Monitoring And Control System

Question 1

7 A computer at a remote weather station is performing three tasks:

- measuring and recording the temperature every 10 seconds
- measuring and recording the wind speed every 10 seconds
- sending the previous day's temperature and wind speed readings to a scientist at another location via the Internet.

The operating system is managing the multitasking of these tasks.

(a) At one point in time:

- the temperature measuring and recording task is idle
- the wind speed is being recorded
- the task to send the previous day's temperature and wind speed readings is waiting for an internet connection.

Identify the process state for each task. Give a reason why each task is in that process state.

Temperature measuring and recording process state .....

Reason .....

.....

.....

Wind speed measuring and recording process state .....

Reason .....

.....

.....

Sending process state .....

Reason .....

.....

.....

[6]

(b) The weather station computer uses an operating system.

Explain how this operating system uses interrupts to schedule the measuring and recording tasks.

.....

.....

.....

.....

.....

.....

.....

..... [4]

Question 2

9 A train cannot move if any of the eight automatic train doors are open. The train door monitoring system, set out below, checks that all the doors are closed before the train can move.

- If a monitoring system detects that a door is open, it sets a specific bit in address `500` to `1`.
- If the bit for door one is equal to `1`, the binary value for hexadecimal `FF` is sent to address `501`. The contents of address `501` are changed to make door 1's light flash when the door is open.
- If the bit for door two is equal to `1`, the binary value for hexadecimal `FF` is sent to address `502`. The contents of address `502` are changed to make door 2's light flash when the door is open.

  This is repeated for each door from 3 to 8.
- Each door sets its bit in address `500` to zero when the door closes, and the contents of the corresponding door address are set to zero.
- The train manager can identify which door is open from the flashing light.

The current contents of address `500` are:

	Door number							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Address `500`	1	0	0	1	0	0	1	0

(a) Complete the following table by writing the values stored in addresses `503` to `508`. Use the contents of address `500` shown above. Note that addresses `501` and `502` are complete.

501	1	1	1	1	1	1	1	1	Door 1
502	0	0	0	0	0	0	0	0	Door 2
503									Door 3
504									Door 4
505									Door 5
506									Door 6
507									Door 7
508									Door 8

[2]

(b) The following table shows assembly language instructions for the processor controlling the train door monitoring system that has one general purpose register, the Accumulator (ACC).

	Instruction		Explanation
Label	**Op code**	**Operand**	
	LDM	&n	Load the hexadecimal number n to ACC
	LDD	<address>	Load the contents of the location at the given address to ACC
	STO	<address>	Store the contents of ACC at the given address
	AND	&n	Bitwise AND the contents of ACC with the hexadecimal number n
	CMP	&n	Compare the contents of ACC with the hexadecimal number n
	JPE	<address>	Following a compare instruction, jump to <address> or <label> if the compare was True
<label>:	<op code>	<operand>	Labels an instruction
	WAIT		Macro to wait one second before the next instruction is executed

After rechecking the doors, address `500` now contains `10101010`.

(i) Complete the table by writing the values of the Accumulator (ACC) and the contents of address `501` as these instructions are executed **once** to check door **1**.

	Instruction		ACC	501
Label	**Op code**	**Operand**		
CHECK1:	LDD	500		
	AND	&80		
	CMP	&00		
	JPE	DOOR1		
	LDM	&FF		
DOOR1:	STO	501		
	WAIT			
	LDM	&00		
	STO	501		
	WAIT			
	JMP	CHECK1		

[4]

(ii) Write the assembly language instructions to check door **2**.

	Instruction	
Label	**Op code**	**Operand**

[4]

(c) Explain how the check door routines show a flashing light or no light.

.....

.....

.....

..... [2]

Question 3

7 A company has a number of lorries that deliver items around the country. The items in each lorry are its load. Each lorry has a monitoring system that provides information to the driver about the state of the load and other data from each trip.

