Unit 59: Microprocessor Systems and Applications

Unit code: Y/600/7114

QCF Level 3: BTEC Nationals

Credit value: 10

Guided learning hours: 60

Aim and purpose

This unit will introduce learners to the principles of microprocessors and give them experience of using and programming a microprocessor system for the operation or control of peripheral devices.

Unit introduction

Microprocessors can be found in a wide range of commercial, industrial and domestic applications such as electronic thermometers, weighing scales, remote controls, vending machines and cameras.

This unit will provide an introduction to the terminology (eg bits, bytes, words) and concepts (eg instructions, operation codes and operands, instructions sets, use of mnemonics, coding methods, programs, memories, assemblers, linkers and debuggers, subroutines) related to microprocessor applications.

Learners will be introduced to the ways in which microprocessor-based systems can be applied, including industrial, commercial and domestic applications. Learners will examine a range of input and output devices and consider the implications of connecting devices to a system (interfacing consideration). This will take into account signal types (eg analogue, digital) and look at the finer detail of packaging and cooling, environmental considerations, issues relating to electromagnetic compatibility (EMC) and safety.

The unit will also develop learners' understanding of the architecture and operation of microprocessor-based systems and the use of decimal, binary and hexadecimal number systems, instructions and subroutines for programming.

Finally, learners will experience the use of a microprocessor development system to prepare, run and test a typical microprocessor program application.

Learning outcomes

On completion of this unit a learner should:

1. Know how microprocessor-based systems can be applied
2. Understand the architecture and operation of a microprocessor system
3. Understand decimal, binary and hexadecimal number systems, instructions and subroutines
4. Be able to use a microprocessor development system to prepare and run a program.

Unit content

---

1 **Know how microprocessor-based systems can be applied**

Microprocessor system applications: types of application (industrial, commercial, domestic) eg robotic manufacture, vending machine, photocopier, burglar alarm, remote controls, electronic lock, electronic thermometer, keyboard interface, electronic tape measure; input/output devices eg sensors, transducers, seven-segment light emitting diode (LED), liquid crystal display (LCD), matrix and multi-line alphanumeric displays; interfacing consideration; signal types eg analogue, digital; packaging and cooling; environmental considerations; issues relating to electromagnetic compatibility (EMC), safety

2 **Understand the architecture and operation of a microprocessor system**

Architecture: central processing unit (CPU); registers; arithmetic and logic unit (ALU); instruction decoder; data paths eg internal or external busses; memory (random access (RAM), read only (ROM), erasable programmable ROM (EPROM), electrically erasable/programmable ROM (EEPROM)); input/output (I/O) ports; analogue-to-digital (ADC) and digital-to-analogue (DAC) conversion

Principles of operation: address selection and enabling; chip select/enable; consequence of data bus conflict and avoidance with a tri-state device control; address decoding; commercial chips eg 74LS138, 8205; clock (clk); read/ write function; memory map for system; fetch-execute sequence (operation of system and actions between the CPU and memory that involve instruction register and role of program counter)

3 **Understand decimal, binary and hexadecimal number systems, instructions and subroutines**

Number systems: express numbers in decimal, binary and hexadecimal; conversion between number systems; use of ASCII codes; manipulation eg binary addition/subtraction, signed binary, two's complement; bit-wise AND/OR operations

Instruction groups: eg data transfer, arithmetic, logic, branch/jump, test-compare, stack

Subroutines and the stack: subroutines and interrupt routines eg time delay routine, arithmetic operation, I/O operation; call and return instructions; purpose of the stack eg to hold subroutine addresses, data storage

4 **Be able to use a microprocessor development system to prepare and run a program**

Program operations: data handling eg data I/O, arithmetic operations, time delay routines; programming language eg hexadecimal, assembly language; graphical interpretation eg flowcharts, data flow diagrams; program function eg production of square wave, switch testing, traffic light sequencing

Enter, assemble, download, run and test a program: use of text editor eg create and store program; assemble program to create object file; download, run and test/debug eg tracing, trace table, variable watches, single stepping, breakpoints; typical program applications eg linear program, I/O initialisation, microprocessor system I/O, iteration and single loop time delay, logic operation, arithmetic operation, subroutines, macros, simple interfacing via parallel I/O port

Assessment and grading criteria

In order to pass this unit, the evidence that the learner presents for assessment needs to demonstrate that they can meet all the learning outcomes for the unit. The assessment criteria for a pass grade describe the level of achievement required to pass this unit.

