

Entwicklerdokumentation

Mikroprogrammierung Aufgabe 2.35

IFETCH (Opcode 0x0)

ie	irpt	k	cnst	AM 2901				Y MUX		AM 2904				AM 2910		BAR	BZ										
				src	func	dest	ra	asel	rb	bsel	abus	dbus	cin	shift	ceμ	cem	sr	test	ccen	cmd	bar	ld	ed	inc	ea	ird	mwe
DIS	X	X	X	ZB	ADD	RAMF	X	X	F	MR	H	H	C1	X	H	H	X	X	PS	CONT	X	H	H	H	E	H	R
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	X	X	PS	CONT	X	H	H	I	H	L	R
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	X	X	PS	JMAP	X	H	H	H	H	H	R

Das Mikroprogramm IFETCH wurde modifiziert um den Befehlszähler zu realisieren. Das ALU Register r15 wird inkrementiert indem der Wert mit 0 und dem Carry-In Bit 1 addiert und wieder in das Register zurückgeschrieben wird. Gleichzeitig wird im ersten Takt des Mikroprogramms die normale Funktionalität von IFETCH im ersten Takt ausgeführt.

ADD RA, RB (Opcode 0x10)

ie	irpt	k	cnst	AM 2901				Y MUX		AM 2904				AM 2910		BAR	BZ										
				src	func	dest	ra	asel	rb	bsel	abus	dbus	cin	shift	ceμ	cem	sr	test	ccen	cmd	bar	ld	ed	inc	ea	ird	mwe
DIS	X	X	X	AB	ADD	RAMF	X	IR	X	IR	H	H	C10	X	H	L	X	X	PS	CJP	IFETCH	H	H	H	H	H	R

Im Mikroprogramm ADD RA, RB werden beide Operanden addiert und in RB zurückgeschrieben. Dies geschieht indem die übergebenen Werten A und B die Register RA und RB im Baustein AM2901 adressieren. Die Werte aus den Registern werden ausgelesen und die Werte über die ALU-Quelloperandensteuerung an die ALU-Eingänge R und S angelegt.

Die beiden Werte werden in der ALU aufaddiert und das Ergebnis an den F Ausgang gegeben. Als Ergebnis der Berechnung wird der Wert in das Register RB zurückgeschrieben. Der Wortrandlogikbaustein AM2904 erhält die Statussignale die die Berechnung erzeugt hat um sie in das Maschinenstatusregister zu schreiben. Nach Ausführung von ADD wird wieder zu IFETCH gesprungen.

JMPNZ imm (Opcode 0x20)

ie	irpt	k	cnst	AM 2901				Y MUX		AM 2904				AM 2910		BAR	BZ										
				src	func	dest	ra	asel	rb	bsel	abus	dbus	cin	shift	ceμ	cem	sr	test	ccen	cmd	bar	ld	ed	inc	ea	ird	mwe
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	MSR	NZ	C	CJP	34	H	H	H	H	H	R
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	X	X	PS	CJP	IFETCH	H	H	I	H	H	R
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	X	X	PS	CONT	X	H	H	H	E	H	R
DIS	X	X	X	X	X	NOP	X	X	X	X	H	H	X	X	H	H	X	X	PS	CJP	IFETCH	L	H	H	H	H	R

Dieser Befehl führt einen bedingten Sprung aus, falls das Zero-Flag im Maschinenstatusregister nicht gesetzt ist. Das Zero-Flag die durch vorhergehende Operationen, z.B. Compare, arithmetische Operationen oder ähnliches gesetzt wurden. Dafür werden 2 Takte bei keinem Sprung bzw. 3 Takte bei dem Fall eines Sprungs benötigt.

Im ersten Takt wird das Zero Flag im Maschinenstatusregister im Baustein AM2910 auf Null geprüft. Ist das Flag gesetzt wird zu Möglichkeit II gesprungen, ist es nicht gesetzt so wird das Mikroprogramm bei Möglichkeit I fortgesetzt.

Möglichkeit I:

Es erfolgt kein Sprung, es wird also zu IFETCH zurückgesprungen und die Maschinenbefehlsausführung fortgesetzt. Dabei muss der Befehlszähler inkrementiert werden, da dieser sonst auf den Immediate-Wert zeigen würde.

Möglichkeit II:

Es erfolgt ein Sprung. Um dies zu realisieren muss nun im zweiten Takt (Zeile 3, Mikroprogramm-speicher 0x22) der Immediate Wert aus dem Hauptspeicher geladen werden. Dieser steht eine Speicherzelle nach dem Befehl, also in dieser auf die auch der Befehlszähler zeigt. Durch Setzen des Instruktionsbits "Befehlszähler Enable Adressbus" auf 0 wird der Inhalt des Befehlszählers auf den Adressbus gegeben.

Im dritten und letzten Takt wird der Befehlszähler mit dem Datum aus dem Hauptspeicher auf den Datenbus geladen, das die Adresse darstellt auf die gesprungen werden soll. Damit wird das Mikroprogramm geladen zu dem gerade gesprungen wurde.

IEC RB (Opcode 0x30)

		AM 2901				Y MUX		AM 2904				AM 2910	BAR		BZ												
ie	irpt	k	cnst	src	func	dest	ra	asel	rb	bsel	abus	dbus	cin	shift	ceμ	cem	sr	test	ccen	cmd	bar	ld	ed	inc	ea	ird	mwe
DIS	X	X	X	ZA	ADD	RAMF	F	MR	X	IR	H	H	CIO	X	H	H	X	X	PS	CJP	IFETCH	H	H	H	H	H	R

Diese Funktion lädt den aktuellen Wert des Befehlszählers in das übergebene Register RB. Der Befehlszähler wird durch eine abgewandelte IFETCH-Methode realisiert da jedes Mikroprogramm von IFETCH geladen wird und der Befehlszähler dadurch vor jeder Maschinenbefehlsausführung inkrementiert wird.

Der aktuelle Maschinenbefehlszählerstand wird ausgelesen in dem das Register r15 in dem der Wert enthalten ist mit 0 addiert wird. Quelloperanden sind also Zero und das r15 Register, Carry-In wird ebenfalls auf 0 gesetzt. Das Ergebnis ist der Wert der nun ohne Veränderung durch die ALU durchgereicht wurde und nun am F Ausgang anliegt wo er nun in das Register RB abgelegt wird um dem Programmierer zur Verfügung zu stehen. Nach der Ausführung wird noch im selben Takt zu IFETCH gesprungen.