M. Befehle in Installationsdateien

Installationsdateien haben immer die Namenserweiterung '.INS'. Sie sind reine Textdateien, die mit jedem beliebigen Editor erstellt werden können. Jede Zeile beginnt mit einem Schlüsselwort (= Befehl) oder mit einem Kommentarzeichen, Leerzeilen sind zulässig. Alle zu einem Schlüsselwort gehörigen Parameter müssen in einer Zeile stehen.

Untenstehende Schlüsselwörter sind definiert und dürfen in beliebiger Reihenfolge und Häufigkeit verwendet werden. Der erste Befehl einer Installationsdatei muß aber immer ein M2DEVICE Befehl sein.

M2DEVICE	Anwahl einer Multi-LAB/2 Karte (optional mit Hardware-Reset der Karte und Laden eines Betriebssystems)
M2INST	Installieren eines Echtzeitprogrammes
M2PAR	Parameter einer Task setzen
M2PROC	Prozedur einer Task aufrufen
M2FUNC	Funktion einer Task aufrufen
M2CMD	Makrobefehl zur Multi-LAB/2 Karte senden

Alle Parameter sind hexadezimal anzugeben. Die maximale Länge ist in der Beschreibung jeweils angegeben. Führende Nullen können entfallen. In der folgenden Syntaxbeschreibung stehen eckige Klammern für optionale Angaben. Die kursiv gedruckten Bezeichner müssen durch Zahlenwerte oder Texte ersetzt werden.

M2DEVICE

Mit diesem Befehl wird eine Multi-LAB/2 angewählt, optional zurückgesetzt und optional ein Betriebssystem geladen. Alle darauffolgenden Befehle beziehen sich dann auf diese Karte - bis zum nächsten Befehl M2DEVICE, mit dem eine neue Karte gewählt wird. Wenn Sie die in MLC angewählte Karte eintragen wollen, dann können Sie im MLC-Editor die dafür nötige Befehlsfolge mit [Alt]+[D] an der Position des Cursors in die Installationsdatei einkopieren. Beachten Sie, daß im automatischen Eintrag von M2DEVICE immer RESET angegeben ist und gegebenenfalls gelöscht werden muß.

Syntax

M2DEVICE *adr* [TIMEOUT=*time*] [RESET *osname*]

adr: vierstellige Basisadresse (I/O-Adresse, hexadezimal), unter der die Karte im PC ansprechbar ist.

TIMEOUT: Stellt den Timeout-Wert für die Kommunikation zwischen Multi-LAB/2 Karte und PC ein. Wenn 'TIMEOUT' fehlt, wird ein Timeout von einer Sekunde (entspricht TIMEOUT=10) eingesetzt.

time: Größe des Timeout in Zehntelsekunden (*time*=10 entspricht 1 s, erlaubt sind 1 bis 100, dezimal)

RESET: Wenn das Schlüsselwort RESET angegeben wird, dann wird die Multi-LAB/2 Karte beim Ausführen (Laden) der Installationsdatei mit einem Hardware-Reset zurückgesetzt.

osname: Für einen RESET der Multi-LAB/2 muß ein Betriebssystem von Diskette oder Festplatte geladen werden. Dessen Name wird nach RESET angegeben. Pfadangaben sind zulässig.

M2INST

Dieser Befehl dient zum Installieren von Anwenderprogrammen.

Die Programmdatei des zu installierenden Programmes muß sich (sofern es sich nicht um ein ROM-Programm handelt) in einem der drei folgenden Verzeichnisse befinden:
- in dem Verzeichnis, von dem aus Sie MLC gestartet haben.
- in dem Verzeichnis, in dem die Installationsdatei steht.
- in dem Verzeichnis, in dem das Programm MLC steht.