- Data is stored in three memory locations with addresses `801` to `803`.
- Location `801` contains the distance travelled in kilometres for the current trip, stored as a binary integer.
- Location `802` contains the quantity of fuel used in litres for the current trip, stored as a fixed-point binary number with six places before the binary point and two places after the binary point.
- The four most significant bits of location `803` are flags used to identify problems with the load, for example it is too heavy. A flag is set to 1 if there is a problem, or 0 if not.
  The problems are:
  - Bit 7 – load too heavy
  - Bit 6 – load too high
  - Bit 5 – load unstable
  - Bit 4 – load not secured (risk of the load falling off)
  - Bits 0 to 3 are not used

(a) The current contents of addresses `801` to `803` are:

	Most significant bit							Least significant bit
801	0	1	1	0	1	1	0	0
802	0	0	1	0	1	0	0	1
803	0	0	1	0	0	0	0	0

State the information that the current contents of addresses `801` to `803` will provide to the driver.

.....

.....

.....

..... [3]

(b) A lorry has a load that is too heavy and is not secured. It has travelled 120 kilometres and used 35.25 litres of fuel.

Complete the contents of the addresses to record this information.

801								
802								
803								

[3]

(c) The following table shows the instructions for the lorry load monitoring system in assembly language. There is one general purpose register, the Accumulator (ACC).

Table 7.1

Instruction			Explanation
Label	**Op code**	**Operand**	
	LDM	#n	Load the number n to ACC
	LDD	<address>	Load the contents of the location at the given address to ACC
	STO	<address>	Store the contents of ACC at the given address
	AND	#n	Bitwise AND operation of the contents of ACC with the operand
	CMP	#n	Compare the contents of ACC with number n
	JPE	<address>	Following a compare instruction, jump to <address> or <label> if the compare was True
	JMP	<address>	Jump to the given address or label
<label>:	<op code>	<operand>	Labels an instruction

Note:

# denotes immediate addressing
B denotes a binary number, for example B01001010
& denotes a hexadecimal number, for example &4A

(i) Write **assembly language** instructions to set the contents of addresses `801` and `802` to zero, and set all four most significant bits of the contents of address `803` to one.
Use the instruction set from **Table 7.1**.

.....

.....

.....

.....

.....

..... [3]

(ii) A program written in assembly language, continuously checks the flags. If a flag is set, the program jumps to the error-handling routine at the specified label. For example, if the load is too heavy, the program jumps to the error-handling routine with the label `TOOHEAVY`. The error-handling routine instructions have not been provided.

A programmer has written most of the instructions for the program in the following table. There are four missing operands.

Complete the assembly language program by writing the **four** missing operands.

Label	Op code	Operand
CHECKLOAD:	LDD	803
	AND	&F0
	STO	TEMP
	AND	&80
	CMP	&80
	JPE	TOOHEAVY
	LDD	TEMP
	AND	&40
	CMP	
	JPE	TOOHIGH
	LDD	TEMP
	AND	
	CMP	&20
	JPE	UNSTABLE
	LDD	
	AND	&10
	CMP	&10
	JPE	NOTSECURED
	JMP	
TEMP:		

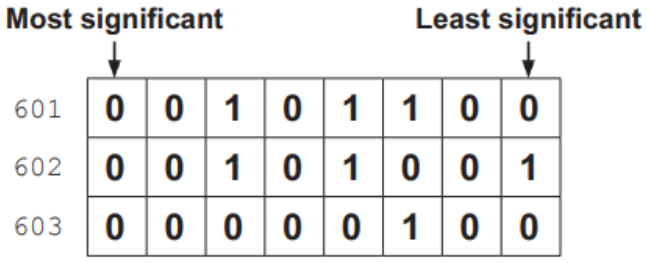
[4]

Question 4

8 A car monitoring system provides information to the driver about the car's performance and alerts the driver to possible problems.

- Data about the car's performance is stored in three memory locations with addresses `601` to `603`.
- Location `601` contains the distance travelled in kilometres for the current trip as a binary integer.
- Location `602` contains the quantity of fuel used in litres for the current trip, as a fixed-point binary number with 5 places before the binary point and three places after the binary point.
- The four least significant bits of location `603` are flags used to identify problems with the car, for example, the fuel is low. A flag is set to 1 if there is a problem, or 0 if not. These problems are:
    - Bit 0 - high engine temperature
    - Bit 1 - low oil pressure
    - Bit 2 - low battery
    - Bit 3 - low fuel
    - Bits 4 to 7 are not used

(a) The current contents of addresses `601` to `603` are:



State the information that the current contents of addresses `601` to `603` will provide to the driver.

.....

.....

.....

