Zak Rowland
10/29/2019
CST337
Lab 4 answers

3.

- Resetting the device did not change any of the RAM.

Global variable	Virtual address	Value
a	0x9D000D38	CST 337 Lab 4\\0
b	0x800002B4	UUUUUUUUUUUUUUUUUUUUUUU
c	0x9D000D48	Initialized Constant String\0
d	0x800002CC	UUUUUUUUUUUUUUUUUUUUUUU
gc	0x800002AC	0x55555555
gd	0x800002B0	0x55555555
ge	0x80000280	0x55555555

- Global constant addresses are located in Program Flash under KSEG0, and global variable addresses are located in RAM under KSEG0.

4.

- The initial value of sp, fp, and gp is 0.
- The address of the PC just after reset is confirmed to be 0xBFC00000. This address is located in Boot Flash under KSEG1. MIPS Volume III indicates that this address is the exception vector base address for the reset, soft reset, and NMI exceptions.

5.

- The final value of the sp register is 0x8007FFF8, and the final value of the gp register is 0x80008270. The sp register is pointing to RAM under KSEG0, and the gp register is pointing to RAM under KSEG0.

6.

- Neither sp or gp changed during the run to main.
- The global values b, d, gc, gd, and ge changed.
- gd was initialized to 0 even though the code didn't request it.
- Data is initialized using the linker-generated data-initialization template.

7.

- The local variables that already have their initial values contain "static" in their declarations.
- The variables that are located in Program Flash contain are constants.
- I think these values were initialized upon startup.

8. Skip.

Zak Rowland
10/29/2019
CST337
Lab 4 answers

9.

Non-static local variable	Address
ic	0x8007FF90
id	0x8007FF94
ie	0x8007FFB0
icf	0x8007FF98
Lc	0x8007FFA0
Ld	0x8007FFA8
Le	r02 (CPU)
la	0x8007FFC0

- The sp and fp registers have values close to these addresses (0x8007FF80.)
- The stack.
- Const local variables technically aren't constant because they reside in RAM. When trying to change these variables using their labels the compiler won't let you, but using a pointer to its address in RAM will allow changes.

10.

- The data used to initialize ic is stored in two parts in the two separate instructions that load from immediate values. The upper 16 bits are located at 0x9D000658, and the lower 16 bits are located at 0x9D00065C.
- These addresses are located in Program Flash under KSEG0.

11.

- The first byte of the string data used to initialize la is stored at 0x9D000D64.
- This address is located in Program Flash under KSEG0.
- The initialization sets up pointers to the different words of data that make up the string then stores them. This seems to be a good idea because this method doesn't have to load a copy of another strings data like strcpy does.

12.

- The address of the data used to initialize b is 0x9D000254, d is 0x9D00026C, and ge is 0x9D0002E8.
- These initializers are located in Program Flash under KSEG0.
- The addresses are not sequential because constants are stored in their own area of memory and of a fixed size. In the case of b and d, the memory has to account for the ability to be overwritten.
- Using the const declaration is more efficient because it only reserves the memory it needs. In the case of b, 19 extra words of space were reserved in Program Flash and RAM even though they aren't used.

Zak Rowland
10/29/2019
CST337
Lab 4 answers

13.

- The problem with this strcpy is that some data is lost and the string isn't null terminated, this is a buffer overflow.
- The extra data overwrites whatever comes after the allocated memory in RAM.
- strcpy() looks for a NULL character rather than length.
- The debugger determines how long strings are by the memory that is allocated to them.
- I think b would overwrite the beginning of d, but d will no longer be NULL terminated.
- A runtime exception occurred because b isn't NULL terminated and strcpy requires that NULL character to finish.

14.

- Starting address: 0x9D000648
- Ending address: 0x9D0008A0

15.

RAM Section	Starting Address	KSEG (0/1)	Label	Address
.sdata	0x8000_0280	0		
			ge	0x8000_0280
			isf	0x8000_0284
.sbss	0x8000_02AC	0		
			gc	0x8000_02AC
			gd	0x8000_02B0
.data	0x8000_02B4	0		
			b	0x8000_02B4
			d	0x8000_02CC
heap	0x8000_0338	N/A		
stack	0x8000_0360	N/A	Bottom of the stack	0x8000_0360
			Top of the stack	0x8007_FFF8

Zak Rowland
10/29/2019
CST337
Lab 4 answers

Flash Section	Starting Address	KSEG (0/1)	Label	Address
.text.main_entry	0x9D00_0110	0		
.app_excpt	0x9D00_0180	0		
.vectors	0x9D00_0200	0		
.dinit	0x9D00_0248	0		
			b init	0x9D00_0254
			d init	0x9D00_026C
			ge init	0x9D00_02E8
.text	0x9D00_0648	0	Begin. of main	0x9D00_0648
			End of main	0x9D00_08A0
.rodata	0x9D00_0D38	0		
			a	0x9D00_0D38
			c	0x9D00_0D48
			la init	0x9D00_0D64
			Lcc	0x9D00_0DA8
			iscf	0x9D00_0DB0
Boot Flash Section				
.reset	0xBFC0_0000	1		
.bev_excpt	0xBFC0_0380	1		

16. The address of strcpy.o is 0x9D0008A4.

17.

- The .reset section is used for reset exceptions. The processor performs a full reset initialization.
- The .bev_excpt section is used for BEV exceptions.
- The .app_excpt section is used for general exceptions.

18.

- The bottom of the stack moves up because the heap grows upwards. Since we changed the heap to 512 bytes, the stack has to move.
- These variables are stored in Program Flash, so since we added more code in main the addresses are all changed.