

HÖHERE TECHNISCHE BUNDESLEHRANSTALT

HOLLABRUNN

Höhere Abteilung für Elektronik – Technische Informatik

Klasse / Jahrgang: 4BHEL	Übungsbetreuer: Dipl. Ing. Josef Reisinger
Übungsnummer: ASM_Z80/1	Übungstitel: 2er-Lauflicht-Zeit
Datum der Vorführung: 11.10.2019	Gruppe: Platajs Martin
Datum der Abgabe: 23.10.2019	Unterschrift:

Beurteilungskriterien

Programm:	Punkte
Programm Demonstration	
Erklärung Programmfunktionalität	
Protokoll:	Punkte
Pflichtenheft (Beschreibung Aufgabenstellung)	
Beschreibung SW Design (Flussdiagramm, Blockschaltbild,..)	
Dokumentation Programmcode	
Testplan (Beschreibung Testfälle)	
Kommentare / Bemerkungen	
Summe Punkte	

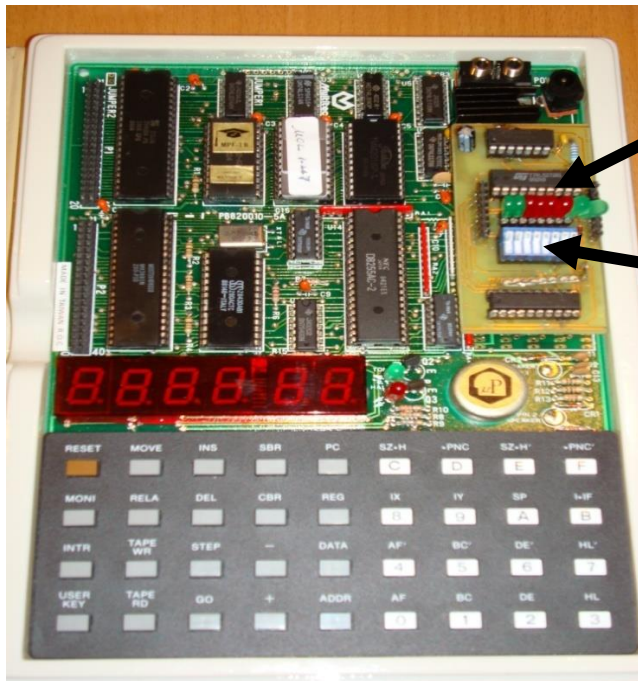
Note: _____

Inhaltsverzeichnis

1	Produktanforderungen.....	3
2	Softwaredesign.....	4
3	Speicher / Registerbelegung.....	5
4	Berechnung der verwendeten Warteschleife	5
5	Programmierung	6
6	Testdaten	8
7	Abschließende Bemerkungen	8
7.1	Probleme	8
7.2	Erkenntnisse	8
8	Zeitaufwand.....	8

1 Produktanforderungen

Es ist ein mit Unterprogrammen strukturiertes Programm für den „Microprofessor“ μ PF1 zu schreiben, welches ein links / rechts umschaltbares Lauflicht mit Dunkelphase (1 oder 2 Bit rotieren) und mit einstellbarer Zeit realisiert.



8 LED des μ PF1
(L7, L6, L5, L4, L3, L2, L1, L0)

8 Schalter des μ PF1
(S7, S6, S5, S4, S3, S2, S1, S0)

Schalter: oben => "1"
unten => "0"

Schalter - Belegung:

S7	S6	S5	S4	S3	S2	S1	S0
0...Linkslauf 1...Rechtslauf	0...1 Bit 1...2 Bit	Zeit 0,1 bis 6,3 Sekunden einstellbar					

