# CISC 3310-TR11: MARIE Instruction Set

Instructor: Hui Chen

The basic instructions are in Table 1.

Table 1: MARIE's Instruction Set

| Opcode | | (Mnemonic) | Meaning | |
|---|---|---|---|---|
| Binary | Hex | Instruction | RTL | Description |
| 0001 | 1 | Load X | $MAR \leftarrow X$, $MBR \leftarrow M[MAR]$, $AC \leftarrow MBR$ | Load the contents of address X into AC. |
| 0010 | 2 | Store X | $MAR \leftarrow X$, $MBR \leftarrow AC$, $M[MAR] \leftarrow MBR$ | Store the contents of AC at address X. |
| 0011 | 3 | Add X | $MAR \leftarrow X$, $MBR \leftarrow M[MAR]$, $AC \leftarrow AC + MBR$ | Add the contents of address X to AC and store the result in AC. |
| 0100 | 4 | Subt X | $MAR \leftarrow X$, $MBR \leftarrow M[MAR]$, $AC \leftarrow AC - MBR$ | Subtract the contents of address X from AC and store the result in AC. |
| 0101 | 5 | Input | $AC \leftarrow InREG$ | Input a value from the keyboard into AC. |
| 0110 | 6 | Output | $OutREG \leftarrow AC$ | Output the value in AC to the display. |
| 0111 | 7 | Halt | | Terminate the program. |
| 1000 | 8 | Skipcond | See below | Skip the next instruction on condition. |
| 1001 | 9 | Jump X | $PC \leftarrow IR[11-0]$ or $PC \leftarrow X$ | Load the value of X into PC. |

The `Skipcond` instruction is the most complex one, whose RTL is as follows:

**if** $IR[11-10] = 00$ **then**
  **if** $AC < 0$ **then**
    $PC \leftarrow PC + 1$
  **end if**
**else if** $IR[11-10] = 01$ **then**
  **if** $AC = 0$ **then**
    $PC \leftarrow PC + 1$
  **end if**
**else if** $IR[11-10] = 10$ **then**
  **if** $AC > 0$ **then**
    $PC \leftarrow PC + 1$
  **end if**
**end if**

The instruction is particularly useful to realize branching and loops, in conjunctions with the `Jump X` instruction.