<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>7(a)</td><td>One mark for working, (all three columns P, Q and R)
One mark for each correct column Y, Z

<table>
<tr><th>A</th><th>B</th><th>C</th><th>P</th><th>Q</th><th>R</th><th>Y</th><th>Z</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
</table></td><td>3</td></tr>
<tr><td>7(b)</td><td>Full adder</td><td>1</td></tr>
<tr><td>7(c)</td><td>One mark for each point

$Y = \overline{A}\ \overline{B}\ C + \overline{A}\ B\ \overline{C} + A\ \overline{B}\ \overline{C} + A\ B\ C$
Purpose: Sum bit

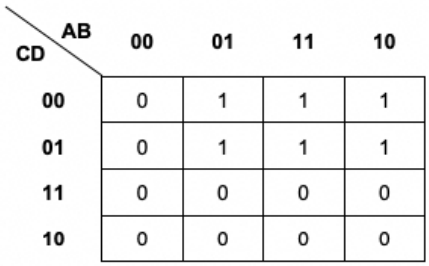
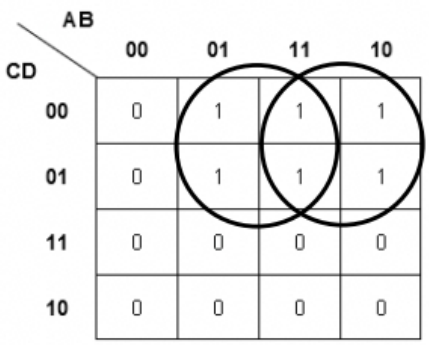
$Z = \overline{A}\ B\ C + A\ \overline{B}\ C + A\ B\ \overline{C} + A\ B\ C$
Purpose: Carry output</td><td>4</td></tr>
</table>

Question	Answer	Marks
7(a)	**One** mark per two correct products **(Max 3)** (**Z** =) **A** $\overline{\textbf{B}}$ $\overline{\textbf{C}}$ **D** + **A** $\overline{\textbf{B}}$ **C D** + **A B** $\overline{\textbf{C}}$ $\overline{\textbf{D}}$ + **A B** $\overline{\textbf{C}}$ **D** + **A B C** $\overline{\textbf{D}}$ + **A B C D**	**3**
7(b)(i)	**One** mark for every two correct rows or columns **(Max 2)** **AB** **CD** **00** 0 0 1 0 **01** 0 0 1 1 **11** 0 0 1 1 **10** 0 0 1 0	**2**
7(b)(ii)	**One** mark for correct loop **(Max 2)** **AB** **CD** **00** 0 0 1 0 **01** 0 0 1 1 **11** 0 0 1 1 **01** 0 0 1 0	**2**
7(b)(iii)	**One** mark per correct marking point **(Max 2)** • **A B // A D** • **+ A D // + A B** (**Z** =) **A B + A D // A D + A B**	**2**
7(b)(iv)	(**Z** =) **A (B + D) // A (D + B)**	**1**

Question	Answer	Marks
4(a)	RISC **max 2** any **two** from: • Uses simple instructions • Uses fixed length instructions • Instructions only require one clock cycle • Uses many registers • Makes use of pipelining • Hardwired CU CISC **max 2** any **two** from: • Uses many **instruction formats** • Uses variable length instructions • Makes use of different addressing modes • Uses few registers • Has a large instruction set • Requires **complex circuits** • Frequently uses cache • Instructions (converted to sub-instructions that) may require many clock cycles • Programmable CU	**4**
4(b)	**One mark** for each difference **max 2** from: • RISC has fewer instructions // CISC has more instructions • RISC has many registers // CISC has few registers • RISC's instructions are simpler // CISC's instructions are more complex • RISC has a few instruction formats / CISC has many instruction formats • RISC usually uses single-cycle instructions// CISC uses multi-cycle instructions • RISC uses fixed-length instructions // CISC uses variable-length instructions • RISC has better pipelineability // CISC has poorer pipelineability • RISC requires less complex circuits// CISC requires more complex circuits • RISC has fewer addressing modes // CISC has more addressing modes • RISC makes more use of RAM// CISC makes more use of cache/less use of RAM • RISC has a hard-wired control unit // CISC has a programmable control unit • RISC only uses load and store instructions to address memory // CISC has many types of instructions to address memory	**2**