Assessment and grading criteria		
To achieve a pass grade the evidence must show that the learner is able to:	**To achieve a merit grade the evidence must show that, in addition to the pass criteria, the learner is able to:**	**To achieve a distinction grade the evidence must show that, in addition to the pass and merit criteria, the learner is able to:**
P1 describe three different types of microprocessor system application	M1 compare the architecture and principles of operation of two different microprocessors	D1 explain the use of interrupts and evaluate and contrast the operation of a conventional programmed subroutine with that of an interrupt driven routine
P2 explain the architecture and principles of operation of a microprocessor-based system	M2 explain the benefits of using an assembler, and describe two examples of assembler directives	D2 produce program code to facilitate digital input and output of data using appropriate interfacing and explaining the operation of the interfacing device(s).
P3 use decimal, binary and hexadecimal number systems to represent and manipulate data [IE1, IE4]	M3 identify and correct a linear and a subroutine programming error in given fragments of program.	
P4 identify and explain a programming example for representative instructions taken from five different instruction groups		
P5 explain an example of the use of a subroutine, including the operation of call and return instructions and the function of the stack		
P6 explain the operation of a given section of a program with a specific function		

Assessment and grading criteria		
To achieve a pass grade the evidence must show that the learner is able to:	**To achieve a merit grade the evidence must show that, in addition to the pass criteria, the learner is able to:**	**To achieve a distinction grade the evidence must show that, in addition to the pass and merit criteria, the learner is able to:**
P7 use a microprocessor development system to enter, assemble, download, run and test a typical program application [IE1, IE4].		

PLTS: This summary references where applicable, in the square brackets, the elements of the personal, learning and thinking skills applicable in the pass criteria. It identifies opportunities for learners to demonstrate effective application of the referenced elements of the skills.

Key	IE – independent enquirers CT – creative thinkers RL – reflective learners TW – team workers SM – self-managers EP – effective participators

Essential guidance for tutors

Delivery

Delivery of this unit should be mainly based on practical exercises in order to give learners experience in the use of microelectronic devices. Microcontrollers are suitable for the delivery of the required knowledge, but traditional 8-bit processor training kits, for example based on the Z80, 6809, 8088 or 6502 are also appropriate. Computer simulation can also be used to give learners an opportunity to develop code, test and debug their programs before downloading the finalised software to a target system.

Learning outcome 1 requires the learner to develop an appreciation of the wide range of applications for small microprocessor systems and could be used as an introductory topic, as no previous knowledge is assumed at this point.

For learning outcome 4, a complete development system is strongly recommended. At the very least, the learner should be able to use hand-assembly techniques, writing the program first in assembly language code before converting it manually to hexadecimal machine code. This could then be input manually where facilities for electronic assembly and download of code to the target system are unavailable.

Note that the use of 'eg' in the content is to give an indication and illustration of the breadth and depth of the area or topic. As such, not all content that follows an 'eg' needs to be taught or assessed.

Outline learning plan

The outline learning plan has been included in this unit as guidance and can be used in conjunction with the programme of suggested assignments.

The outline learning plan demonstrates one way in planning the delivery and assessment of this unit.

Topic and suggested assignments/activities and/assessment
Whole-class teaching: – introduction to unit, scheme of work and assessment – describe the different types of microprocessor systems and their industrial, commercial and domestic applications – describe input/output devices, interfacing considerations, signal types, packaging and cooling requirements – describe environmental and safety considerations and EMC – practical *Small group activities:* – investigation and report on types and applications of microprocessor systems
Whole-class teaching/practical demonstration: – explain the architecture of a microprocessor system and purpose and function of component parts – explain address and chip selection and enabling and address decoding – explain consequences of data bus conflict and use of tri-state control – explain operation and applications of commercial chips *Individual learner activities:* – analysis of microprocessor system architecture
Prepare for and carry out **Assignment 1: Applications and Operation of Microprocessor Systems** (P1, P2, M1)
Whole-class teaching/practical demonstration: – explain and demonstrate the use of and conversion between decimal, binary and hexadecimal number systems – explain and demonstrate use of ASCII and bitwise operations *Individual learner activities:* – use of and conversion between number systems
Prepare for and carry out **Assignment 2: Use of and Conversion Between Number Systems** (P3)

Topic and suggested assignments/activities and/assessment
Whole-class teaching: – explain use of programming instructions taken from instruction groups for a microprocessor family – explain use of a sub-routine including the operation of call and return instructions and the function of the stack *Individual learner activities:* – investigation of instruction groups *Whole-class teaching:* – explain use of programming languages, data handling and program functions
Prepare for and carry out **Assignment 3: Instruction Groups** (P4)
Prepare for and carry out **Assignment 4: Subroutines and Program Operations** (P5, P6, M2)
Whole-class teaching/practical demonstration: – explain and demonstrate the use text editor and use of program to create object file – demonstrate process for downloading, running and testing programs *Individual learner activities:* – enter, assemble, download, run and test a program
Prepare for and carry out **Assignment 5: Developing and Testing Programs** (P7, M3, D1, D2)
Feedback on assessment and unit evaluation

Assessment

The assessment evidence for P1 could be a brief written description supported by appropriate diagrams of three different microprocessor applications. To meet the requirements of the unit content, these applications should be drawn from the different types of applications listed (industrial, commercial, domestic). For example an automotive engine management system, a photocopier and a domestic washing machine.

