

Assembler – Befehle der Minimaschine

Speicherbefehle

| | |
|----------------------|--|
| LOAD adresse | Lädt den Wert von der angegebenen Adresse in den Akkumulator. |
| LOADI zahl | Lädt die angegebenen Zahl in den Akkumulator, negative Werte sind möglich, Adressen sind nicht zulässig. |
| STORE adresse | Speichert den Wert im Akkumulator an der angegebenen Adresse. |

Arithmetikbefehle

| | |
|--------------------|---|
| ADD adresse | Addiert den Wert von der angegebenen Adresse zum Akkumulator. |
| SUB adresse | Subtrahiert den Wert der angegebenen Adresse vom Akkumulator. |
| MUL adresse | Multipliziert den Wert von der angegebenen Adresse zum Akkumulator. |
| DIV adresse | Dividiert den Wert im Akkumulator durch den Wert der angegebenen Adresse. |
| MOD adresse | Dividiert den Wert im Akkumulator durch den Wert der angegebenen Adresse und speichert den Rest im Akkumulator. |
| CMP adresse | Vergleicht den Wert der angegebenen Adresse mit dem Akkumulator und setzt Null- und Negativflag entsprechend. |
| ADDI zahl | Addiert den angegebenen Wert zum Akkumulator. |
| SUBI zahl | Subtrahiert den angegebenen Wert vom Akkumulator. |
| MULI zahl | Multipliziert den angegebenen Wert zum Akkumulator. |
| DIVI zahl | Dividiert den Wert im Akkumulator durch den angegebenen Wert. |
| MODI zahl | Dividiert den Wert im Akkumulator durch den angegebenen Wert und speichert den Rest im Akkumulator. |
| CMPI zahl | Vergleicht den angegebenen Wert mit dem Akkumulator und setzt Null- und Negativflag entsprechend. |

Bitmanipulationsbefehle

| | |
|---------------------|---|
| AND adresse | Bitweise logische Und-Verknüpfung des Werts der angegebenen Adresse mit dem Akkumulator. |
| OR adresse | Bitweise logische Oder-Verknüpfung des Werts der angegebenen Adresse mit dem Akkumulator. |
| XOR adresse | Bitweise logische Exklusiv-Oder-Verknüpfung des Werts der angegebenen Adresse mit dem Akkumulator. |
| NOT | Bitweise logische Invertierung des Werts im Akkumulator. |
| SHL adresse | Die Bitfolge im Akkumulator wird um die unter der angegebenen Adresse gespeicherten Anzahl Bits nach links geschoben. Frei werdene Bits werden mit 0-Werten belegt. |
| SHR adresse | Die Bitfolge im Akkumulator wird um die unter der angegebenen Adresse gespeicherten Anzahl Bits nach rechts geschoben. Frei werdene Bits werden mit 0-Werten belegt. |
| SHRA adresse | Die Bitfolge im Akkumulator wird um die unter der angegebenen Adresse gespeicherten Anzahl Bits nach rechts geschoben. Frei werdene Bits werden mit dem ursprünglich vordersten Bit belegt (Vorzeichenerhalt). |
| ANDI zahl | Bitweise logische Und-Verknüpfung der angegebenen Zahl mit dem Akkumulator. |

| | |
|-------------------|--|
| ORI zahl | Bitweise logische Oder-Verknüpfung der angegebenen Zahl mit dem Akkumulator. |
| XORI zahl | Bitweise logische Exklusiv-Oder-Verknüpfung der angegebenen Zahl mit dem Akkumulator. |
| SHLI zahl | Die Bitfolge im Akkumulator wird um die angegebene Anzahl Bits nach links geschoben. Frei werdene Bits werden mit 0-Werten belegt. |
| SHRI zahl | Die Bitfolge im Akkumulator wird um die angegebene Anzahl Bits nach rechts geschoben. Frei werdene Bits werden mit 0-Werten belegt. |
| SHRAI zahl | Die Bitfolge im Akkumulator wird um die angegebene Anzahl Bits nach rechts geschoben. Frei werdene Bits werden mit dem ursprünglich vordersten Bit belegt (Vorzeichenerhalt). |

Sprungbefehle

| | |
|----------------------|--|
| JMPP adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation positiv (> 0) war, d. h. weder N noch Z-Flag sind gesetzt. |
| JMPNN adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht negativ (≥ 0) war, d. h. das N-Flag ist nicht gesetzt. |
| JMPN adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation negativ (< 0) war, d. h. das N-Flag ist gesetzt. |
| JMPNP adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht positiv (≤ 0) war, d. h. das N-Flag oder das Z-Flag ist gesetzt. |
| JMPZ adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation null ($= 0$) war, d. h. das Z-Flag ist gesetzt. |
| JMPNZ adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht null ($\neq 0$) war, d. h. das Z-Flag ist nicht gesetzt. |
| JMPV adresse | Springt zur angegebenen Adresse, wenn die letzte Operation einen Überlauf verursacht hat, d. h. das V-Flag ist gesetzt. |
| JMP adresse | Springt zur angegebenen Adresse. |

Alternative Notation der Sprungbefehle

| | |
|--------------------|--|
| JGT adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation positiv (> 0) war, d. h. weder N noch Z-Flag sind gesetzt. |
| JGE adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht negativ (≥ 0) war, d. h. das N-Flag ist nicht gesetzt. |
| JLT adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation negativ (< 0) war, d. h. das N-Flag ist gesetzt. |
| JLE adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht positiv (≤ 0) war, d. h. das N-Flag oder das Z-Flag ist gesetzt. |
| JEQ adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation null ($= 0$) war, d. h. das Z-Flag ist gesetzt. |
| JNE adresse | Springt zur angegebenen Adresse, wenn das Ergebnis der letzten Operation nicht null ($\neq 0$) war, d. h. das Z-Flag ist nicht gesetzt. |
| JOV adresse | Springt zur angegebenen Adresse, wenn die letzte Operation einen Überlauf verursacht hat, d. h. das V-Flag ist gesetzt. |

Sonstige Befehle

| | |
|--------------|---|
| HOLD | Hält den Prozessor an. Dieser Befehl hat keine Adresse. |
| RESET | Setzt den Prozessor auf den Startzustand zurück. Dieser Befehl hat keine Adresse. |
| NOOP | Tut einfach nichts (NO OPeration). Dieser Befehl hat keine Adresse. |

Speicherorganisation

| | |
|------------------|---|
| WORD zahl | Besetzt eine Speicherzelle mit der angegebenen Zahl, negative Werte sind möglich. |
|------------------|---|