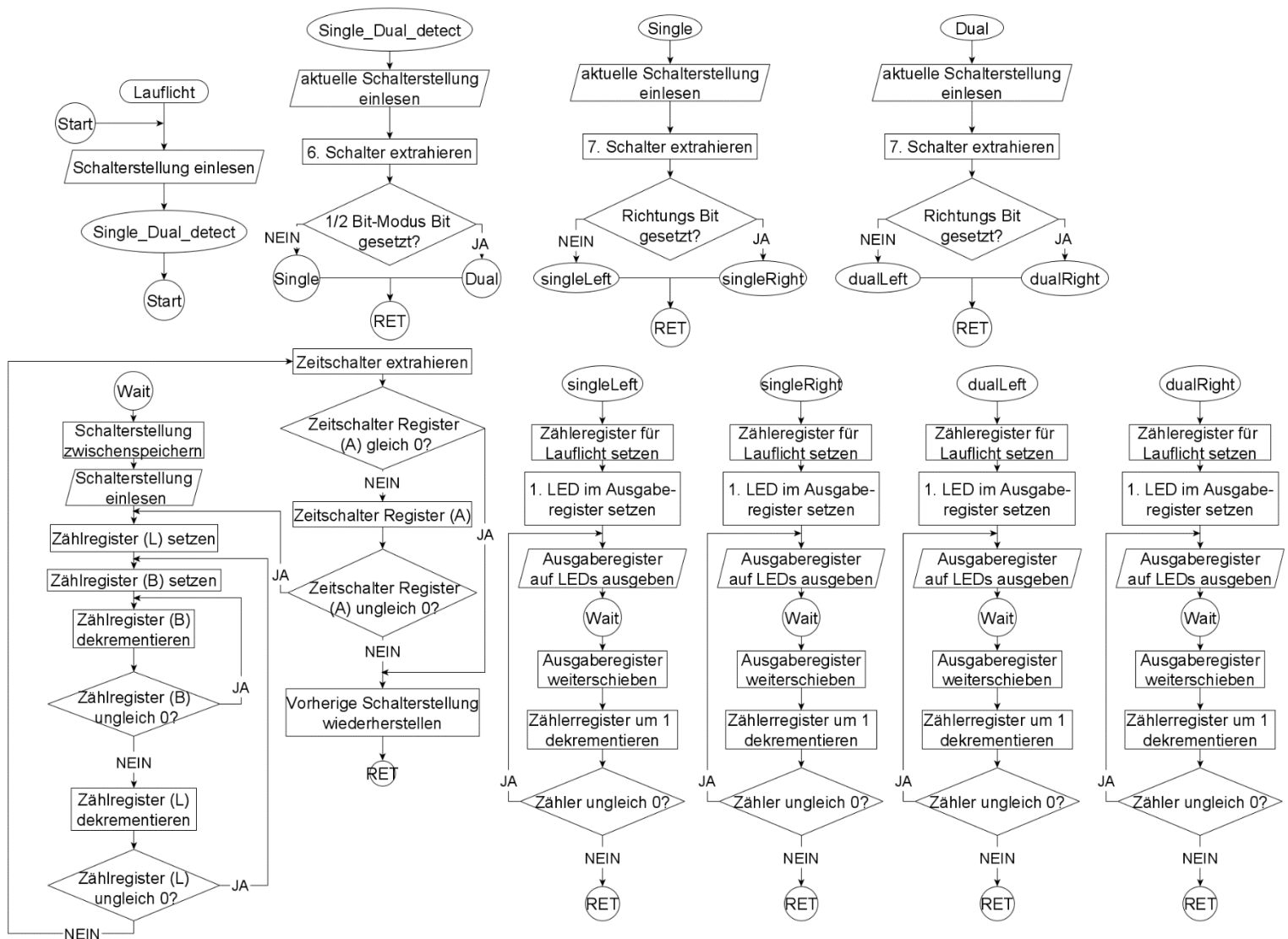
LED - Belegung:

L7	L6	L5	L4	L3	L2	L1	L0
LED 7	LED 6	LED 5	LED 4	LED 3	LED 2	LED 1	LED 0

Mit S7 wird entschieden ob das Lauflicht nach links oder nach rechts läuft. Mit S6 wird bestimmt ob 1 oder 2 Bits durchlaufen. Die restlichen Schalter S5-S0 sind für die einstellbare Zeit zuständig. Die Dunkelphase ist jene Zeit, in welcher keine LED leuchtet. Sie tritt zwischen den Durchläufen der LEDs auf, außerdem ist sie so lang wie die eingestellte Zeit.

Wenn die Schalter S5-S0 alle 0 sind liegen 0,1 Sekunden zwischen dem Weiterschalten der LEDs. Die Richtung der LEDs wird erst nachdem einem vollständigen durchlauf geändert, während die Zeit sofort übernommen wird, wenn die Schalter geändert werden.

