Computer Science

A-Level

Bridging Opportunities:

Images by Pixabay.com

This booklet is designed to guide you through some key ideas and topics that would be useful to have a functional knowledge in before you start the course in September. It is broken down into 3 key sections:

- Activity 1…"Developing Python Coding Skills".
- Activity 2… "Binary Subtraction and the Hardware".
- Activity 3… "Transistors"- A Research Project.

Activity 1: Developing Python Coding Skills

Mostly, we will be using Python 3 in lessons over the duration of the course. This will eventually incorporate the use of Pygame, through which you can learn how to code games such as snake and Pac-man. You are very welcome to code in other languages whilst on the course, but improving your coding skills in Python is a great way to prepare for September. There are a huge number of online

resources to help you do this. I thoroughly recommend you start with the w3schools tutorials and work your way through them. The site is: https://www.w3schools.com/python/

Task:

Once you are confident in your coding skills, test them by picking two activities from the OCR code challenges booklet :

https://www.ocr.org.uk/Images/269433-code-challenges-for-a-level-21-40-.pdf

There are many opportunities in this document to show off your coding skills through tasks set at various levels of challenge. Pick two challenges from the booklet and complete the solution to it using Python. It would be great to see these in September. I personally use IDLE to code and run Python, which is free to download and install.

As an alternative to IDLE you can always use online python editors. I also use "trinket" which will work on smart phones, with no downloads needed.

Activity 2:

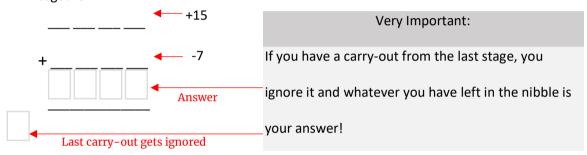## Binary Subtraction and the Hardware

Have you ever wondered how a computer uses adding circuits to subtract binary numbers? Surely, that makes no sense at all, unless of course one of the numbers is made negative. This is exactly how subtraction works in a computer. When we want to subtract one number from another, we take one number and make it negative. This process of converting a positive binary number into an equivalent negative number is done by taking what is known as the "two's complement". This isn't covered at GCSE and is the topic I would like you to investigate as a mini project here. You will get the chance to carry out two's complement and design and simulate a circuit that will be able to carry out binary addition and subtraction. The CPU consists of billions of transistors and each one of the logic gates used to make the adder circuit contains several transistors. This is going to be the subject of the research project in Activity 3.
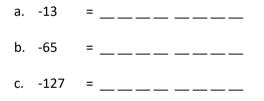
You are going to subtract 7 from 15 in binary. I will guide you through every step needed to do this ranging from introducing the theory to the implementation.

1. Convert the denary value 7 into a binary nibble... __ __ __ __

2. Convert the denary value 15 into a binary nibble... __ __ __ __

3. Take the binary nibble for 7 and change every logic "1" into a "0" and logic "0" into a "1", this is called the "one's complement". __ __ __ __

4. Now add a single bit "1" to the ones complement of 7, this is now two's complement and it represents "-7". __ __ __ __

5. You now have two numbers, one representing the binary for -7 and one for +15.

6. You can now add them together, just in the same way all binary numbers are added together:

__ __ __ __   ← +15

+ __ __ __ __   ← -7

[ ][ ][ ][ ] ← Answer
_____

[ ] ← Last carry-out gets ignored

Very Important:

If you have a carry-out from the last stage, you ignore it and whatever you have left in the nibble is your answer!

7. Of course, you will have known the answer from the start, it should be +8. Check your answer in the binary sum above is correct.

Now take the two's complement of these values, use a byte to represent each number.

   a. -13 = __ __ __ __ __ __ __ __

   b. -65 = __ __ __ __ __ __ __ __

   c. -127 = __ __ __ __ __ __ __ __

There are many online sites you can use to check your answers, one is here:

https://ncalculators.com/digital-computation/1s-2s-complement-calculator.htm

8. Now carry-out these subtractions in binary, using 2's complement, show the answers as bytes:

   a. 63-13 = __ __ __ __ __ __ __ __

   b. 76-65 = __ __ __ __ __ __ __ __

   c. 128-127 = __ __ __ __ __ __ __ __

9. Watch the video at the following website.

   https://www.youtube.com/watch?v=x_wZjDSKHPk

   The video covers full-adder circuits from 1:38 onwards.

