SQA Advanced Unit Specification

General information for centres

Unit title: MCU/MPU I/O Hardware Control

Unit code: HT1J 48

Unit purpose: This Unit is designed to enable candidates to gain knowledge and understanding and apply a number of input/output techniques to the writing of assembly language programs in an engineering environment. Typically these would be embedded software implemented on a microcontroller or microprocessor based system. Outcome 1 has been written to introduce the principles of two-way handshake control of data transfer. Outcomes 2 and 3 then introduce the control of digital to analogue and analogue to digital converters. Outcome 4 applies interrupt handling techniques to data transfer.

On completion of this Unit candidates should be able to:

1. Implement two-way handshake control of the exchange of a block of data.
2. Generate software to control an application of a digital to analogue converter.
3. Generate software to control an application of an analogue to digital converter.
4. Generate software to control an interrupt driven exchange of a block of data between peripheral devices and the processor.

Credit value: 1 SQA Credit at SCQF level 8: (8 SCQF credit points at SCQF level 8*)

**SCQF credit points are used to allocate credit to qualifications in the Scottish Credit and Qualifications Framework (SCQF). Each qualification in the Framework is allocated a number of SCQF credit points at an SCQF level. There are 12 SCQF levels, ranging from National 1 to Doctorates.*

Recommended prior knowledge and skills: Candidates should have a basic knowledge and understanding of assembly language programming. This may be evidenced by possession of the SQA Advanced Unit, MCU/MPU Assembly Language Programming.

Core skills: There may be opportunities to gather evidence towards Core Skills in this Unit, although there is no automatic certification of Core Skills or Core Skills components.

Context for delivery: This Unit was developed for the SQA Advanced Diploma in Electronics award. If this Unit is used in another group award(s) it is recommended that it should be taught and assessed within the context of the particular group award(s) to which it contributes.

Assessment: It is recommended that this Unit is assessed on an Outcome by Outcome basis, although centres may find it advantageous to combine teaching of this Unit with that of Signal Processing and Conditioning in which case Outcomes 2 and 3 will be combined.

It is recommended that assessment takes the form of laboratory experiments, followed up by written reports. Practical exercises should be undertaken under controlled, supervised conditions. Centres should supply candidates with guidelines on the necessary standard of documentation. Candidates should submit written reports within 14 days of the practical work being completed.

Centres should take every reasonable effort to ensure that reports are the candidates' own work. It may be possible to issue each candidate with a slightly different specification of equal complexity, or alternatively where there is a suspicion of copying or plagiarism, an appropriate response may be to interview candidates. A checklist should be used to record oral evidence of the candidates' understanding.

Centres are recommended to develop appropriate checklists to support the assessment requirements for each of the knowledge and skills items. Centres are also recommended to produce a marking schedule based on the evidence requirements listed indicating clearly the required content of the report. Candidates who do not meet the standard should be obliged to correct and resubmit their work.

SQA Advanced Unit Specification: Statement of standards

Unit title: MCU/MPU I/O Hardware Control

Unit code: HT1J 48

The sections of the Unit stating the Outcomes, knowledge and/or skills, and evidence requirements are mandatory.

Where evidence for Outcomes is assessed on a sample basis, the whole of the content listed in the knowledge and/or skills section must be taught and available for assessment. Candidates should not know in advance the items on which they will be assessed and different items should be sampled on each assessment occasion.

Outcome 1

Implement two-way handshake control of the exchange of a block of data.

Knowledge and/or skills

- Single and double handshaking and related timing diagrams
- Handshaking features of interface IC
- Design of handshaking programs to exchange a block of data

Evidence requirements

All parts of the knowledge and skills listed above shall be assessed. Evidence will be provided in the form of a written report. The candidate's response will be judged to be satisfactory where evidence provided is sufficient to meet the requirements for each item by showing that the candidate is able to:

- design assembly language software to exchange a block of data under handshake control
- draw timing diagrams showing the handshake signals
- implement and test the software
- record results of the exchange of data
- write a report which includes the following:
  - a listing of the program used
  - details of test data
  - evidence of successful transfer
  - timing diagrams of program
  - conclusions

There is no requirement for candidates to write software from scratch during the assessment. They may be permitted to modify programs that they have developed in tutorial exercises, or combine sections of code adapted from other programs.