..... [3]

(b) A car has low oil pressure and low fuel. It has travelled 80 kilometres and used 7.25 litres of fuel.

Complete the contents of the addresses to record this information.

601								
602								
603								

[3]

(c) The following table shows the assembly language instructions for the car performance monitoring system. There is one general purpose register, the Accumulator (ACC).

Table 8.1

Instruction			Explanation
Label	**Op code**	**Operand**	
	LDM	#n	Load the number n to ACC
	LDD	<address>	Load the contents of the location at the given address to ACC
	STO	<address>	Store the contents of ACC at the given address
	AND	#n	Bitwise AND operation of the contents of ACC with the numeric operand
	CMP	#n	Compare the contents of ACC with the number n
	JPE	<address>	Following a compare instruction, jump to <address> or <label> if the compare was True
	JMP	<address>	Jump to <address> or <label>
<label>:	<op code>	<operand>	Labels an instruction

Note:

# denotes immediate addressing
B denotes a binary number, for example B01001010
& denotes a hexadecimal number, for example &4A

(i) Write **assembly language** instructions to set the contents of addresses `601` and `602` to zero, and set all four least significant bits of the contents of address `603` to one.
Use the instruction set from **Table 8.1**.

.....

.....

.....

.....

.....

..... [3]

(ii) A program continuously checks the flags. If a flag is set, the program moves to the error-handling routine at the specified label. For example, if the engine temperature is high, the program jumps to the label for the error-handling routine `HIGHTEMP`. The error-handling routine instructions have not been provided.

A programmer has written most of the instructions for the program in the following table. There are four missing operands.

Complete the assembly language program by writing the **four** missing operands.

Label	Op code	Operand
CHECKFLAGS:	LDD	603
	AND	&0F
	STO	TEMP
	AND	&01
	CMP	&01
	JPE	HIGHTEMP
	LDD	TEMP
	AND	&02
	CMP	
	JPE	LOWOIL
	LDD	TEMP
	AND	
	CMP	&04
	JPE	LOWBATT
	LDD	
	AND	&08
	CMP	&08
	JPE	LOWFUEL
	JMP	
TEMP:		

[4]

Question 5

6 Monitoring and control systems have many different applications.

(a) Explain the importance of feedback in a control system.

.....

.....

.....

.....

.....

..... [3]

(b) An indoor swimming pool is to be kept at a constant temperature of 28 degrees.

Describe the use of feedback in this control system.

.....

.....

.....

.....

.....

.....

.....

..... [4]

(c) Give **one** example of a monitoring system. Explain why this is a monitoring system.

Monitoring system .....

.....

Explanation .....

.....

.....

.....

[3]

Question 6

6 A company sells plant watering systems that automatically turn on water sprinklers when the soil becomes too dry.

The plant watering system has a processor and connecting cables.

Identify **two** other hardware devices that are required in this system. State the purpose of each device.

Device 1 .....

Purpose .....

.....

Device 2 .....

Purpose .....

.....

[4]

Question 7

5 A weather station uses monitoring and control systems.

(a) Describe the difference between a monitoring system and a control system.

.....

.....

.....

..... [2]

Question 8

7 A museum stores antique items that need to be kept at constant temperature.

The museum is not sure about the actual temperatures. The museum installs some equipment. This records the temperatures every hour and ensures the temperature stays within a set range.

(a) Identify the type of system described.

.....[1]

(b) The system has a temperature sensor.

Identify **two** other items of hardware that the museum can use for the type of system identified.

Describe the purpose of each item.

Item 1 .....

Purpose .....

.....

Item 2 .....

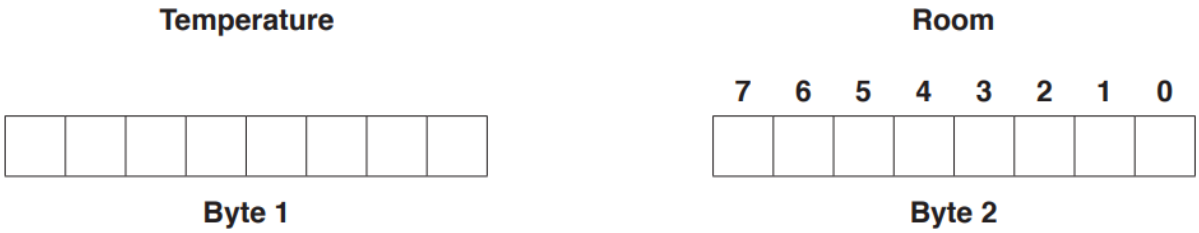
Purpose .....

