

			Rahmen
INT_CB1	EQU	\$0038	; Register und Speicherzellen
INT_CB2	EQU	\$003C	; Interrupt-Vektor für CB1
INT_T3	EQU	\$002E	; Interrupt-Vektor für CB2
INT_V24	EQU	\$0040	; Interrupt-Vektor für Timer #3
CR_T1	EQU	\$F018	; Interrupt-Vektor für V. 24-Schnittstelle
CR_T3	EQU	\$F018	; Steuerregister Timer #1
CR_T2	EQU	\$F019	; Steuerregister Timer #3
SR_T	EQU	\$F019	; Steuerregister Timer #2
L_T3	EQU	\$F01E	; Timer-Statusregister
DRB	EQU	\$F002	; Latches für Timer #3
DDRB	EQU	\$F002	; Datenregister Port PB
CRB	EQU	\$F003	; Datenrichtungsregister Port PB
DR_V24	EQU	\$F008	; Steuerregister Port PB
SR_V24	EQU	\$F009	; Datenregister V24-Schnittstelle
IR_V24	EQU	\$F00A	; Statusregister V24-Schnittstelle
CR_V24	EQU	\$F00B	; Befehlsregister V24-Schnittstelle
			; Monitorroutinen
T7SG	EQU	\$F100	; Tetrade in 7-Segment-Code umwandeln
B7SG	EQU	\$F103	; 2 Tetraden in 7-Segment-Code umwandeln
D7SG	EQU	\$F106	; 4 Tetraden in 7-Segment-Code umwandeln
CLRDISP	EQU	\$F110	; Anzeige löschen
SHOWA	EQU	\$F113	; Akku A in Anzeige binär darstellen
SHOWD	EQU	\$F116	; Akku D=(A, B) in Anzeige binär darstellen
SHOWYD	EQU	\$F119	; Y und D in Anzeige binär darstellen
SHOWT7SG	EQU	\$F11C	; Tetrade im 7-Segment-Code darstellen
SHOWB7SG	EQU	\$F120	; 2 Tetraden im 7-Segment-Code darstellen
SHOWD7SG	EQU	\$F123	; 4 Tetraden im 7-Segment-Code darstellen
CLDBUF	EQU	\$F130	; Puffer löschen
SHOWDBUF	EQU	\$F133	; Puffer in Anzeige bringen
RRDBUF	EQU	\$F136	; Puffer um 1 Position nach rechts rotieren
RLDBUF	EQU	\$F139	; Puffer um 1 Position nach links rotieren
COPYDBUF	EQU	\$F13C	; Puffer im Speicher kopieren
KEY	EQU	\$F140	; Tastaturabfrage ohne Halt
HALTKEY	EQU	\$F143	; Tastaturabfrage mit Halt
SHOWKEY	EQU	\$F146	; Tastatur lesen und anzeigen ohne Halt
SHOWHKEY	EQU	\$F149	; Tastatur lesen und anzeigen mit Halt
INDATA	EQU	\$F14C	; Byte einlesen
SHOWDATA	EQU	\$F150	; 2 Bytes einlesen
INADR	EQU	\$F153	; 4-stellige Adresse einlesen
SHOWADR	EQU	\$F156	; 4-stellige Adresse einlesen und darstellen
DLY1MS	EQU	\$F160	; 1 ms verzögern
RANDOM	EQU	\$F163	; Zufallszahl erzeugen
COPYXD	EQU	\$F166	; Speicherbereich kopieren

ORG        \$0400    ; Programmzähler auf \$0400 stellen

             ; Hier folgt Ihr Programm

END