Assessment guidelines

Centres are recommended to develop appropriate checklists to support the assessment requirements for each of the knowledge and skills items. Centres are also recommended to produce a marking schedule based on the evidence requirements listed above indicating clearly the required content of the report. Candidates who do not meet the standard should be obliged to correct and resubmit their work.

Outcome 2

Generate software to control an application of a digital to analogue converter

Knowledge and/or skills

- Control of a D/A converter in assembly language software
- Control of the output data rate
- Applications of D/A programs

Evidence requirements

All parts of the knowledge and skills listed above shall be assessed. Evidence will be provided in the form of a written report. The candidate's response will be judged to be satisfactory where evidence provided is sufficient to meet the requirements for each item by showing that the candidate is able to:

- develop an assembly language program to output digital data to a D/A converter
- record analogue output of the program for appropriate test data
- write a report which includes the following:
  - a listing of the program used
  - a description of the timing diagram for the D/A converter
  - details of test data and output signal
  - details of the data rate of the program
  - conclusions

There is no requirement for candidates to write software from scratch during the assessment. They may be permitted to modify programs that they have developed in tutorial exercises, or combine sections of code adapted from other programs.

Assessment guidelines

Centres are recommended to develop appropriate checklists to support the assessment requirements for each of the knowledge and skills items. Centres are also recommended to produce a marking schedule based on the evidence requirements listed above indicating clearly the required content of the report. Candidates who do not meet the standard should be obliged to correct and resubmit their work.

Centres are recommended to assess this Outcome at the same time as Outcome 3 by issuing a specification for a program to sample an analogue signal and reconstruct it via a digital to analogue converter. It may then be possible to combine this assessment with that of Signal Processing and Conditioning Outcomes 1 and 2.

Outcome 3

Generate software to control an application of an analogue to digital converter

Knowledge and/or skills

- Control of an A/D converter in assembly language software
- Control of the sampling rate
- Applications of A/D programs

Evidence requirements

All parts of the knowledge and skills listed above shall be assessed. Evidence will be provided in the form of a written report. The candidate's response will be judged to be satisfactory where evidence provided is sufficient to meet the requirements for each item by showing that the candidate is able to:

- develop an assembly language program to read analogue data from an A/D converter
- record digital output of the program for an appropriate test signal input
- write a report which includes the following:
  - a listing of the program used
  - a description of the timing diagram for the A/D converter
  - details of test signals and outputs
  - details of the sampling rate of the program
  - conclusions

Centres should take every reasonable effort to ensure that reports are the candidates' own work. It may be possible to issue each candidate with a slightly different specification, or alternatively where there is a suspicion of copying or plagiarism, an appropriate response may be to interview candidates. A checklist may be used to record oral evidence of the candidates' understanding.

There is no requirement for candidates to write software from scratch during the assessment. They may be permitted to modify programs that they have developed in tutorial exercises, or combine sections of code adapted from other programs.

Assessment guidelines

Centres are recommended to develop appropriate checklists to support the assessment requirements for each of the knowledge and skills items. Centres are also recommended to produce a marking schedule based on the evidence requirements listed above indicating clearly the required content of the report. Candidates who do not meet the standard should be obliged to correct and resubmit their work.

Centres are recommended to assess this Outcome at the same time as Outcome 2 by issuing a specification for a program to sample an analogue signal and reconstruct it via a digital to analogue converter. It may then be possible to combine this assessment with that of Signal Processing and Conditioning Outcomes 1 and 2.

Outcome 4

Generate software to control an interrupt driven exchange of a block of data between peripheral devices and the processor

Knowledge and/or skills

- Purpose of interrupts and applications in hardware control
- Use of assembly language programs to service interrupts
- Background and foreground tasks
- Exchange of blocks of data

Evidence requirements

All parts of the knowledge and skills listed above shall be assessed. Evidence will be provided in the form of a written report. The candidate's response will be judged to be satisfactory where evidence provided is sufficient to meet the requirements for each item by showing that the candidate is able to:

- develop an assembly language interrupt handler to exchange blocks of data
- develop a program to set up the interrupt handler and perform foreground tasks
- test program with suitable data
- record results of tests
- write a report which includes the following:
  - a listing of the program(s) used
  - details of test data
  - evidence of successful transfer
  - conclusions

There is no requirement for candidates to write software from scratch during the assessment. They may be permitted to modify programs that they have developed in tutorial exercises, or combine sections of code adapted from other programs.

