**Little Man Computer (LMC) Reference**

|  |  |  |  |
| --- | --- | --- | --- |
| Numeric code | Mnemonic code | Instruction | Description |
| 1xx | **ADD** | ADD | Add the value stored in mailbox xx to whatever value is currently on the accumulator (calculator). |
| 2xx | **SUB** | SUBTRACT | Subtract the value stored in mailbox xx from whatever value is currently on the accumulator (calculator). |
| 3xx | **STA** | STORE | Store the contents of the accumulator in mailbox xx (destructive). |
| 5xx | **LDA** | LOAD | Load the value from mailbox xx (non-destructive) and enter it in the accumulator (destructive). |
| 6xx | **BRA** | BRANCH (unconditional) | Set the program counter to the given address (value xx). That is, value xx will be the next instruction executed. |
| 7xx | **BRZ** | BRANCH IF ZERO (conditional) | If the accumulator (calculator) contains the value 000, set the program counter to the value xx. Otherwise, do nothing. |
| 8xx | **BRP** | BRANCH IF POSITIVE (conditional) | If the accumulator (calculator) is 0 or positive, set the program counter to the value xx. Otherwise, do nothing. |
| 901 | **INP** | INPUT | Go to the INBOX, fetch the value from the user, and put it in the accumulator (calculator) |
| 902 | **OUT** | OUTPUT | Copy the value from the accumulator (calculator) to the OUTBOX. |
| 0 | **HLT** | HALT | Stop working. |
|  | **DAT** | DATA | This is an assembler instruction which simply loads the value into the next available mailbox. |

xx refers to a Mailbox number (Memory Address).