# The Brookshear Machine

The Brookshear machine has 16 registers, numbered (in hexadecimal) 0x to Fx. There are 256 cells (bytes) in memory, numbered from 00x to FFx. Each instruction in the language is coded in two bytes. The first half-byte contains the opcode. The thirteen forms of instruction are described in the table below, with their assembly code mnemonics.

| | | |
|---|---|---|
| **0iii** | No-operation **NOP** | |
| **1rxy** | Load register **r** with contents of location **xy** | **MOV [xy] -> Rr** |
| **2rxy** | Load register **r** with value **xy** | **MOV xy -> Rr** |
| **3rxy** | Store contents of register **r** at location **xy** | **MOV Rr -> [xy]** |
| **4irs** | Move contents of register **r** to register **s** | **MOV Rr -> Rs** |
| **5rst** | Add contents of registers **s** and **t** as binary numbers, place result in register **r** | **ADDI Rs, Rt -> Rr** |
| **6rst** | Add contents of registers **s** and **t** as floating-point numbers, place result in register **r** | **ADDF Rs, Rt -> Rr** |
| **7rst** | **OR** together the contents of registers **s** and **t**, place result in register **r** | **OR Rs, Rt -> Rr** |
| **8rst** | **AND** together the contents of registers **s** and **t**, place result in register **r** | **AND Rs, Rt -> Rr** |
| **9rst** | **XOR** together the contents of registers **s** and **t**, place result in register **r** | **XOR Rs, Rt -> Rr** |
| **Ariz** | Rotate the contents of register **r** one bit to the right, **z** times | **ROT Rr, z** |
| **Brxy** | Jump to instruction at **xy** if contents of register **r** equal contents of register **0** | **JMPEQ xy, Rr** |
| **Ciii** | Halt | **HALT** |

| The following conventions are used in the above table: | |
|---|---|
| **r,s,t** | Register numbers |
| **xy** | A one-byte address or data value |
| **z** | A half byte value |
| **i** | Ignored when the instruction is de-coded: usually entered as **0** |

**References:**

The Brookshear machine design is from "Computer Science: An Overview", J. Glenn Brookshear, Prentice Hall (author's web site: http://www.mscs.mu.edu/~glennb/).

The assembly code mnemonics are from "Extended Brookshear Machine emulator and assembler", David Young, http://www.mathworks.co.uk/matlabcentral/fileexchange/22593-extended-brookshear-machine-emulator-and-assembler/content/bmcode/html/bmhelp.html