Assessment guidelines

Centres are recommended to develop appropriate checklists to support the assessment requirements for each of the knowledge and skills items. Centres are also recommended to produce a marking schedule based on the evidence requirements listed above indicating clearly the required content of the report. Candidates who do not meet the standard should be obliged to correct and resubmit their work.

Administrative information

Unit code: HT1J 48

Unit title: MCU/MPU I/O Hardware Control

Superclass category: CA

Date of publication: August 2017

Version: 01

Source: SQA

SQA acknowledges the valuable contribution that Scotland's colleges have made to the development of SQA Advanced Qualifications.

FURTHER INFORMATION: Call SQA's Customer Contact Centre on 44 (0) 141 500 5030 or 0345 279 1000. Alternatively, complete our Centre Feedback Form.

SQA Advanced Unit Specification: Support notes

Unit title: MCU/MPU I/O Hardware Control

This part of the Unit specification is offered as guidance. The support notes are not mandatory.

While the exact time allocated to this Unit is at the discretion of the centre, the notional design length is 40 hours.

Guidance on the content and context for this Unit

This Unit has been written in order to allow candidates to develop the knowledge, understanding and skills in the following areas:

1) Implement two-way handshake control of the exchange of a block of data
2) Generate software to control an application of a digital to analogue converter
3) Generate software to control an application of an analogue to digital converter
4) Generate software to control an interrupt driven exchange of a block of data between peripheral devices and the processor

This Unit was designed to permit successful candidates to apply assembly language to the control of input/output hardware, and follows on from the Unit, MCU/MPU Assembly Language Programming. It extends the previous Unit to cover a variety of input/output applications, namely handshake control, D/A and A/D converters, and interrupt driven data exchange. Candidates are required to demonstrate the ability to access hardware directly through assembly language programs, rather than being supplied with program modules such as macros or subroutines.

This Unit was developed as one of four SQA Advanced Diploma options and is at SCQF level 8. Access to this Unit should be limited to candidates who have elected to complete the Unit, MCU/MPU Accessibly Language Programming option or who can otherwise demonstrate assembly language skills, for example by industrial experience or by completion of a unit from earlier engineering frameworks.

In designing this Unit the Unit writers have identified the topics they would expect to be covered by lecturers. The writers have also given recommendations as to how much time should be spent on each Outcome. This has been done to help lecturers decide what depth of treatment should be given to the topics attached to each of the Outcomes. While it is not mandatory for a centre to use this list of topics it is strongly recommended that they do so to ensure continuity of teaching and learning across the Engineering Computing Units and because the assessment exemplar pack for this Unit is based on the knowledge and/or skills and list of topics in each of the Outcomes.

The Unit does not include specifications for the type of target system - this is left up to the discretion of centres. It is appreciated that centres will have access to a wide variety of hardware, such as personal computers, microprocessor development boards, microcontrollers, and DSP devices, all of which may be used for the purposes of this Unit. There is also the possibility that technology will change throughout the lifetime of this Unit, so it would be unwise to restrict centres to a particular type of microprocessor based system.

A list of topics is given below. Lecturers are advised to study this list in conjunction with the assessment exemplar pack for this Unit so they can get a clear indication of the standard of achievement expected of candidates taking this Unit.

