4.1 Branch group of instructions

Branch instructions provide lot of convenience to the programmer to perform operations selectively, repetitively etc.

**Conditional Jump instructions**

Conditional Jump instructions in 8086 are just 2 bytes long. 1-byte opcode followed by 1-byte signed displacement (range of −128 to +127).

**Jumps Based on a single flag**

- **JZ r8** ;Jump if zero flag set (if result is 0). JE also means same.
- **JNZ r8** ;Jump if Not Zero. JNE also means same.
- **JS r8** ;Jump if Sign flag set to 1 (if result is negative)
- **JNS r8** ;Jump if Not Sign (if result is positive)
- **JC r8** ;Jump if Carry flag set to 1. JB and JNAE also mean same.
- **JNC r8** ;Jump if No Carry. JAE and JNB also mean same.
- **JP r8** ;Jump if Parity flag set to 1. JPE (Jump if Parity Even) also means same.
Ex. JNP r8 ;Jump if No Parity. JPO (Jump if Parity Odd) also means same.

Examples for JE

JO r8 ;Jump if Overflow flag set to 1 (if result is wrong)

JNO r8 ;Jump if No Overflow (if result is correct)

or JZ inst

JE is abbreviation for Jump if Equal. JNE is abbreviation for Jump if Not Equal.

JB is abbreviation for Jump if Below. JNAE is for Jump if Not Above or Equal.

JZ, JNZ, JC and JNC used after arithmetic operation

Ex. for forward jump

JE, JNE, JB, JNAE, JAE and JNB are used after a compare operation.

Only examples using JE instruction given for forward and backward jumps.

<table>
<thead>
<tr>
<th>Should be&lt;=127 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMP SI, DI</td>
</tr>
<tr>
<td>JE SAME</td>
</tr>
<tr>
<td>ADD CX, DX ;Executed if Z = 0</td>
</tr>
<tr>
<td>: (if SI not equal to DI)</td>
</tr>
<tr>
<td>:</td>
</tr>
<tr>
<td>SAME: SUB BX, AX ;Executed if Z = 1</td>
</tr>
<tr>
<td>: (if SI = DI)</td>
</tr>
</tbody>
</table>
Ex. for backward jump

```
    BACK: SUB BX,AX ;Executed if Z = 1 (if SI=DI)
            :
            :
            CMP SI, DI
            JE BACK
            ADD CX,DX ;Executed if Z = 0 (if SI <> DI)
```

Jumping beyond -128 to +127?

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Then do this!</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMP SI, DI</td>
<td>CMP SI, DI</td>
</tr>
<tr>
<td>JE SAME</td>
<td>JNE NEXT</td>
</tr>
</tbody>
</table>

What if >127 bytes

```
    ADD CX, DX
            :
            :
            SUB BX, AX
    SAME:     SUB BX, AX
```

Range for JMP (unconditional jump) can be $2^{15} = \pm 32K$. JMP instruction discussed in detail later

Source: [http://elearningatria.files.wordpress.com/2013/10/cse-iv-microprocessors-10cs45-notes.pdf](http://elearningatria.files.wordpress.com/2013/10/cse-iv-microprocessors-10cs45-notes.pdf)