

Programmierbarer Taktgeber

MCT Prüfungsvorbereitung: Entprellen

Version 2, © 2005/10 by B.Wamister

Funktionsweise des Taktgebers:

Für eine, mit einer MCU gesteuerte Werkzeugmaschine muss ein Taktgeber für die Verarbeitungsgeschwindigkeit programmiert werden.

Der Taktgeber muss ein Signal von 1Hz bis 7Hz ausgeben können. Diese Frequenz soll in 1Hz Schritten durch die 2 Tasten (Up und Down) eingestellt werden können. Die vorgewählte Frequenz wird im Dual-Code durch die 3 LEDs (V0,V1, V2) angezeigt.









Durch einen Impuls der Taste Go wird die vorgewählte Frequenz sofort übernommen (positive Flanke) und die LED Takt blinkt mit der voreingestellten Frequenz.

Wird die Taste Stop betätigt, wird das System unmittelbar angehalten, und das Ausgangssignal Takt auf 0 gesetzt. Die Vorwahl sowie die Vorwahlanzeige (V0, V1, V2) bleiben erhalten und können auch bei aktivem Stop eingestellt werden. Nach einem Stop wird mit der Taste Go die Takt Ausgabe gestartet. Dabei hat jedoch die Taste Stop höhere Priorität.

Nach einem Systemreset ist der Taktgeber im Stop-Modus, die Vorwahl auf 1Hz und die Takt Ausgabe kann mit der Taste Go gestartet werden.

Dauernd gedrückte Tasten oder Signale an nicht verwendeten Porteingängen dürfen den Taktgeber nicht blockieren.

Bedienungselemente:

Taster	LEDs
 Up	 V0
 Down	 V1
 Go	 V1
 Stop	 Up

Weitere Anforderungen

Genauigkeit Periodendauer Takt	+/- 5ms
Tastegrad Takt	0.5 (+/- 5ms)
Minimale Dauer Tastendruck	40ms
Maximale Dauer Tastendruck	beliebig
Prellzeit der Tasten	< 5ms

Programmierung:

Für allgemeine Anwendungen sollen folgende Funktionen entwickelt werden:

Ein einzelnes Bit am PortX auf 0 oder 1 setzen, ohne dass die anderen Bits ändern:

```
void BitWrite_PX( char Zustand, char BitNr );
```

Zustand: 1 oder 0; BitNr: 0 bis 7

Ein einzelnes Bit am PortX entprellen, ohne dass die übrigen Bits einen Einfluss haben:

```
int BitEntprell_PX( char BitNr );
```

BitNr: 0 bis 7

Return: Anzahl Male aufgerufen mit gedr. Taste

Lösungshinweis:

Dieses Problem wird am einfachsten gelöst, indem die Endlosschleife des Programms in regelmässigen Zeitabständen ausgeführt wird. Bei professionellen Anwendungen wird dazu ein Interrupt benutzt. Sie können in Ihre Endlosschleife einen Delay einbauen. Dieser Delay sollte wesentlich länger sein als die Programmlaufzeit jedoch mindestens so kurz wie das kleinste benötigte Zeitintervall. Durch das Zählen der Anzahl Durchläufe können Sie so beliebige Zeitintervalle generieren.