.....

[4]

(c) The equipment records the temperature in all seven rooms in the museum.

Each recording is stored as two successive bytes in memory. The format is as shown.



The room is indicated by the setting of one of the bits in **Byte 2** to 1. For example, room 7 is indicated by setting bit 7 to 1.

Bit 0 of **Byte 2** is a flag:

- The flag's initial value is zero.
- When the reading has been processed, the flag's value is set to 1.

Byte 1 contains the temperature reading as an unsigned integer.

One reading returns the following binary data.

(i) Analyse the data contained in the two bytes.

.....

.....

.....

.....

.....[3]

(ii) The system receives a temperature reading of 238 from room number 4.

Complete the bytes to show the two bytes for this recording. The reading has not yet been processed.

[2]

Question 9

6 **(a)** There are five scenarios on the left and two types of system on the right.

Draw a line to link each scenario to its correct type of system.

Scenario	System
Car speed display	
Aeroplane autopilot	Control
Rollercoaster	Monitoring
Recording the rainfall at a weather station	
Robot loading a part onto a conveyor belt	

[2]

(b) Mary has six fish tanks. The temperature of the water in each tank needs to be within a specific range.

Identify **three** items of hardware that Mary can add to her tanks to help maintain the temperature. Describe the purpose of each item.

Item 1 .....

Purpose .....

.....

Item 2 .....

Purpose .....

.....

Item 3 .....

Purpose .....

.....

[6]

(c) A temperature reading is taken from each tank once per minute. The temperature reading is stored as two successive bytes. The format is shown:

Fish tank

7	6	5	4	3	2	1	0

Byte 1

Temperature reading

Byte 2

The fish tank number is indicated by setting one of the bits in **Byte 1** to 1. For example, fish tank number 5 is indicated by setting bit 5 to 1.

Bit 7 of **Byte 1** is a flag:

- the flag's initial value is zero
- when the reading has been processed, the flag's value is set to 1

Bit 0 of **Byte 1** is unused.

Byte 2 contains the temperature reading as a two's complement integer.

(i) After a temperature reading has been taken, the bytes contain the following data.

7	6	5	4	3	2	1	0
0	0	0	0	1	0	0	0

Byte 1

0	0	0	0	1	0	1	1

Byte 2

Analyse the data contained in the two bytes.

.....

.....

.....

.....

.....

.....[3]

(ii) The system receives a temperature reading of –2 from fish tank number 4.

Complete the bytes to show the values for this reading after it has been processed.

7	6	5	4	3	2	1	0

Byte 1

Byte 2

[2]

(d) A hardware device to affect the temperature of each tank is on or off depending on the value of a bit in memory location `6753`.

If bit 4 is 1, then the hardware device in fish tank 4 is on.

Write **assembly language** instructions to set bit 4 of memory location `6753` to 1 without changing any other bits. Use the instruction set provided.

.....

.....

.....

.....[3]

Instruction set

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the location at the given address to ACC.
STO	<address>	Store the contents of ACC at the given address.
AND	#n	Bitwise AND operation of the contents of ACC with the operand.
AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>.
XOR	#n	Bitwise XOR operation of the contents of ACC with the operand.
OR	#n	Bitwise OR operation of the contents of ACC with the operand.
OR	<address>	Bitwise OR operation of the contents of ACC with the contents of <address>. <address> can be an absolute address or a symbolic address.

Question 10

6 A computer system is used to manage some of the functions in a vehicle. The vehicle has a number of sensors and actuators. One sensor is used to monitor the moisture on the screen. If the moisture exceeds a pre-set value, the windscreen wiper motor turns on automatically.

The software used in the computer system is dedicated to the sensor management functions. When the system starts, the software runs some initial tasks. It then loops continuously until the system is switched off.

(a) (i) State the name given to the type of system described.

.....[1]

(ii) Explain your answer to **part (i)**.

.....

