Sixtus CPU Manual

Owen Krupa & Michael Buzzeta

Brief Overview:

The Sixtus CPU is a 32-bit CPU and has **4 general purpose registers** which can store **one byte** each.

How to Use:

To use the machine, please follow the instructions below:

1) **Write your code** as a series of the above instructions in the .txt file named '*instructions.txt*'. Please note that while commas (,) are optional, spaces between mnemonics and operands are strictly **required**. You **must** refer to the registers as Xn (with n = 0, 1, 2, 3).
2) **Run the assembler** when you have coded your instructions. To run the assembler, go to the terminal and run the following command:
   ```
   $ python3 assembler.py
   ```
   (Authors Note: This command worked on our Ubuntu Virtual Machine, though we are unsure for other machines)
3) **Open the .circ file** named '*CPU_circuit.circ*'. Be sure to reset the simulation (CTRL + R) and find the ROM.
4) **Right-click on the ROM** and then click '*Load Image*' then select the text file named '*answer.txt*'.
5) **Move through the Clock Cycles** by going to '*Simulate*' listed at the top and select '*Auto Click Enabled*' if you wish to move through clock cycles automatically. You may also do so manually.

Instructions:

Listed below are the available instructions you may use on this machine.

Instruction	Opcode	Operands		
MOV Xd Xn	`‘0111011’`	Xn (Source)	Xd (Destination)	
MOV Xd imm	`‘0111011’`	imm	Xd	
ADD Xd Xn Xm	`‘1101011’`	Xn	Xm	Xd
SUB Xd Xn Xm	`'1101111'`	Xn	Xm	Xd
LDR Xd Xn	`'0111111'`	Xn	Xd	
DIV Xd Xn Xm/Xd	`'1000111'`	Xn	Xm	Xd
MUL Xd Xn Xm/Xd	`'1010111'`	Xn	Xm	Xd