Syntax:

M2INST *filename number task interrupt datasize flags*

filename: Dateiname (max. 8 Zeichen) und Extension (max. 4 Zeichen inkl. Punkt). Ein Suchpfad ist nicht zulässig. Die Datei muß sich in einem der oben genannten Verzeichnisse befinden.

number: Programm-Nummer (1 - ffffh, hexadezimal). Die Nummer muß mit der in der PDT eingetragenen Nummer des Programmes übereinstimmen (vierstellig, hexadezimal).

task: Tasknummer (1-3ffh, hexadezimal)

interrupt: Nummer des Interrupts, für den die Task installiert werden soll (bei NI-Tasks = 00, zweistellig, hexadezimal)

datasize: Länge des Datenbereichs der Task (sechsstellig, hexadezimal)

flags: Installierungsflags (achtstellig, hexadezimal)

Der Parameter *flags* spezifiziert Installierungsoptionen, die der nachfolgenden Tabelle zu entnehmen sind. Das Flag kann einfach gebildet werden, indem die aus der Spalte 'Wert' ermittelten Zahlen der gewünschten Optionen addiert werden.

Beachten Sie bitte, daß für 'EXE'-Dateien (z.B. von Ihnen erstellte Hochsprachenprogramme) in flags *das Programmformat "Exe not relocated" angegeben werden muß.* **!**

Die Parameter Tasktyp (in *flags*), *interrupt* und *datasize* werden in der Regel von den Echtzeitprogrammen mit Werten vorbesetzt (in der Programm-Deskriptor-Tabelle). Ein Echtzeitprogramm kann durch entsprechende Flags in der PDT erzwingen, daß diese Einstellungen verwendet werden, unabhängig von den bei **M2INST** übergebenen Werten. Dabei sind Tasktyp und Interrupt immer miteinander gekoppelt. Sofern es das Echtzeitprogramm zuläßt, kann der Installierungsbefehl beide Parameter einstellen (*flags* Bit-3=1) oder die vom Programm voreingestellten Werte

übernehmen (*flags* Bit-3=0). Die Größe des Datenbereichs kann der Installierungsbefehl - wiederum nur, wenn es das Echtzeitprogramm zuläßt - aus drei Angaben in der PDT auswählen (minimaler Datenbereich, maximaler Datenbereich und Datenbereichsgröße) oder mit *datasize* frei bestimmen (*flags* Bits 9 und 10).

Im folgenden finden Sie die Bedeutung der einzelnen Bits in *flags*. Den Gesamtwert des Flags ermitteln Sie einfach, indem Sie die entsprechenden Zahlen der Spalte 'Wert' addieren:

Bit	Wert	Bedeutung
2-0		Tasktyp-Festlegung, wenn Bit 3 des Flags der PDT =0 ist und Bit 3 des Parameters *flags* =1 ist
	0	(=000b): NI-Task (Nicht-Interrupt-Task)
	1	(=001b): II-Task (Indirekte Interrupt-Task)
	2	(=010b): DI-Task (Direkte Interrupt-Task)
3		Wer legt Tasktyp und Interrupt-Nummer fest?
	0	(=0b): PDT legt Tasktyp und Interrupt-Nummer fest
	8	(=1b): wenn Bit 3 des Flags der PDT Null ist, dann werden der Tasktyp und die Interrupt-Nummer durch die Parameter von **M2INST** festgelegt
5-4		Privilegstufe des Programmes
	0	(=00b): höchste Privilegstufe (Systemprogramme)
	10h	(=01b): zweithöchste Privilegstufe
	20h	(=10b): dritthöchste Privilegstufe
	30h	(=11b): niedrigste Privilegstufe (Anwendungsprogramme)
8-6		Programmformat

	flagbits	Wo?	Format	Adresse	Typ
0	(=000b):	RAM	PDT (tiny)	Anfang PDT	Assembler
100h	(=100b):	RAM	PDT (large)	Anfang PDT	Assembler
180h	(=110b):	RAM	Exe not reloc.	EXE-Header	C / Pascal
80h	(=010b):	RAM	EXE relocated	START_UP Code	Reserviert