2 Softwaredesign



3 Speicher / Registerbelegung

Der **RAM Speicher** wurde lediglich für den Programmcode verwendet.

Verwendete Register:

A	Diverses
B	Zählregister für Warteschleife
C	Zählregister für Lauflicht
D	Hilfsregister zum Zwischenspeichern vom A Register
L	Zählregister für Warteschleife

I/O-Bausteine:

LED/Schalter Die LED/Schalter Platine wurde unter der Adresse C0_H angesprochen.

4 Berechnung der verwendeten Warteschleife

Label	Programmcode	Taktzyklen	Anzahl der Durchläufe
Wait:	LD D,A	4	1
	IN A,(C0)	11	1
Loop:	LD L, #32	7	A
outer:	LD B, #00	7	A*50
inner:	DEC B	4	A*50*256
	JP NZ, inner	10	A*50*256
	DEC L	4	A*50
	JP NZ, outer	10	A*50
	AND #3F	7	A
	JP Z, skip	10	A
	DEC A	4	A-1
	JP NZ, Loop	10	A-1
skip:	LD A,D	4	1
	RET	10	1

1 Taktzyklus des µPF1 dauert $1 / 1,79\text{MHz} = 0,56 \mu\text{s}$

Dauer Warteschleife:

$0,56 \mu\text{s} * (4+11+7*A+7*A*50+4*A*50*256+10*A*50*256+4*A*50+10*A*50+7*A+10*A+4*(A-1)+10*(A-1)+4+10)$
 = ? (von A Register abhängig)

5 Programmlisting

Adresse	OP-Code	Label	Mnemonik	Kommentare
1800	DB C0	Main:	IN A,(C0)	aktuelle Schalterstellungen einlesen
			CALL	
1802	CD 10 18		singledualdetect	überprüfen ob single or dual Modus
1805	C3 00 18		JP Main	
1810	DB C0	Single_dual_detect:	IN A, (C0)	aktuelle Schalterstellungen einlesen
1812	E6 40		AND #40	7. und 5.-0. Schalter maskieren
1814	C2 1B 18		JP NZ, D	wenn 6. Schalter 1 => dual sonst single
1817	CD 30 18		CALL single	eine LED läuft
181A	C9		RET	
181B	CD 90 18	D:	CALL dual	zwei LEDs laufen
181E	C9		RET	
1830	DB C0	single:	IN A, (C0)	aktuelle Schalterstellungen einlesen
1832	E6 80		AND #80	6.- 0. Schalter maskieren
1834	CA 3B 18		JP Z, L	wenn 7. Schalter 0 => Linkslauf sonst Rechtslauf
1837	CD 50 18		CALL singleRight	1 Bit Rechtslauf
183A	C9		RET	
183B	CD 70 18	L:	CALL singleLeft	1 Bit Linkslauf
183E	C9		RET	
1850	0E 09	singleRight:	LD C, #09	Zähler setzten damit LEDs durchlaufen + Dunkelzeit
1852	3E 80		LD A, #80	1. LED setzen
1854	D3 C0	Loop:	OUT (C0), A	Ausgabe auf LEDs
1856	CD EF 18		Call Wait	Warteschleife aufrufen
1859	CB 3F		SRL A	Register weiterschieben
185B	0D		DEC C	Zähler um eins verringern
185C	C2 54 18		JP NZ, Loop	Solange LEDs nicht durchgelaufen => wiederholen
185F	C9		RET	
1870	0E 09	singleLeft:	LD C, #09	Zähler setzten damit LEDs durchlaufen + Dunkelzeit
1872	3E 01		LD A, #01	1. LED setzen
1874	D3 C0	Loop:	OUT (C0), A	Ausgabe auf LEDs
1876	CD EF 18		Call Wait	Warteschleife aufrufen
1879	CB 27		SLA A	Register weiterschieben
187B	0D		DEC C	Zähler um eins verringern
187C	C2 74 18		JP NZ, Loop	Solange LEDs nicht durchgelaufen => wiederholen
187F	C9		RET	

