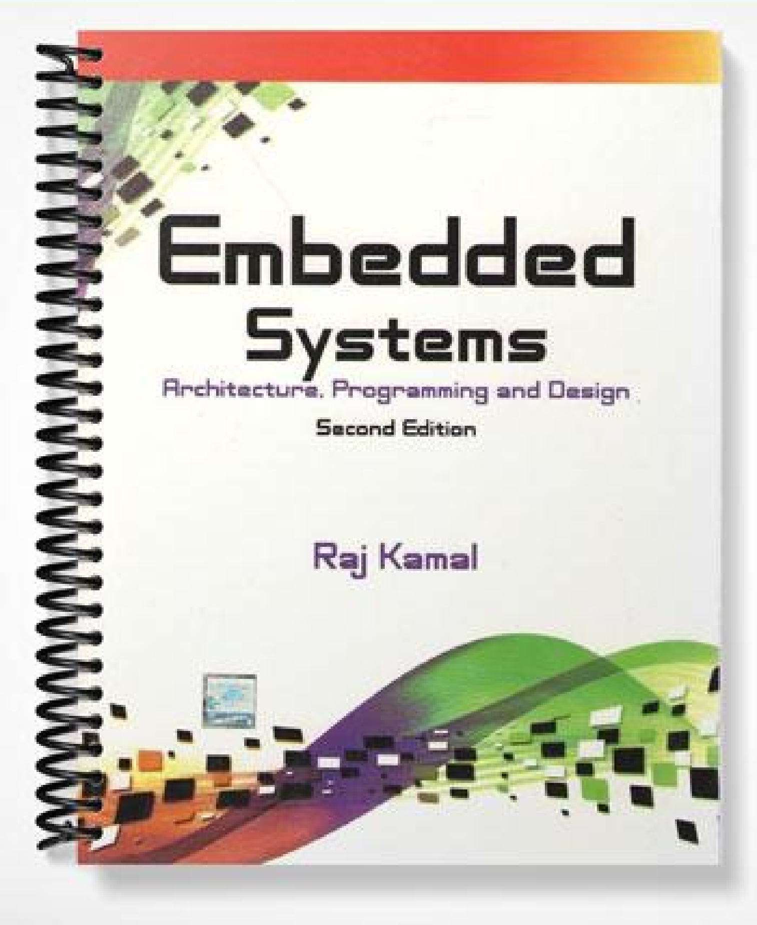
SOLUTIONS MANUAL



Embedded Systems

Architecture, Programming and Design

Second Edition

Raj Kamal

SOLUTIONS TO THE EXERCISES –

Chapter 2

17. *A two by three matix multiplies by another three by two matrix. If data tranfer from a register to another takes 2 ns, addition takes 20 ns, multiplication takes 50 ns, what will be the execution time. How will a MAC unit help, assume that these times are same in a DSP with a MAC unit?* .

When a two by three matix A multiplies by another three by two matrix B, a new three by three matrix with total 9 new elements generates. Assume there are only four registers, R1 and R2 for the 8-bit operands and R3 and R4 for saving the results of multiplication of R1 and R2 or result of addition of 16-bit R1-R2 with 16 bit. R3-R4.

Without a MAC unit

Before the multiplications, two elements A11 and B11 transfer to the registers R1 and R2. This takes 2 . 2 ns = 4 ns. Mulrplication will take another 50ns and let the result is in R4 and R3. To save these, it will take another 4 ns. Total time taken is 58 ns.

Now R2 are again given another element B21. This takes 2 ns. Now multiplication takes another 50 ns and transfer another 4ns. This operation will take another 56 ns.

Now R2 is again given another element B31, The multiplication and save takes totla 56 ns.

Now the results of above three operations are to be added. Total number of register to register additions will be 4. Time taken will be 2 * (4 ns + 4 ns + 20 ns + 4 ns) = 64 ns.

A new matrix C element C11 takes (58 + 56 + 56) ns + 64 ns = 234 ns.

For nine elements of C, the matix multiplitication will take 2106 ns.

With a MAC unit like in a DSP

A MAC does a single cycle MAC. Hence, it take 2 ns for a MAC operation. For finding each element of C, it will take 4 ns + 2 ns. Total time will be only 9 * (6 ns) = 54 ns.

Therefore a MAC unit is essential for certain operations.

