

Kontrollfragen – Block 5: Adressierungsarten & Programmier Techniken

1. Welche Arten von Transport-Befehlen kennen Sie?

<i>LDA</i>	<i>Load Accumulator from Memory</i>
<i>LDHX</i>	<i>Load Index Register from Memory</i>
<i>LDX</i>	<i>Load X (Index Register Low) from Memory</i>
<i>MOV</i>	<i>Move</i>
<i>STA</i>	<i>Store Accumulator in Memory</i>
<i>STHX</i>	<i>Store Index Register</i>
<i>STX</i>	<i>Store X (Index Register Low) in Memory</i>
<i>TAP</i>	<i>Transfer Accumulator to Processor Status Byte</i>
<i>TAX</i>	<i>Transfer Accumulator to X (Index Register Low)</i>
<i>TPA</i>	<i>Transfer Processor Status Byte to Accumulator</i>
<i>TSX</i>	<i>Transfer Stack Pointer to Index Register</i>
<i>TXA</i>	<i>Transfer X (Index Register Low) to Accumulator</i>
<i>TXS</i>	<i>Transfer Index Register to Stack Pointer</i>

2. Was bewirkt der Befehl TAP bei der HCS08 CPU?

<i>TPA</i>	<i>Transfer Processor Status Byte to Accumulator</i>
------------	------------------------------------------------------

Der Inhalt vom Akku wird in das CCR (Condition Code Register), „Flag-Register“ übertragen.

3. Welche arithmetischen Flags (Condition Codes) der HCS08 CPU kennen Sie?

<i>C</i>	<i>Carry</i>
<i>Z</i>	<i>Zero</i>
<i>N</i>	<i>Negative</i>
<i>I</i>	<i>Interrupt Mask</i>
<i>H</i>	<i>Half-Carry (From Bit 3)</i>
<i>V</i>	<i>Two's Complement Overflow</i>

4. In welchem Register wird beim HCS08 das Resultat arithmetischer Operationen gespeichert?

Im Akku.

5. Woran erkennt die ALU ob bei den Befehlen ADD und SUB signed oder unsigned gerechnet werden muss?

Gar nicht, es spielt für ADD / SUB keine Rolle.