P2 requires learners to explain the architecture and principles of operation of a microprocessor-based system. This work can form the basis of an extended study of one of the systems that learners have examined for P1. Alternatively, a detailed study of a single microprocessor system could be used by the learner as a focus for the other criteria and in particular, the microprocessor that they will develop a program for to satisfy P7.

Evidence to support P2 would typically take the form of a summary of the features of the microprocessor-based system including the type and principal features of the CPU (eg address range, I/O facilities, internal registers, instruction set, power requirements, physical encapsulation, manufacturing technology) and each of its principal support devices (eg RAM, ROM, I/O devices, clock generator).

Learners should be able to identify the functions of each of the components and they should produce a diagram showing the architecture of the system identifying principal features (eg address bus, data bus, control bus, serial and parallel I/O). They should also be able to show that they understand the need for address decoding and they should be able to explain the function of read/write and chip enable signals.

For P3, learners need to be able to use and convert between all three number systems (decimal, binary and hexadecimal) to represent and manipulate data. This could be achieved through a series of appropriate programming examples, which are then explained (eg the result of an AND operation between two hexadecimal numbers where the learner converts the hexadecimal number to binary before applying the AND logic function to each binary digit in turn). The assessment method used should sufficiently guide learners to ensure that all aspects of the criterion and related content are covered and also enable their achievement to be tracked before the criterion is awarded. To ensure the relevance of the work for this criterion some fragmentation of the criterion might be required but this should be avoided wherever possible or at least kept to a minimum.

P4 requires learners to identify and explain a programming example for representative instructions taken from five different instruction groups. Learners could be given a selection of typical programming instructions taken from at least five major instruction groups as appropriate to the microprocessor family. These might include load and exchange, block transfer and search, arithmetic and logical, rotate and shift, bit manipulation (set, reset, test), jump, call, and return, input/output and CPU control. Learners should be able to identify and then explain what each instruction does and it might be helpful to set this within a typical example of its use.

Assessment of P5 and P6 could be linked. P5 requires learners to explain an example of the use of a subroutine (eg a time delay), including the operation of call and return instructions and the function of the stack. Provided that the function and operation of the subroutine is adequately and clearly explained, this could easily be used to cover P6 as well. The explanation of the program operations must include the way that the data is handled, the programming language used, applications of graphical interpretation and the function of the program. The content section for this criterion lists a range of examples for each of these operations.

Appropriate evidence for P7 should include documentation and tutor observation of learners' work relevant to each stage of developing and testing the program (enter, assemble, download, run and test). This will require learners to use a text editor eg to create and store the program, assemble program to create object file, download, run and test/debug the program. Typical program applications might be a linear program, I/O initialisation, microprocessor system I/O, iteration and single loop time delay, logic operation, arithmetic operation, subroutines, macros or simple interfacing via parallel I/O port.

For M1, learners should be able to extend the work done for P1 and P2 to encompass a different microprocessor, or base their work on a microcontroller device (eg PIC16C84, PIC16F877) rather than a general purpose microprocessor. In either case, learners need to compare the architecture and principles of operation of two different microprocessor systems.

In order to satisfy M2, learners need to explain the benefits of using an assembler compared with hand assembly involving entry of program instructions in hexadecimal format. They will also need to describe two examples of assembler directives eg those used for conditional assembly or those used to determine program location. Learners should provide a written description of the use of two common assembler directives, explaining how they work and what they do in conjunction with a typical assembly code routine.

For M3, learners should be provided with two code fragments containing different errors. One code fragment should contain linear code error and the other should contain a subroutine error. Typical errors might be the use of an incorrect address mode (linear error), an incorrect offset in a relative jump instruction, or incorrect use of the stack when calling a subroutine. Learners should apply standard debugging procedures, correcting the code fragment before testing it (eg by single stepping) in order to confirm its correct operation. Assessment evidence of this work is likely to be in the form of a short report supported by appropriate program trace and register dump, clearly showing the program operation before and after implementing the correction.

Learners should annotate any print outs to identify both the error and the corrective action taken to resolve the linear and the subroutine programming errors within the given fragments of program.

For D1 learners need to explain the use of interrupts and evaluate and compare the operation of a conventional programmed subroutine with that of an interrupt driven routine. A typical activity might be comparing a polled I/O routine with a comparable interrupt-driven I/O routine.