18. *An array has 10 integers, each of 32-bits. Let an integer be equal to its index in the array multiplied by 1024. Let the base- address in memory be 0x4800. How will the bits be stored for the 0th, 4th and 9th element in (a) big-endian mode (b) little endian mode.*

Ten integer and their saving at the 32-bit memory as big endian and little endian

Index	Integer	Big endian	Little endian
0	0	0000 0000 0000 0000 0000 0000 0000 0000	0000 0000 0000 0000 0000 0000 0000 0000
1	1024	0000 0000 0000 0100 0000 0000 0000 0000	0000 0000 0000 0000 0000 0100 0000 0000
2	2048	0000 0000 0000 1000 0000 0000 0000 0000	0000 0000 0000 0000 0000 1000 0000 0000
3	3072	0000 0000 0000 1100 0000 0000 0000 0000	0000 0000 0000 0000 0000 1100 0000 0000
4	4096	0000 0000 0001 0000 0000 0000 0000 0000	0000 0000 0000 0000 0001 0000 0000 0000
5	5120	0000 0000 0001 0100 0000 0000 0000 0000	0000 0000 0000 0000 0001 0100 0000 0000
6	6144	0000 0000 0001 1000 0000 0000 0000 0000	0000 0000 0000 0000 0001 1000 0000 0000
7	7165	0000 0000 0001 1100 0000 0000 0000 0000	0000 0000 00000000 0001 1100 0000 0000
8	8192	0000 0000 0010 0000 0000 0000 0000 0000	0000 0000 0000 0000 0010 0000 0000 0000
9	9216	0000 0000 0010 0100 0000 0000 0000 0000	0000 0000 0000 0000 0010 0100 0000 0000

Note that first 8 bits in little endian becomes last eight bits in big endian. Next 8 bits in little endian becomes last but one 8 bits.

19. *We can assume that the memory of an embedded system is also a device. List the reasons for it. [Hint: Use of pointers like access control registers and the concept of virtual file and RAM disk devices.]*

- Memory address is like a port address in the memory mapped IO architecture in the processor. We can therefore assume a file on a disc equivalent to a file in memory. It is called RAM disk. We can operate same functions (open, close, write) and same statuses (read only, write only, read and write) on RAM disk as to a file.

- Just like a byte stream to a print-buffer, we can create a pipe device in the RAM and operate the functions (open, close, write) and same statuses (read only, write only, read and write) on to a pipe.
- Pointer address for a byte in the RAM behaves like a port address in case of a virtual device.
- Use of pipes and file devices in embedded systems is a useful concept.

20. *What are the advantages of parameterised block RAM and parameterised distributed RAM in a fast transceiver?*

- Parameterised block RAM is used when a specific block of the RAM is dedicated for use by a subunit only, for example, a MAC unit. A parameterised block RAM is used when an access by the bus is slow compared to processing speed of a subunit.

- The RAM distributes in a Parameterised distributed RAM into the various system subunits. IO buffers and transceiver subunits can have a slice of RAM each and system stack can be at another slice. Distribution provides buffering of memory at the subunits before they are fetched and processed by the processor. It facilitates faster inputs at the I/O devices than the processor system buses access the memory devices.

21. *Nowadays high-performance embedded-systems use either a RISC processor or processor with RISC core with code optimised for the CISC instruction set. Why?*

RISC architecture gives higher performance than a CISC. This is because (i) there are *very few addressing modes* and therefore there is a smaller instruction set. There are just the Store, Push and Load and Pop instructions for storing in, and loading from, the memory. (ii) Most instructions and the instructions in ALU are implemented by operands from the stack or registers only. (iii) Stack also is at a register-set present within the processor. When the stack is also within the microprocessor like the registers, the accesses are fast. (iv) There may be register-windows, each with a set of registers. A window can store the values of the variables and stack in the different subroutines. This results in a fast switching between the two processes (tasks). [Sections 4.6 and 8.1] (v) Instructions execute in the hardwired circuits, which are separate for each given instruction and fetched from the instruction register and (vi) There is single cycle execution for each instruction.

Due to high-performance, nowadays high-performance embedded-systems use a RISC processor or processor with RISC core with code optimised CISC instruction set. A ‘C’ program or high level program converts to CISC instructions and these converts after suitable optimisation into the RISC instructions and then run at the RISC core. Compilation is also within a subunit associated with the processor.