1. Implement two-way handshake control of the exchange of a block of data (12 hours)

Single and double handshaking
Timing diagrams
Handshaking features of interface IC
Accessing individual bits
Designing handshaking programs
Testing handshaking programs

Areas for experimental work may include:

Construction and test of software
Use of test equipment to record waveforms

2. Generate software to control an application of a digital to analogue converter (8 hours)

Timing diagram for D/A converter
Control of output data rate
Applications of D/A programs

Areas for experimental work may include:

Construction and test of D/A software
Recording analogue output of program
Use of test equipment to analyse D/A signal timing

3. Generate software to control an application of an analogue to digital converter (8 hours)

Timing diagram for A/D converter
Control of input data rate
Applications of A/D programs

Areas for experimental work may include:

Construction and test of A/D software
Recording digital input of program
Use of test equipment to analyse A/D signal timing

4. Generate software to control an interrupt driven exchange of a block of data between peripheral devices and the processor (12 hours)

Purpose of interrupts and applications in hardware control
Background and foreground tasks
Exchange of blocks of data

Areas for experimental work may include:

Construction and test of interrupt handling software
Recording results of program
Use of test equipment to analyse interrupt timing

Guidance on the delivery and assessment of this Unit

It should be noted that this Unit can be delivered on a free standing basis, or alternatively Outcomes 2 and 3 can be combined using signal processing examples, thereby accomplishing parts of the Unit, Signal Processing and Conditioning. This provides course planners with the flexibility to keep Units separate or combine them for teaching and learning and assessment purposes.

Outcome 1 requires candidates to write software to read and write blocks of data, but not at the same time. A typical specification would be to read 16 bytes of parallel data then output them to a printer while controlling the handshake lines directly. Copying the data to the printer while using a 16-byte buffer would be more complex than is intended. Refer to the exemplar for a sample assessment.

It is appreciated that centres will have access to a limited range of test equipment and software such as oscilloscopes, logic analysers, logic probes, and debuggers. The Unit has been written in such a way that the choice of test equipment has been left open for the centre. Test results could be recorded using a storage oscilloscope or logic analyser, or by having the candidate sketch a CRO trace and record signal amplitude and frequency.

Lecturers should be aware that some operating systems intercept calls to input/output devices and interrupt user programs, leading to technical difficulties which may hinder the learning of candidates. In particular, creation of interrupt driven programs may prove problematic. It may be best to write a high level program to intercept an existing interrupt and write the interrupt handler in assembly language.

Open learning

This Unit is more suitable for laboratory delivery however it could be delivered by distance learning provided the candidate has access to a microprocessor development system and external hardware interfaces. This may require some degree of on-line support. However, with regard to assessment, planning would be required by the centre concerned to ensure the sufficiency and authenticity of candidate evidence. Arrangements would be required to be put in place to ensure that the assessment whether done at a single or at multiple events was conducted under controlled, supervised conditions.

To keep administrative arrangements to a minimum, it is recommended that a single assessment paper (taken by candidates at a single assessment event) be used for distance learning candidates.

For information on normal open learning arrangements, please refer to the SQA guide *Assessment and Quality Assurance of Open and Distance Learning* (SQA 2000).

Equality and inclusion

This Unit specification has been designed to ensure that there are no unnecessary barriers to learning or assessment. The individual needs of learners should be taken into account when planning learning experiences, selecting assessment methods or considering alternative evidence.

Further advice can be found on our website www.sqa.org.uk/assessmentarrangements.

General information for candidates

Unit title: MCU/MPU I/O Hardware Control

Embedded systems are appliances which contain microprocessor-type devices hidden from the user. Common examples are video recorders and car engine management systems. They are often programmed in assembly language, and a large part of their work involves reading in and generating digital or analogue signals.

The purpose of this Unit is to provide you with the ability to write assembly language programs to transfer data using handshaking signals, read in analogue data from an analogue to digital converter, generate analogue signals with a digital to analogue converter, and handle interrupts. This will give you the necessary knowledge and skills to apply assembly language programming expertise to engineering solutions such as embedded systems.

You will have the opportunity to develop input/output programs and test them with suitable analogue signals in the laboratory. Each part of this Unit will be introduced by experiment, after which there will be an assessed practical exercise. These will be assessed mostly by laboratory reports. The lecturer will also be completing checklists to ensure that you have completed all necessary parts of the exercises. You may also be required to submit to a brief oral examination to ensure that the work you have submitted is your own and that you fully understand the material.

The centre where you are studying may choose to combine parts of this Unit with the Unit, Signal Processing and Conditioning. This will have the benefit of reducing the amount of assessment you have to undertake. It will be the responsibility of the centre to ensure that all parts of the Unit are still covered.