1890	DB C0	dual:	IN A, (C0)	aktuelle Schalterstellungen einlesen
1892	E6 80		AND #80	6.- 0. Schalter maskieren
1894	CA 9B 18		JP Z, L	wenn 7. Schalter 0 => Linkslauf sonst Rechtslauf
1897	CD B0 18		CALL dualRight	2 Bit Rechtslauf
189A	C9		RET	
189B	CD D0 18	L:	CALL dualLeft	2 Bit Linkslauf
189E	C9		RET	
18B0	0E 09	dualRight:	LD C, #09	Zähler setzten damit LEDs durchlaufen + Dunkelzeit
18B2	3E C0		LD A, #C0	1. 2 LEDs setzen
18B4	D3 C0	Loop:	OUT (C0), A	Ausgabe auf LEDs
18B6	CD EF 18		Call Wait	Warteschleife aufrufen
18B9	CB 3F		SRL A	Register weiterschieben
18BB	0D		DEC C	Zähler um eins verringern
18BC	C2 B4 18		JP NZ, LoopRe2	Solange LEDs nicht durchgelaufen => wiederholen
18BF	C9		RET	
18D0	0E 09	dualLeft:	LD C, #09	Zähler setzten damit LEDs durchlaufen + Dunkelzeit
18D2	3E 03		LD A, #03	1. 2 LEDs setzen
18D4	D3 C0	Loop:	OUT (C0), A	Ausgabe auf LEDs
18D6	CD EF 18		Call Wait	Warteschleife aufrufen
18D9	CB 27		SLA A	Register weiterschieben
18DB	0D		DEC C	Zähler um eins verringern
18DC	C2 D4 18		JP NZ, Loop	Solange LEDs nicht durchgelaufen => wiederholen
18DF	C9		RET	
18EF	57	Wait:	LD D,A	Registerstand zwischenspeichern
18F0	DB C0		IN A,(C0)	aktuelle Schalterstellungen einlesen
18F2	2E 19	Loop:	LD L, #19	25 bzw 50 * 4ms bzw 2ms
18F4	06 00	outer:	LD B, #00	2ms Schleife
18F6	05	inner:	DEC B	
18F7	C2 F6 18		JP NZ, inner	
18FA	2D		DEC L	
18FB	C2 F4 18		JP NZ, outer	
18FE	E6 3F		AND #3F	variable Zeit einlesen (A Register) * 100ms
1900	CA 07 19		JP Z, skip	wenn A 0 soll nur 100ms gewartet werden
1903	3D		DEC A	
1904	C2 F2 18		JP NZ, Loop	
1907	7A	skip:	LD A,D	Vorheriger Registerstand wiederherstellen
1908	C9		RET	

6 Testdaten

Schalter	Wirkung	Anmerkung
Linkslauf mit 1Bit	1 Bit läuft beginnend von rechts nach links durch	
Linkslauf mit 2 Bit	2 Bits laufen beginnend von rechts nach links durch	
Rechtslauf mit 1Bit	1 Bit läuft beginnend von links nach rechts durch	
Rechtslauf mit 2 Bit	2 Bits laufen beginnend von links nach rechts durch	
Einstellbare Zeit auf 0 gestellt	Jede LED leuchtet einzeln 0,1s auf und sind nach dem Durchlauf (nach 0,8s) 0,1s dunkel (Dunkelphase).	
Einstellbare Zeit auf 25 gestellt	Jede LED leuchtet einzeln 2,5s auf und sind nach dem Durchlauf (nach 20s) 2,5s dunkel (Dunkelphase).	

7 Abschließende Bemerkungen

7.1 Probleme

Beim Erstellen des Programms wurde zuerst nicht berücksichtigt, dass Register in Unterprogrammen überschrieben wurden, ohne den Inhalt zwischenzuspeichern.

7.2 Erkenntnisse

- Umgang mit der Hardware µPF1
- Grundsätze der Assemblerprogrammierung kennengelernt
- Übersetzung in den OP-Code
- Richtiges setzen von Adressen
- Fehlersuche in einem Assemblerprogramm

8 Zeitaufwand

Tätigkeit	Aufwand
Erstellung des Pflichtenhefts	0,5h
Erstellung des Systemdesign (Flussdiagramm bzw. Struktogramm und ev. UI Design)	3h
Programmcodierung (incl. Fehlersuche)	4h
Testen der Software	2h
Dokumentation (Protokoll)	3h
Gesamt:	12,5h