# Unit 1.1 Systems Architecture; Lesson 2 – Activity 3

## FDE Cycle

The purpose of this task is to simulate the fetch-decode execute cycle within the class. The following roles are needed so this is best suited to work for groups of 5.

* Main Memory (will hold all of the instructions to be executed)
* Processor (completes and executes instructions)
* Program Counter (keeps track of which instructions are currently being fetched)
* Address Bus (carries the instruction number to be fetched)
* Data Bus (carries the instruction back from main memory to the processor)

Print out the following instructions for holding in Main Memory (alternatively you can create your own set of instructions)



Wave to everybody in the class and say “goodbye”

Write the name and address of somebody on the envelope

Put the paper into an envelope

Folder the paper in half again

Write your name on the piece of paper

Fold a piece of paper in half

**Instruction 6**

**Instruction 5**

**Instruction 4**

**Instruction 3**

**Instruction 2**

**Instruction 1**

## Simulation:

Using mini-whiteboard and whiteboard pens – the Program Counter will start by writing instruction 1 on a mini whiteboard – this will be passed to the address bus to transport across the room to main memory.

Main memory will look for the address on the board (in the first instance address number 1) and give the Data bus the instruction card which matches the address.

The data bus will carry this back to the processor. The processor will then carry out this instruction.

This process will be repeated for the 6 instructions.

## Extension to the activity for more able students



Write the name of somebody and address number 7 on the envelope

Put the paper into an envelope

Folder the paper in half again

Write your name on the piece of paper

Fold a piece of paper in half

**Instruction 5**

**Instruction 4**

**Instruction 3**

**Instruction 2**

**Instruction 1**

The envelope will get sent as Data back along the data bus to Address 7 (addressed by the address bus).



Open the envelope provided at the next instruction

**Instruction 6**

Students will then be simulating the returning of data back to main memory.

We’d like to know your view on the resources we produce. By clicking on ‘[Like](mailto:resources.feedback@ocr.org.uk?subject=I%20liked%20the%20OCR%20GCSE%20(9-1)%20Computer%20Science%20Challenges%20(Computational%20Logic)%20Unit%201.1%20Systems%20Architecture;%20Lesson%202%20–%20Activity%203)’ or ‘[Dislike](mailto:resources.feedback@ocr.org.uk?subject=I%20disliked%20the%20OCR%20GCSE%20(9-1)%20Computer%20Science%20Challenges%20(Computational%20Logic)%20Unit%201.1%20Systems%20Architecture;%20Lesson%202%20–%20Activity%203)’ you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click ‘Send’. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here: [www.ocr.org.uk/expression-of-interest](http://www.ocr.org.uk/expression-of-interest)

**OCR Resources**: *the small print*OCR’s resources are provided to support the teaching of OCR specifications, but in no way constitute an endorsed teaching method that is required by the Board, and the decision to use them lies with the individual teacher. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources.   
© OCR 2016 - This resource may be freely copied and distributed, as long as the OCR logo and this message remain intact and OCR is acknowledged as the originator of this work.

OCR acknowledges the use of the following content: n/a

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: [resources.feedback@ocr.org.uk](mailto:resources.feedback@ocr.org.uk)