3.6.2 Bit manipulation to monitor and control devices

Wednesday, 18 November 2020 3:22 PM

3.6.2 Bit manipulati...

3.6.2 Bit manipulation to monitor and control devices

Computer Science (9608)

May/June 2015.P31/P32

5 (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:

Greenhouse location

7	6	5	4	3	2	1	0

Byte 1

Temperature reading

Byte 2

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]**

3.6.2 Bit manipulation to monitor and control devices

May/June 2015.P33

6 Each greenhouse has eight sensors (numbered 1–8).
• The byte at address 150 is used to store eight 1-bit flags.
• A flag is set to indicate whether its associated sensor reading is waiting to be processed.
• More than one sensor reading may be waiting to be processed at any particular moment.
• Data received from the sensors is stored in a block of eight consecutive bytes (addresses 201–208).
• The data from sensor 1 is at address 201, the data from sensor 2 is at address 202, and so on.

Sensor number

	1	2	3	4	5	6	7	8
150	0	1	0	0	0	1	0	1

201	0	0	0	0	0	0	0	0
202	0	0	0	0	0	1	0	0
203	0	0	0	0	0	0	0	0
204	0	0	0	1	0	0	0	0
205	0	0	0	0	0	0	1	0
206	0	0	0	1	0	1	0	0
207	0	0	0	1	0	0	1	0
208	0	0	0	1	0	0	1	0

(d) (i) Interpret the current reading for sensor 2. **[2]**

(ii) The accumulator is loaded with the data from location 150.
Write the assembly language instruction to check whether there is a value waiting to be processed for sensor 6.

```
LDD 150          // data loaded from address 150
```
[3]

May/June 2016.P31/P32

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:

Not used				Sensor 4	Sensor 3	Sensor 2	Sensor 1
				8	4	2	1

3.6.2 Bit manipulation to monitor and control devices

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

3.6.2 Bit manipulation to monitor and control devices

(c) Part of the assembly code is:

	Op code	Operand
SENSORS:		B00001010
COUNT:		0X2
VALUE:		1248
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	#1
	JGT	ALARM

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

3.6.2 Bit manipulation to monitor and control devices

BITREG	COUNT	VALUE	ACC
B00001010	0	1	

[4]

(ii) The operand for the instruction labelled TEST is missing.
State the missing operand. #1 **[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 ....CMP #8....
Write the amended instruction ....CMP #128.... **[2]**

May/June 2018.P31/P33

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.
(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.

Temperature

Byte 1

Room

7	6	5	4	3	2	1	0

Byte 2

3.6.2 Bit manipulation to monitor and control devices

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.

Temperature

1	0	1	1	0	0	1	1

Byte 1

Room

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

Byte 2

(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.

Byte 1

Byte 2 **[2]**