Bit	Wert	Bedeutung		
10-9		Wie wird Größe des Datenbereichs festgelegt? Die Einstellung ist nur wirksam, wenn Bit 9 im PDT-Flag =0 ist.		
		flagbits	Größe richtet sich nach	fix/variabel
	600h	(=11b):	"Größe" in PDT	fix
	400h	(=10b):	"Minimum" in PDT	fix
	200h	(=01b):	"Maximum" in PDT	fix
	0	(=00b):	*datasize*	variabel
11		Auto-Init-Prozedur nach dem Installieren aufrufen?		
	0	(=0b): Auto-Init-Prozedur nicht aufrufen		
	800h	(=1b): Auto-Init-Prozedur aufrufen		
12		Task nach dem Installieren sofort aktivieren?		
	0	(=0b): Task nicht aktivieren		
	1000h	(=1b): Task aktivieren		
13-31		reserviert		

Installieren bedeutet nicht unbedingt, daß der Programmcode auf die Karte geladen wird. Wenn Sie in MLC unter 'Linken-Laden/Optionen' die Option 'Mehrfach-Installierung' eingeschaltet haben, prüft MLC zunächst, ob bereits eine Task mit der gleichen Programm-Nummer installiert ist. Wenn ja, wird der Code nicht auf die Karte geladen, sondern der Code der bereits installierten Task verwendet. Das Programm muß natürlich so geschrieben sein, daß es eine Mehrfachnutzung erlaubt. **!**

M2PAR

Mit diesem Befehl können ein oder mehrere Parameter einer Task gesetzt werden. Die Daten werden byteweise übergeben.

Syntax:

M2PAR *task start b1* [*b2*] [*b3*] ... [*bm*]

task: Nummer der Task, deren Parameter gesetzt werden sollen (vierstellig, hexadezimal)

start: Nummer des ersten Parameters, der gesetzt werden soll (vierstellig, hexadezimal)

b1: Wert, auf den der in *start* angegebene Parameter gesetzt werden soll (zweistellig, hexadezimal)

b2: Wert, auf den der nächste (*start*+1) Parameter gesetzt werden soll (zweistellig, hexadezimal)

b3: ...

bm: Wert, auf den der Parameter mit der Nummer *start*+m-1 gesetzt werden soll (zweistellig, hexadezimal)

M2PROC

Dieser Befehl ruft die Prozedur einer Task auf.

Syntax:

M2PROC *task procnr*

task: Nummer der Task, deren Prozedur aufgerufen werden soll (vierstellig, hexadezimal)

procnr: Nummer der Prozedur, die aufgerufen werden soll (vierstellig, hexadezimal)

M2FUNC

Dieser Befehl ruft die Funktion einer Task auf. Falls die Funktion eine Antwort zurückgibt, erscheint diese im Informationsfenster (in MLC die Echostufe auf 'jede Ausführung bestätigen' stellen).

Syntax:

M2FUNC *task funcnr* [*p1*] [*p2*] ... [*pn*]

task: Nummer der Task, deren Funktion aufgerufen werden soll (vierstellig, hexadezimal)

funcnr: Nummer der Funktion, die aufgerufen werden soll (vierstellig, hexadezimal)

p1..pn: Parameterbyte, die der Funktion übergeben werden (zweistellig, hexadezimal).

M2CMD

Mit diesem Befehl wird beim Laden ein Makrobefehl zur angewählten Karte gesendet. Falls die Multi-LAB/2 Karte eine Antwort auf den Makrobefehl zurückgibt, erscheint diese im Informationsfenster (in MLC die Echostufe auf 'jede Ausführung bestätigen' stellen).

Syntax:

M2CMD *code format p1* [*p2*] .. [*pn*]

code: Makrobefehls-Code (zweistellig, hexadezimal)

format: Formatbyte (siehe Kapitel 12, zweistellig, hexadezimal)

p1..pn: Parameter, des Makrobefehls (zweistellig, hexadezimal, Angabe byteweise).

Kommentartext

Nach einem Strichpunkt (;) oder einem Hochkomma (') kann beliebiger Text stehen. Er wird komplett (bis zum Ende der Zeile) ignoriert. Eventuell enthaltene Befehls-Schlüsselwörter werden natürlich auch ignoriert.