Question	Answer	Marks
6	**Two** marks each benefit description **max 4** New system can be tried on different virtual hardware (1) without need to purchase the hardware (1) Easier to recover if software emulating the new computer causes system crash (1) as VM provides protection to other software (1) Emulate programs for the new computer system that are not compatible with the host computer / operating system (1) by using the guest operating system on the old computer (1) More than one new computer system can be emulated (1) this allows multiple operating systems to coexist on a single computer(1) **Two** marks each limitation description **max 2** from: Virtual machines may not be able to emulate the new hardware (1) because this hardware may have been developed since the virtual machine was developed (1) Using virtual machine means execution of extra code // A virtual machine might not be as efficient // resources e.g. memory or processor time are shared (1) processing time increased // performance degrades (1) Use of a virtual machine increases the maintenance overheads (1) because both host system and the virtual machine must be maintained (1)	**6**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>9(a)</td><td><code>LDM #500:</code> Immediate 500
<code>LDD 500:</code> Direct 100
<code>LDI 500:</code> Indirect 20</td><td>3</td></tr>
<tr><td>9(b)</td><td>
<table>
<tr><th rowspan="2">Label</th><th colspan="2">Instruction</th></tr>
<tr><th>Opcode</th><th>Operand</th></tr>
<tr><td></td><td>LDM</td><td>#20</td></tr>
<tr><td></td><td>STO</td><td>Twenty</td></tr>
<tr><td></td><td>LDI</td><td>Y</td></tr>
<tr><td></td><td>ADD</td><td>Twenty</td></tr>
<tr><td></td><td>STO</td><td>Z</td></tr>
<tr><td>Twenty:</td><td>#20</td><td></td></tr>
<tr><td>Y:</td><td></td><td></td></tr>
<tr><td>Z:</td><td></td><td></td></tr>
</table>
One mark for <code>LDM #20</code> seen
One mark for storing <code>20</code> at any address
One mark for labelling that address e.g. <code>Twenty</code> away from the program code
One mark for labelling addresses away from the program code as <code>Y</code> and <code>Z</code>
One mark for correct use of <code>LDI Y</code>
One mark for correct use of <code>STO Z</code>
One mark for correct use of <code>ADD</code> with labelled address
</td><td>7</td></tr>
</table>

Question	Answer	Marks
7(a)	**Two** marks if no errors present **One** mark if one error present 	**2**
7(b)	**One** mark for correct loop **(Max 2)** 	**2**

CD \ AB	00	01	11	10
00	0	1	1	1
01	0	1	1	1
11	0	0	0	0
10	0	0	0	0

Question	Answer	Marks
7(c)	**One** mark for each point • Any correct Boolean term • Boolean terms and operator correct and no other terms present (Z =) $B\overline{C} + A\overline{C}$ **One** mark for simplest form (Z =) $\overline{C}\ (A + B)$	**3**

Question	Answer	Marks
8(a)	**One** mark for each correct point (**Max 3**) • Disk / secondary storage is used to extend the RAM / memory available • … so the CPU appears to be able to access more memory space than the available RAM • Only the data in use needs to be in main memory so data can be swapped between RAM and virtual memory as necessary • Virtual memory is created temporarily.	**3**
8(b)	**One** mark for a correct statement about the difference between paging and segmentation e.g. • Paging allows the memory to be divided into fixed size blocks and Segmentation divides the memory into variable sized blocks. • The operating system divides the memory into pages, the compiler is responsible for calculating the segment size. • Access times for paging is faster than for segmentation.	**1**

Question	Answer	Marks
10(a)	**Two** marks for all five rows correct **One** mark for four rows correct	**2**

Statement	RISC	CISC
uses a smaller instruction set	✓	
uses single-cycle instructions and limited addressing modes	✓	
uses fewer general-purpose registers		✓
uses both hardwired and micro coded control unit		✓
uses a system where cache is split between data and instructions	✓	

Question	Answer	Marks
10(b)	**One** mark for each correct point (**Max 4**) • Instructions are divided into subtasks / 5 stages • … Instruction fetch / IF, Instruction decode / ID, operand fetch / OF, opcode/instruction execute IE, result store / write back result / WB • Each subtask is completed during one clock cycle • No two instructions can execute their same stage at the same clock cycle • The second instruction begins in the second clock cycle, while the first instruction has moved on to its second subtask. • The third instruction begins in the third clock cycle while the first and second instructions move on to their second and third subtasks, respectively, etc.	**4**