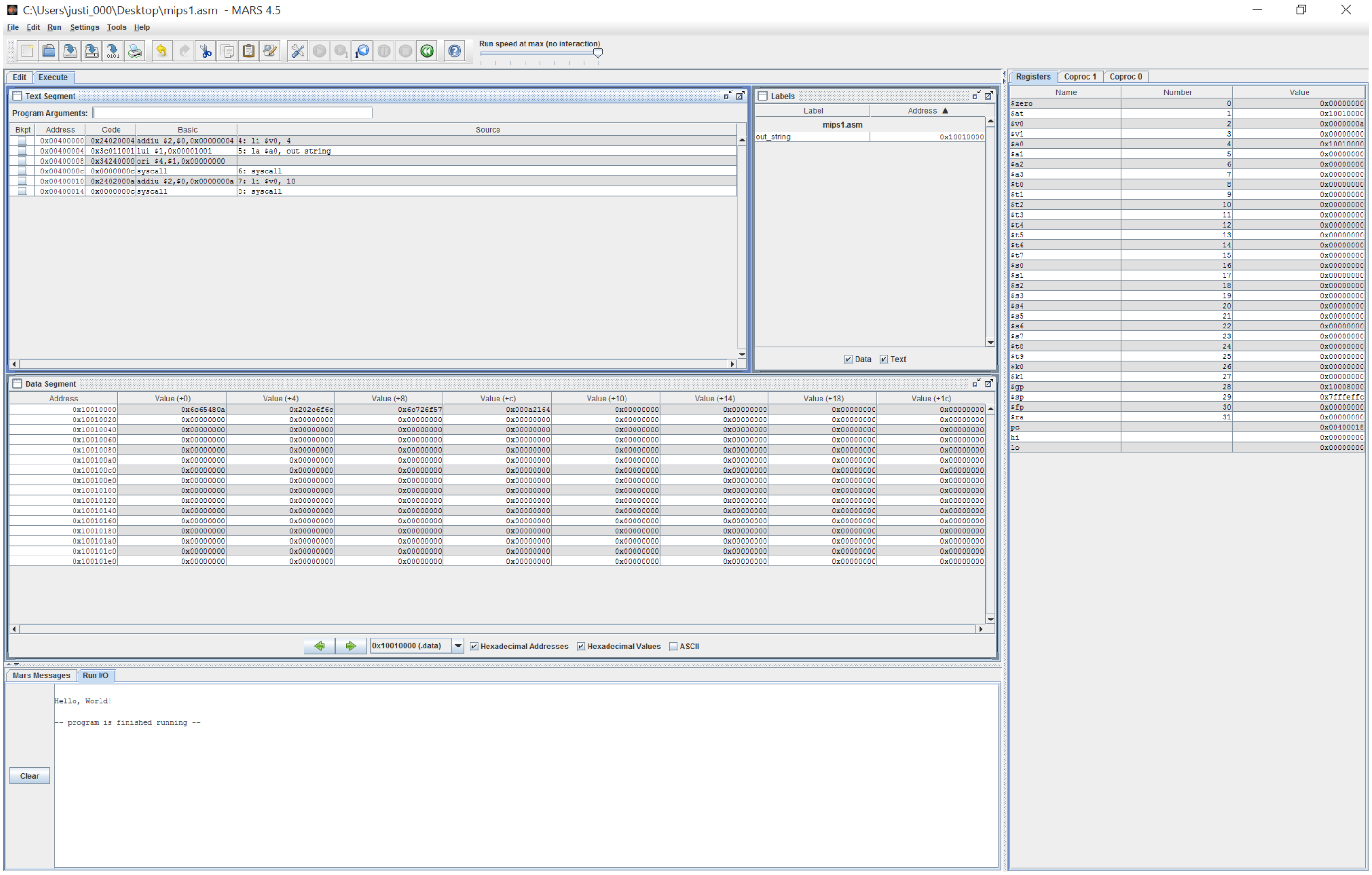
datarate: 8frames/second * 256KB = 2048KB/second = 0.002048GB/second
Total hours recording: 64GB / 0.002048GB/sec * (1min/60sec) * (1h/60min) = approx 8.68055556 hours

datarate: 4proc * 128KB/(sec*proc) = 512KB /sec
required capacity: (512KB / sec) * (60sec / min) * (60min / h) * (24h / day) * 3day = 132,710,400 KB

2 month backup: 6centers * 64servers * 128nodes * 32TB = 1,572,864 TB
2 years worth of 2 month backups: 1,572,864 TB * 6backups/year * 2 years = 18,874,368 TB

Part2 How processor and memory interact:
Accessing data: In the event of a processor requesting data from memory, the processor will set the address via the address bus which will cause the memory to return the data in that address to the processor via the data bus.
Storing Data: In the event of a processor writing data to the memory, the processor will set the address bus as well as the data bus, causing the memory to store the data on the data bus at the address location specified on the address bus.
Adding Data: If a processor needs to add two values, it will access two separate memory locations to get their data and then the processor will add the data before storing it back in memory or performing another operation on the result.

Part 3:



C:\Users\justi_000\Desktop\mips1.asm - MARS 4.5

File Edit Run Settings Tools Help
Run speed at max (no interaction)
Edit Execute
Text Segment
Program Arguments:

Bkpt	Address	Code	Basic	Source
□	0x00400000	0x24020004	addiu $2,$0,0x00000004	4: li $v0, 4
□	0x00400004	0x3c011001	lui $1,0x00001001	5: la $a0, out_string
□	0x00400008	0x34240000	ori $4,$1,0x00000000	
□	0x0040000c	0x0000000c	syscall	6: syscall
□	0x00400010	0x2402000a	addiu $2,$0,0x0000000a	7: li $v0, 10
□	0x00400014	0x0000000c	syscall	8: syscall

Labels

Label	Address ▲
mips1.asm	
out_string	0x10010000

☑ Data ☑ Text

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0x6c65480a	0x202c6f6c	0x6c726f57	0x000a2164	0x00000000	0x00000000	0x00000000	0x00000000
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

0x10010000 (.data) ☑ Hexadecimal Addresses ☑ Hexadecimal Values □ ASCII

```
Hello, World!

-- program is finished running --
```

Registers | Coproc 1 | Coproc 0

Name	Number	Value
$zero	0	0x00000000
$at	1	0x10010000
$v0	2	0x0000000a
$v1	3	0x00000000
$a0	4	0x10010000
$a1	5	0x00000000
$a2	6	0x00000000
$a3	7	0x00000000
$t0	8	0x00000000
$t1	9	0x00000000
$t2	10	0x00000000
$t3	11	0x00000000
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00000000
$s1	17	0x00000000
$s2	18	0x00000000
$s3	19	0x00000000
$s4	20	0x00000000
$s5	21	0x00000000
$s6	22	0x00000000
$s7	23	0x00000000
$t8	24	0x00000000
$t9	25	0x00000000
$k0	26	0x00000000
$k1	27	0x00000000
$gp	28	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00000000
pc		0x00400018
hi		0x00000000
lo		0x00000000