

# Sprungbefehle

Mnemonicischer Befehl OpCode Operanden		Hex-Code	W	M	Beeinflussung Zustandsbits			Befehlsbeschreibung
			B	Z	CY	OV	AC	
LJMP	adr16	02	3	2	--	--	--	Programmsprung im 64K-Block
SJMP	rel	80	2	2	--	--	--	relativer Programmsprung im Bereich -128 bis +127 zur nachfolgenden Befehlsadresse
AJMP	adr11	01-E1	2	2	--	--	--	Sprung im 2k-Block
JMP	@A+DPTR	73	1	2	--	--	--	Springe zur Adresse, die aus Akku- und Datenpointerinhalt gebildet wird.
JBC	badr,rel	10	3	2	--	--	--	Springe bei <i>gesetztem</i> Bit und lösche es
JB	badr,rel	20	3	2	--	--	--	Springe bei <i>gesetztem</i> Bit
JNB	badr,rel	30	3	2	--	--	--	Springe bei <i>gelöshtem</i> Bit
JC	rel	40	2	2	--	--	--	Springe bei <i>gesetztem</i> Carry-Bit
JNC	rel	50	2	2	--	--	--	Springe bei <i>gelöshtem</i> Carry-Bit
JZ	rel	60	2	2	--	--	--	Springe, wenn Akkuinhalt <i>gleich Null</i>
JNZ	rel	70	2	2	--	--	--	Springe, wenn Akkuinhalt <i>ungleich Null</i>
DJNZ	Rn,rel	D8-DF	2	2	--	--	--	Vermindere Register um Eins und springe, wenn der Rest ungleich Null
DJNZ	dadr,rel	D5	3	2	--	--	--	Vermindere den Speicherinhalt im internen RAM um Eins und springe, wenn der Rest ungleich Null.
CJNE	A,#c8,rel	B4	3	2	■	--	--	Vergleiche Akku mit Konstante und verzweige bei Ungleichheit. Andernfalls fahre im Programm fort.
CJNE	Rn,#c8,rel	B8-BF	3	2	■	--	--	Vergleiche Register mit Konstante und verzweige bei Ungleichheit, andernfalls fahre fort.
CJNE	A,dadr,rel	B5	3	2	■	--	--	Vergleiche Akku- und Speicherinhalt und verzweige bei Ungleichheit.
CJNE	@R0,#c8,rel	B6	3	2	■	--	--	Vergleiche den Inhalt des RAM-internen Speichers mit der Konstante und verzweige bei Ungleichheit. (R0 bzw. R1 enthält die Quellenadresse)
CJNE	@R1,#c8,rel	B7	3	2	■	--	--	
LCALL	Adr16	12	3	2	--	--	--	Unterprogrammaufruf im 64k-Block
ACALL	adr11	11-F1	2	2	--	--	--	Unterprogrammaufruf im 2k-Block
RET		22	1	2	--	--	--	Ende Unterprogramm
RETI		32	1	2	--	--	--	Ende UP plus löschen des INT-Flags