

ETI-Praktikum: Mikroprogrammierung - Gruppe 7

Team:

Philip Lorenz

Philip Daubmeier

Ferdinand Mayet

Aufgabenstellung

Zu implementierende Mikroprogramme:

- IEC RB

Gibt Stand des Maschinenbefehlszählers zurück

- ADD RA, RB

Addiert zwei vorzeichenbehaftete 16-Bit Zahlen und legt das Ergebnis in RB ab

- JMPNZ imm

Führt einen Sprung an Adresse imm aus wenn Zero Flag nicht gesetzt ist.

IEC RB

		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	C10	X	H	H	X	X	PS	CJP	IFETCH	H	H	H	H	H	H	R

- Auslesen des Befehlszählers in das übergebene Register RB.

IFETCH

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	irld	mwe
DIS	X	X	X	ZB	ADD	RAMF	X	X	F	MR	H	H	CI1	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

- Verändertes IFETCH um Befehlszähler zu inkrementieren
- Kein weiterer Takt erforderlich
→ Performance wird nicht beeinträchtigt

JMPNZ imm

	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	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

- Bei Zero Flag = 0:
 - Sprung zu Adresse imm, in dem der Befehlszähler auf den Adressbus gelegt wird. Im nächsten Takt erhalten wir den imm Wert vom Datenbus und geben diesen auf den Befehlszähler
- Sonst:
 - Sprung zu IFETCH und Befehlsausführung an vorhergehender Stelle fortsetzen

ADD RA RB

		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	AB	ADD	RAMF	X	IR	X	IR	H	H	C10	X	H	L	X	X	PS	CJP	IFETCH	H	H	H	H	H	R

- Addiert die zwei Zahlen in den Registern RA und RB
- Legt Ergebnis in RB ab.

Beispielprogramm

Beispiel: vorher R0 = #ffff und R1 = #5 setzen

Speicherzelle	Inhalt	Mnemo	Beschreibung
0	101	ADD R0, R1	Addiert R0 auf R1
1	300	JMPNZ 0	Springt auf Adresse 0 wenn
2	0		Zero Flag nicht gesetzt ist
3	202	IEC R2	Speichert Befehlszähler in R2