

Kontrollfragen – Block 4: Assemblerdirektiven & Adressierungsarten

1. Was bewirkt die Assembler-Direktive DS.W 3 ?

Reserviert Speicher (RAM) für Variablen. Hier 3 Word-Variablen.

2. Wie viele Buszyklen sind für die Ausführung des Befehls LDA \$10A1 erforderlich?

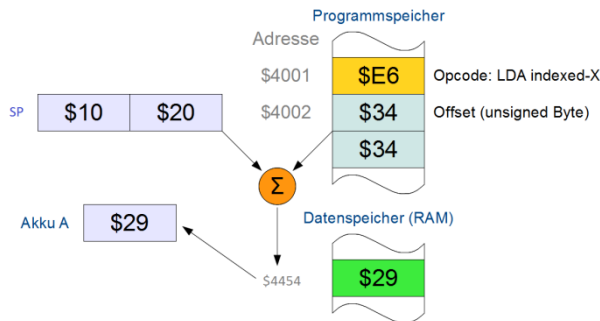
Source Form	Addr. Mode	Machine Code		HCS08 Cycles	Access Detail
		Opcode	Operand(s)		
LDA #opr8l	IMM	A6	ii	2	pp
LDA opr8a	DIR	B6	dd	3	rpp
LDA opr16a	EXT	C6	hh ll	4	prpp
LDA oprx16,X	IX2	D6	ee ff	4	prpp
LDA oprx8,X	IX1	E6	ff	3	rpp
LDA ,X	IX	F6		3	rpf
LDA oprx16,SP	SP2	9ED6	ee ff	5	pprpp
LDA oprx8,SP	SP1	9EE6	ff	4	prpp

LDA opr16a → 4 Buszyklen

3. Warum gibt es keinen STA-Befehl mit immediate Adressierung?

STA speichert den Akku im Memory, dazu muss zwingend eine Adresse angegeben werden.

4. Erklären Sie im Detail die Funktionsweise der indexierten Adressierungsart SP2 !



Für die indexierte Adressierung wird der Stackpointer (SP) verwendet – und 16-bit Offset.

Bsp: LDA oprx16,SP

Nach dem OPCode wird der 16-bit Offset gelesen und zum Wert im Stackpointer (SP) addiert. Danach wird der Wert an der berechneten Adresse in den Akku geladen.

5. Wie viele Bytes Programmspeicher benötigt der Befehl LDHX \$5B,SP ?

Source Form	Addr. Mode	Machine Code		HCS08 Cycles	Access Detail
		Opcode	Operand(s)		
LDHX #opr16l	IMM	45	jj kk	3	ppp
LDHX opr8a	DIR	55	dd	4	rrpp
LDHX opr16a	EXT	32	hh ll	5	prpp
LDHX ,X	IX	9EAE		5	prfp
LDHX oprx16,X	IX2	9EBE	ee ff	6	pprrpp
LDHX oprx8,X	IX1	9ECE	ff	5	prpp
LDHX oprx8,SP	SP1	9EFE	ff	5	prpp

LDHX oprx8, SP → Opcode: 9EFE ff → 3 Bytes

6. Erklären Sie den Sprungbereich der relativen Adressierung!

Der Sprungbereich bei der relativen Adressierung beträgt -126 ... + 129.

Das auf den Opcode folgende Byte wirkt als 2er-Komplement Offset zum bereits weitergezählten Programm Counter. 8Byte wäre -128 .. + 127, da nun aber der PC +2 ist, ist der Sprungbereich -126 ... + 129 relativ zum Ort wo der eigentliche Opcode für den Branch-Befehl steht.