D2 requires learners to produce program code to facilitate digital input and output of data using appropriate interfacing, and an explanation of the operation of the interfacing device(s). Learners should develop, enter and test program code that will provide a means of inputting and outputting digital data in conjunction with a microprocessor system.

This exercise will require the use of one or more programmable I/O ports. The developed code should include appropriate port initialisation routines (configuring individual I/O lines for use as either input or output) as well as routines that will input and output data. A typical application might involve interfacing a number of LED indicators and switches, the switches to be configured as inputs and the LED indicators to be configured as outputs.

Learners should provide appropriate documentary evidence of the programming process (including evidence of correct program operation), an explanation of the operation of the interfacing circuit (together with relevant circuit details showing, eg pull-up resistors, LED drivers) as well as the programmable I/O device (eg showing its internal register moder and I/O address map).

Programme of suggested assignments

The table below shows a programme of suggested assignments that cover the pass, merit and distinction criteria in the assessment and grading grid. This is for guidance and it is recommended that centres either write their own assignments or adapt any Edexcel assignments to meet local needs and resources.

Criteria covered	Assignment title	Scenario	Assessment method
P1, P2, M1	Applications and Operation of Microprocessor Systems	Learners produce a report on the types and operation of microprocessor systems.	A written report and supporting diagrams.
P3	Use of and Conversion Between Number Systems	Learners need to convert between decimal, binary and hexadecimal number systems to represent and manipulate data.	A written report.
P4	Instruction Groups	Learners need to explain programming examples for instruction groups.	A written report.
P5, P6, M2	Subroutines and Program Operations	Learners investigate the operation of subroutines and a section of a programme	A written report.
P7, M3, D1, D2	Developing and Testing Programs	Learners need to use a microprocessor development system to enter and test a program.	Written documentation of practical work supported by tutor observation records and printouts.

Links to National Occupational Standards, other BTEC units, other BTEC qualifications and other relevant units and qualifications

This unit forms part of the BTEC Engineering sector suite. This unit has particular links with:

Level 1	Level 2	Level 3
		Principles and Applications of Microcontrollers

Essential resources

Centres should provide learners with access to a typical 8-bit development system, including the use of assembler software and facilities for software testing (eg single-stepping and/or program tracing). The development system could be based on traditional 8-bit processors or on one or more popular microcontrollers, eg 16C84, 16F84, 16F877.

Learners should also be provided with documentation comprising (as a minimum) fully commented instruction sets for each of the microprocessors/microcontrollers used, manufacturers' data sheets, examples of coding sheets and program documentation and instruction manuals relating to microprocessor development systems.

Employer engagement and vocational contexts

Much of the work for this unit can be set in the context of learners' work placements or be based on case studies of local employers. Further information on employer engagement is available from the organisations listed below:

- Work Experience / Workplace learning frameworks — Centre for Education and Industry (CEI -University of Warwick) — www.warwick.ac.uk/wie/cei/
- Learning and Skills Network — www.vocationallearning.org.uk
- Network for Science, Technology, Engineering and Maths Network Ambassadors Scheme — www.stemnet.org.uk
- National Education and Business Partnership Network — www.nebpn.org
- Local, regional Business links — www.businesslink.gov.uk
- Work-based learning guidance — www.aimhighersw.ac.uk/wbl.htm

Indicative reading for learners

Textbooks

Bates M — *PIC Microcontrollers* (Newnes, 2004) ISBN 0750662670

Smith D — *PIC in Practice* (Newnes, 2006) ISBN 0750668261

Tooley M — *Electronic Circuits: Fundamentals and Applications* (Third Edition, Newnes, 2006) ISBN 0750669233

Tooley M — *PC Based Instrumentation and Control* (Third Edition, Newnes, 2005) ISBN 0750647167

Delivery of personal, learning and thinking skills

The table below identifies the opportunities for personal, learning and thinking skills (PLTS) that have been included within the pass assessment criteria of this unit.

Skill	When learners are …
Independent enquirers	identifying questions to answer and problems to resolve when using a microprocessor development system to enter, assemble, download, run and test a typical program application

Although PLTS are identified within this unit as an inherent part of the assessment criteria, there are further opportunities to develop a range of PLTS through various approaches to teaching and learning.

Skill	When learners are …
Independent enquirers	planning and carrying out research and analysing and evaluating information relevant to microprocessors
Team workers	collaborating with other when working in groups to assemble and test programs

Functional Skills – Level 2

Skill	When learners are …
Mathematics	
Understand routine and non-routine problems in a wide range of familiar and unfamiliar contexts and situations	using decimal, binary, hexadecimal number systems to represent and manipulate data using a microprocessor development system to enter, assemble, download, run and test a typical program application.
English	
Writing – write documents, including extended writing pieces, communicating information, ideas and opinions, effectively and persuasively	explaining the applications and operation of a microprocessor-based system