# Karpentium Instruction Set Architecture Development

The Karpentium processor features a 16bit wide instruction containing a four bit wide opcode; allowing for sixteen unique instructions to be implemented into the design. Four bits were chosen in regards to the goals and requirements, sixteen instructions was deemed enough to complete the requirements. Each instruction follows a basic guideline to ensure clarity and consistency. The four most significant bits contain the opcode for the instruction. The six least sig1nificant bits designate the address location of a memory interacting instruction. The proceeding instruction will contain a full sixteen bit data input if a data instruction is called.

|  |  |  |
| --- | --- | --- |
| 15 12 | **11 6** | **5 0** |
| **OPCODE** |  | **ADDRESS** |

Table 1 outlines the sixteen possible opcodes and their respective function.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Op Code** | **Bit Code** | **Function/Description** |
| **1** | HALT | 0000 | Halt |
| **2** | ADD | 0001 | Add register value to accumulator |
| **3** | SUB | 0010 | Subtract register value from accumulator |
| **4** |  | 0011 |  |
| **5** | JMP | 0100 | Jump to address |
| **6** | LDA | 0101 | Load |
| **7** |  | 0111 |  |
| **8** |  | 1000 |  |
| **9** |  | 1001 |  |
| **10** |  | 1010 |  |
| **11** |  | 1011 |  |
| **12** |  | 1100 |  |
| **13** |  | 1101 |  |
| **14** |  | 1110 |  |
| **15** |  | 1111 |  |

Table : Karpentium Opcodes

**Detailed Instructions**

|  |  |  |
| --- | --- | --- |
| 15 12 | **11** | **0** |
| **0000** |  | |

## 3.1 HALT Instruction

Usage: **HALT**

Description: Stops the processor from continuing until an external signal overrides it.

|  |  |  |
| --- | --- | --- |
| 15 12 | **11 6** | **5 0** |
| **0001** |  | **MEM ADDRESS** |

## 3.2 ADD Instruction

Usage: **ADD addr**

Description: Adds the data within the specified memory location to the Accumulator (ACC).

Flow: ACC⟵ACC + RAM[addr]

|  |  |  |
| --- | --- | --- |
| 15 12 | **11 6** | **5 0** |
| **0010** |  | **MEM ADDRESS** |

## 3.3 SUB Instruction

Usage: **SUB addr**

Description: Subtracts the data within the specified memory location, from ACC.

Flow: ACC⟵ACC - RAM[addr]

|  |  |  |
| --- | --- | --- |
| 15 12 | **11 6** | **5 0** |
| **0100** |  | **MEM ADDRESS** |

## 3.4 JMP Instruction

Usage: **JMP addr**

Description: Sets next instruction to be read by PC, to the attached memory address

Flow: PC⟵addr

|  |  |  |
| --- | --- | --- |
| 15 12 | **11 6** | **5 0** |
| **0101** |  | **MEM ADDRESS** |

## 3.5 LDA Instruction

Usage:  **LDA addr**

|  |  |  |
| --- | --- | --- |
| 15 |  | **0** |
| **16bit DATA** | | |

**DATA**

Flow: MAR⟵addr, MDR⟵data

RAM[MAR] ⟵ MDR

Description: Sets next instruction to be read from program memory as the attached memory address