.....[1]

Question 11

The contents of the 16-bit register are loaded into the 16-bit accumulator:

Accumulator

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0

An instruction is required to achieve the following:

- If bit 9 is zero, set the accumulator to zero.
- If bit 9 is one, set the accumulator to a non-zero value.

Write this instruction using an appropriate bitwise operation.

.....[2]

Question 12

6 A large office building has many floors. On each floor there are security sensors and security cameras. There is the same number of sensors on each floor. The building has a single security room.

The images from the security cameras are output on monitors (one monitor for each floor) placed in the security room.

The data from the sensors are read and processed by a computer system. Sensor readings and warning messages can be displayed on the monitors.

(a) (i) State the name given to the type of system described.

.....[1]

(ii) Explain your answer to **part (i)**.

.....

.....[1]

(iii) State **two** sensors that could be used in this system.

Sensor 1 .....

Sensor 2 .....

[2]

Question 13

6 A large warehouse stores goods that must be kept above a temperature of 15 degrees Celsius. The warehouse has six temperature sensors which are each placed at a different location in the warehouse.

A computer system is programmed to turn on appropriate heaters when one of the sensors is below the minimum temperature.

(a) (i) State the name given to the type of system described.

..... [1]

(ii) Justify your answer to **part (i)**.

.....

..... [1]

(b) Sensors and heaters are two types of device used in this system.

State **two** other devices that are used. Justify your choice.

Device 1 .....

Justification .....

.....

Device 2 .....

Justification .....

.....

[4]

(c) The computer system stores the temperature readings for the six sensors in six 8-bit memory locations.

Six of the bits in an 8-bit register, `LOWREG`, are used to indicate whether a particular reading is below the minimum temperature. A value of `1` means the reading is below the minimum temperature.

For example:

This pattern of bits in `LOWREG` shows that sensor 5, sensor 4 and sensor 1 have readings below the minimum temperature.

		6	5	4	3	2	1
Not used	Not used	0	1	1	0	0	1

The following table shows part of the instruction set for a processor which has one general purpose register, the Accumulator (ACC), and an Index Register (IX).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the given address to ACC.
LDR	#n	Immediate addressing. Load the number n to IX.
LDX	<address>	Indexed addressing. Form the address from <address> + the contents of the index register. Copy the contents of this calculated address to ACC.
STO	<address>	Store the contents of ACC at the given address.
INC	<register>	Add 1 to the contents of the register (ACC or IX).
ADD	<address>	Add the contents of the given address to the ACC.
OR	<address>	Bitwise OR operation of the contents of ACC with the contents of address.
CMP	#n	Compare the contents of ACC with number n.
CMP	<address>	Compare the contents of ACC with the contents of <address>.
JMP	<address>	Jump to the given address.
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True.
JGE	<address>	Following a compare instruction, jump to <address> if the content of ACC is greater than or equal to the number used in the compare instruction.

Part of the assembly language code for updating LOWREG is:

Label	Op code	Operand
LOWTEMP:		15
LOWREG:		B00000000
COUNTER:		1
START:	LDR	#0
LOOP:	LDX	8000
	CMP	LOWTEMP
	JGE	TEMPOK
	LDD	LOWREG
	OR	COUNTER
	STO	LOWREG
TEMPOK:	LDD	COUNTER
Q1:	CMP	#32
	JPE	HEATON
	ADD	COUNTER
	STO	COUNTER
	INC	IX
	JMP	LOOP
HEATON:	LDD	LOWREG

(i) The code uses six memory locations to store the temperature readings. It stores readings for sensors 1 to 6 at addresses `8000` to `8005`.

At a particular time, the memory locations store the following data.

8000	8001	8002	8003	8004	8005
17	14	15	15	16	14

Dry run the assembly language code starting at `START` and finishing when the loop has been processed twice.

LOWTEMP	LOWREG	COUNTER	ACC	IX
15	B00000000	1		

[4]

(ii) Explain why the operand of the instruction labelled `Q1` has the value 32.

.....

.....

.....

..... [2]

(iii) The code beginning at the instruction labelled `HEATON` must make the system turn on the heaters in those areas that are below the minimum temperature.

Describe what this code will have to do.

.....

.....

.....

.....

.....

..... [3]

Question 14

6 The environment in a very large greenhouse is managed by a computer system. The system uses a number of different sensors that include temperature sensors. In addition, the system controls a number of heaters, windows and sprinklers.