10. Pause the video at 2:52. The screen shows what is called a 4-bit cascading full-adder circuit. You now need to build this circuit ready for simulation.

11. Go to site https://logic.ly/demo. Start a new page and start to build the circuit shown in the video, there are a few additions needed before you can simulate it. The changes are discussed in step 12.

12. $C_{0\ in}$ should be set to $\boxed{0}$─○ and all inputs A_3, A_2, A_1, A_0 should be connected individually to 4 toggle switches ▭─○ showing the logic inputs for one of the numbers. Inputs B_3, B_2, B_1, B_0 should also be connected individually to 4 toggle switches ▭─○. The outputs D_3, D_2, D_1 and D_0 should all be connected to a bulb ⊗. $C_{3\ out}$ should also be connected to a bulb to show if there is an overflow from the adder, which will ultimately get ignored!.

13. Once fully built, ensure the toggle switches for the nibble representing "-7" is set correctly on A_3, A_2, A_1, A_0 and that for "+15" set on switches B_3, B_2, B_1, B_0.

14. Simulate the circuit and note the bulbs states "on" as a logic "1" and "off" as logic "0".

15. Confirm you have the correct answer for the sum "15-7"

16. Now try simulating other numbers to confirm the correct operation of the circuit, try "12-3".

Summary of the Activity:

You have now learnt how to take the two's complement of a positive number to provide a representation of the corresponding negative number. You should have carried out binary addition of two binary numbers. You should have also built and simulated a full-adder circuit which has been cascaded to make a 4-bit binary adding circuit. This adder circuit has then been used to carry-out binary subtraction, with one number input in its two's complement form.

Activity 3:

Research Project:

How amazing is our development of technology? In a modern CPU there are over 7 billion transistors, that means there is one transistor for virtually every person walking the planet, surely that's incredible! Probably even more incredible is the journey from raw materials to the final product.

In activity 2 we looked at the full-adder circuit. This would be found in the ALU of the computer's CPU. It is amazing to discover that a CPU is fundamentally made from silicon dioxide, we know this more commonly as sand!

Task:

You should create a written report which should explain to other potential A-level students the journey that a grain of sand takes from the desert to becoming part of the CPU and the full-adder circuit. The scale is such that a single grain of sand could easily accommodate hundreds of the full adder circuits. To get you started with this project I have provided a link to a video produced by intel which nicely animates the process "from sand to silicon". It isn't narrated and by the end of the report you should be able to explain each action happening in this video.

https://www.youtube.com/watch?v=Q5paWn7bFg4

The report should cover the following aspects:

- Raw materials needed
- How transistors work.
- How logic gates such as AND, OR, NOT, NAND, NOR and XOR can be made from transistors.
- What is doping?
- What materials are involved for both N-type material and P-type material.
- How transistors are needed to carry out instructions.
- The future of CPU development... Have we reached the limit or where will all this end?

This is another good video which provides narration and more detail, to get an idea of how CPU's are made.

https://www.youtube.com/watch?v=_VMYPLXnd7E

Further Preparation.

Course Textbook:

The course we follow at Dereham Sixth Form Centre is OCR Computer Science. In my opinion the PG-Online textbook has good coverage of the course we study. I have provided a link to the PGI-online website just so you know what the book looks like and its ISBN. You can of course buy it wherever you like and there are many retailers to check for the best prices. If you were able to purchase it before the start of the course and start to browse through it to get a feel of what is covered over the two years, that would be useful, but not essential at this stage.

OCR AS and A Level Computer Science Textbook

PM Heathcote and RSU Heathcote ISBN: 978-1-910523-05-6

https://www.pgonline.co.uk/resources/computer-science/a-level-ocr/ocr-a-level-textbook/