22. *A circular queue has 100 characters at the memory addresses, each of 32-bits. What will be the total memory space required, including the space for both the queue pointers?*

Refer Figure 2.4(c). A character needs just one bit, while at the each memory address there are 32 bits. Therefore, 100/4 = 25 memory addresses shall be needed for the 100 characters. Therefore the four pointers: start, end, front and back are needed for any circular buffer. Therefore, total 29 memory addresses are a needed in a circular buffer of 100 characters.

23. *Estimate the memory requirement for a 500-images capacity digital camera when the resolution is (a) 1024 x 768 pixels, (b) 640 x 480, (c) 320 x 240 and (d) 160 x 120 pixels and each image stores in compressed jpg format. Assuming that there is 24 bit per pixel.*

Assume average compression is by a factor of 1:16.

1024 x 768 pixels

24 * (1024 * 768 / 16) bit/8 /1024= 144 kB per image. Total 72000 kB [1kB = 1024 Byte]

640 x 480 pixels

24 * (640 * 480 / 16) bit/8/1024 = 56.25 kB per image. Total 28125 kB

320 x 240

24 * (320 * 240 / 16) bit/8/1024 = 14.0625 kB per image. Total 7031.25 kB

160 x 120 pixels

24 * (160 * 120 / 16) bit/8/1024 = 3.5 kB per image. Total 1758 kB

24. What are the special structural units in processors for the systems for digital camera, real-time video processing, speech compression and video game?

Refer to Appendices A to E for the structural units at the various processor and select. A typical selection is as under.,

The special structural units in processors for the exemplary systems

Memory Type	***Special Set of the Structure units***
Digital camera	VLIW CPU with instruction and data caches; MAC units
Deal-time video processing	Video input unit, Video input unit, audio input unit, audio output unit, Timers, I^2C interface, synchronous serial interface, VLD coprocessor, DVDO, VLIW CPU with instruction and data caches, dedicated image coprocessor and PCI-X IO Interface; Built-in optimised code generating

	compiler unit; Broadband infrastructure and image processing VLIWs; Video, audio, graphics, and communications data streams processing [Appendix E Table E.2.1]
Speech compression	Instruction Cache, Data Cache, MAC units, multistage pipeline processing, multi-line superscalar processing for obtaining processing speed higher than one instruction per clock cycle; Two multipliers and six arithmetical units, highly orthogonal, compiler and assembly optimiser, execution resources. [Table D.1.1]
Video game	Instruction Cache, Data Cache, MAC units, Image processing VLIWs; Image co-processor for real time decoding 640 x 480 VGA MPEG2/ MPEG4 video and real time encoding 352 x 288 CIF for MPEG1, MPEG 2 and MPEG4 [Table D.4.4.]

Special Set means the following units plus units given above.

SRS (*System Register Set); PFCU (Prefetch Control Unit); I-Cache (Instruction Cache); BT Cache (Branch Target Cache); D-Cache (Data Cache); MMU Memory-Management Unit; FLPU (Floating Point Processing Unit); FRS (Floating Point Register Set); Advanced Processing Units; AOU (Atomic Operation Unit)*

25. *How does a decoder help in memory and I/O devices interfacing? Draw four exemplary circuits.*

Hints are given below:

Consider four exempalry circuits with the memory addresses as in Figure 2.9(a), (b), (c) and (d). Assume IO port devices at 0x2F8-0x2FF and 0x2F8-0x2FF

Decoder for Map of Figure 2.9(a)

(i) Connect address bits between 111 1111 1111 and 000 0000 0000 to ROM. Hence A0 to A10 connects the ROM. RD connects to ROM. For decoding the bits left are A11 to A17.

(ii) Connect address bits between 1111 1111 and 000 0000 0000 to RAM. Hence A0 to A7connects the RAM. RD and WR connect to RAM. For decoding the bits left are A8 to A17.

(iii) Connect address bits between 1 1111 1111 and 0 0000 0000 to EEPROM. Hence A0 to A10 connects the RAM. RD and WR connect to RAM. For decoding the bits left are A9 to A17.

Design a decoder that (i) selects ROM when A17 to A12 = 00 0000 and A11 = 0, (ii) selects RAM when A17 = 0; A16 =1 and A15 to A12 = 0000 and A11 to A8 = 0, (iii) selects EEPROM when A17 =1 and A16 = 0 and A15 to A9 = 0.

Repeat the similar exercise for the port address 0x2F8-2FF and 0x3F8-3FF. The ports connect IOWR and IORD control signals.

Repeat the same exercise for the other three decoder-designs and then draw the four Circuits.