(a) State **one** other type of sensor that could be used with this system.

Justify your choice.

Sensor .....

Justification .....

.....

[2]

(b) Describe why feedback is important in this system.

.....

.....

.....

.....

.....

.....[3]

(c) **(i)** The system makes use of a number of parameters. These parameters are used in the code that runs the system.

State **one** of the parameters used in controlling the temperature in the greenhouse.

.....[1]

(ii) Explain how the parameter identified in **part (c)(i)** is used in the feedback process.

.....

.....

.....

.....[2]

(d) There are eight temperature sensors numbered 1 to 8. Readings from these sensors are stored in four 16-bit memory locations. The memory locations have addresses from `4000` to `4003`. Each memory location stores two sensor readings as two unsigned binary integers.

Sensor 1 reading is stored in bits 8 to 15 of address `4000`; Sensor 2 reading is stored in bits 0 to 7 of address `4000` and so on. The diagram shows that the current sensor 1 reading has a value of 97.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
4000	0	1	1	0	0	0	0	1	0	0	1	1	1	0	0	1
4001	1	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0
4002	0	0	0	1	0	1	0	0	0	0	0	0	1	1	0	1
4003	1	0	0	0	0	0	1	0	1	1	0	0	0	1	0	1

(i) Give the denary value of the current reading for Sensor 5.

.....

.....

.....

.....[1]

(ii) The following table shows part of the instruction set for a processor. The processor has one general purpose register, the Accumulator (ACC).

Instruction		Explanation
Op code	**Operand**	
LDD	<address>	Direct addressing. Load the contents of the location at the given address to ACC.
AND	#n	Bitwise AND operation of the contents of ACC with the operand.
AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>.
XOR	#n	Bitwise XOR operation of the contents of ACC with the operand.
XOR	<address>	Bitwise XOR operation of the contents of ACC with the contents of <address>.
OR	#n	Bitwise OR operation of the contents of ACC with the operand.
OR	<address>	Bitwise OR operation of the contents of ACC with the contents of <address>. <address> can be an absolute address or a symbolic address.
LSL	#n	Bits in ACC are shifted n places to the left. Zeros are introduced on the right hand end.
LSR	#n	Bits in ACC are shifted n places to the right. Zeros are introduced on the left hand end.

The reading for Sensor 5 is used in a calculation. The calculation is carried out by two assembly language instructions.

The first instruction loads the contents of the 16-bit location that contains the value for Sensor 5.

The second instruction moves the bits in Sensor 5 so that the 16-bit value is the value of Sensor 5.

Complete the two instructions in the following code. Use the instruction set provided.

```
LDD ..... // load the contents of the 16-bit location
          containing the value for Sensor 5 into the
          Accumulator

..... // move the bits in the Accumulator so that the
      Accumulator stores the value of Sensor 5 as an
      unsigned 16-bit binary integer
```

[3]

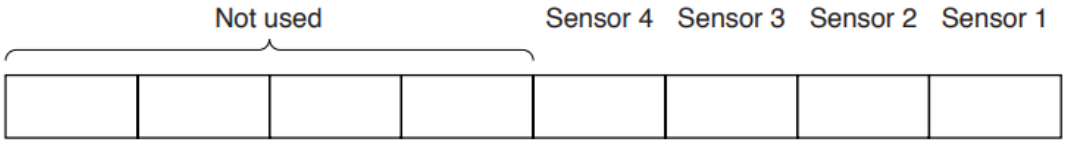
Question 15

6 An intruder detection system for a large house has four sensors. An 8-bit memory location stores the output from each sensor in its own bit position.

The bit value for each sensor shows:

- 1 – the sensor has been triggered
- 0 – the sensor has not been triggered

The bit positions are used as follows:



The output from the intruder detection system is a loud alarm.

(a) (i) State the name of the type of system to which intruder detection systems belong.

.....[1]

(ii) Justify your answer to **part (i)**.

.....

.....[1]

(b) Name **two** sensors that could be used in this intruder detection system. Give a reason for your choice.

Sensor 1 .....

Reason .....

.....

Sensor 2 .....

Reason .....

.....[4]

The intruder system is set up so that the alarm will only sound if two or more sensors have been triggered.
An assembly language program has been written to process the contents of the memory location.

The table shows part of the instruction set for the processor used.

