# WEEK 1 SCRIPT -

-- PIC 1 --

Welcome to week 1 of the computer design and architecture course, in this week, we’re going to build a foundation that’s not only essential for the course, but your further studies in college and indeed your career.

This week of the course is divided into 4 parts, 3 which we will do together and the 4th will be a weekly project.

-- PIC 2 --

In part 1, we will learn about the basic logic gates, such as And, Or, Not, Nor, Nand and so on. We’ll also see a little bit (pardon the pun) of boolean logic and functions. Next, we’ll start with binary numbers - we’ll learn how to add, subtract, convert them to decimal and vice versa.

-- PIC 3 --

Now that we’re done with the boring stuff, let’s move on to part 2 where you’re going to be given a software suite with various programs and simulators which you can run on your computer and actually see the working of the gates that you’ve learnt about earlier.

Now that we have a fairly good idea of how these gates work and having seen them in action, we’re going to go about building them. Now we’re not going to be implementing any of these gates physically, we are going to build them using something called a hardware description language and simulate them using the hardware simulators provided to to you.

-- PIC 4 --

Once we have our basics set and have completed the installation of the software, we’re are going to start with the most elementary logic gate of all, the NAND gate. Now if you’ve never heard of the nand gate before, think of it as a brick, now there are many uses for a brick, but if you have enough of them you can build practically anything. We’re not going to implement this gate, we just assume that this is the abstraction of the actual physical gate. It’s going to be the brick with which we’re going to build our computer.

-- PIC 5 --

Now is the time to introduce the HACK hardware description language, as the name suggests, a hardware description language (or HDL) is used to describe the way bits will be manipulated, using this language, we’ll build the basic gates - namely And, Or and Not - using our basic building block, the nand gate. Now using these basic gates we’re to build more and more complex gates, such as Xor, Nor and Xnor and thus you’ll see your computer growing, we’re going to build it layer by layer like an onion and you’d be surprised how easy and quick it is to reach the outermost layer.

-- PIC 6 --

The Next part is very interesting to me as we’ll build the adders - namely half adder and full adder - and I hope you will share my appreciation for them once you’ve built them when you realise that you’ve just taken a bunch of dead, dumb bricks and taught them how to add numbers for you.

-- PIC 7 --

Moving on, we’ll make the simpler but extremely useful Multiplexers and DeMultiplexers, these are devices that essentially allow you to select one particular stream of information over another. For example have you ever wondered how your computer effortlessly switches from sending audio to your speakers to your headphones the moment you plug them in? That’s similar to the working of the multiplexer and demultiplexer.

-- PIC 8 --

Finally, using all this knowledge and implementing these chips and others we move onto part 4, where YOU going to make our very own Arithmetic and Logic Unit (or ALU), which is the brain of all the computations and logical operations you computer does. You will be given a schematic not unlike the picture behind me and by the end of this week, I promise you will have will have accumulated enough knowledge and skills to go ahead and implement such an integral part of your computer.

-- PIC 9 --

Congratulations! You may not realise this, but you have already built a third of your computer, but a computer is nothing if it can’t remember anything so join me in the second week to build a memory unit for our baby computer.