Instruction		Explanation
Op code	Operand	
LDD	<address>	Direct addressing. Load the contents of the given address to ACC
STO	<address>	Store the contents of ACC at the given address
INC	<register>	Add 1 to the contents of the register (ACC or IX)
ADD	<address>	Add the contents of the given address to the contents of ACC
AND	<address>	Bitwise AND operation of the contents of ACC with the contents of <address>
CMP	#n	Compare the contents of ACC with the number n
JMP	<address>	Jump to the given address
JPE	<address>	Following a compare instruction, jump to <address> if the compare was True
JGT	<address>	Following a compare instruction, jump to <address> if the content of ACC is greater than the number used in the compare instruction
END		End the program and return to the operating system

(c) Part of the assembly code is:

	Op code	Operand
SENSORS:		B00001010
COUNT:		0
VALUE:		1
LOOP:	LDD	SENSORS
	AND	VALUE
	CMP	#0
	JPE	ZERO
	LDD	COUNT
	INC	ACC
	STO	COUNT
ZERO:	LDD	VALUE
	CMP	#8
	JPE	EXIT
	ADD	VALUE
	STO	VALUE
	JMP	LOOP
EXIT:	LDD	COUNT
TEST:	CMP	…
	JGT	ALARM

(i) Dry run the assembly language code. Start at `LOOP` and finish when `EXIT` is reached.

BITREG	COUNT	VALUE	ACC
B00001010	0	1	

[4]

(ii) The operand for the instruction labelled `TEST` is missing.

State the missing operand.

.....[1]

(iii) The intruder detection system is improved and now has eight sensors.

One instruction in the assembly language code will need to be amended.

Identify this instruction .....

Write the amended instruction .....[2]

Question 16

5 A gardener grows vegetables in a greenhouse. For the vegetables to grow well, the temperature needs to always be within a particular range.

The gardener is not sure about the actual temperatures in the greenhouse during the growing season. The gardener installs some equipment. This records the temperature every hour during the growing season.

(a) Name the type of system described.

.....[1]

(b) Identify **three** items of hardware that would be needed to acquire and record the temperature data. Justify your choice for each.

Item 1 .....

Justification .....

.....

Item 2 .....

Justification .....

.....

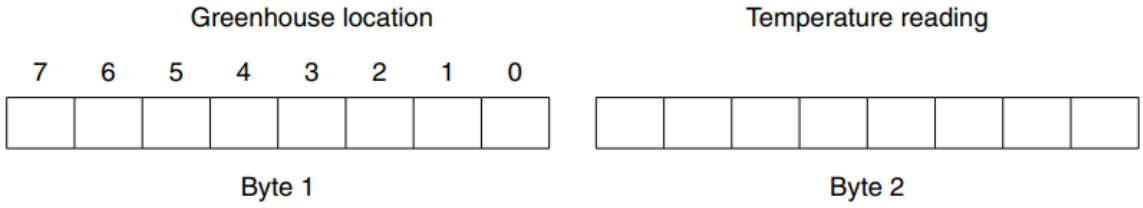
Item 3 .....

Justification .....

.....[6]

(c) The equipment records temperatures in the greenhouse. It does this for seven locations.

Each recording is stored as two successive bytes. The format is shown below:



The location is indicated by the setting of one of the seven bits in byte 1. For example, location 4 is indicated by setting bit 4.

Bit 0 of byte 1 acts as a flag:

- the initial value is zero
- when the reading has been processed it is set to 1

Byte 2 contains the temperature reading (two's complement integer).

(i) Interpret the data in byte 1 shown below:

7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	1

Byte 1

0	0	0	1	1	0	0	0

Byte 2

.....

.....

.....

.....[2]

(ii) The system receives a temperature reading of –5 degrees from sensor 6.

Complete the boxes below to show the two bytes for this recording. The reading has not yet been processed.

7	6	5	4	3	2	1	0

Byte 1

Byte 2

[2]

(d) **(i)** The accumulator is loaded with the value of byte 1 from location 106.

Write the assembly language instruction to check whether the reading in byte 2 came from location 4.

```
LDD 106            // data loaded from address 106
```

.....[4]

(ii) Write the assembly language instruction to set the flag (bit 0) of the byte contained in the accumulator to 1.

.....[2]