

Interrupt

Interrupts sind Unterbrechungen der normalen Programmroutine, um in kritischen Situationen sofort reagieren zu können. Beim Auftreten eines Interrupts verlässt der µC den normalen Programmablauf und führt eine vorher geschriebene Programmroutine aus. Danach kehrt er wieder in den normalen Programmablauf zurück ohne das irgendwelche Befehle übersprungen werden.

Was löst einen Interrupt aus?

Die Liste der Interrupteinsprungsadressen:

Quelle		Bit	Interruptadresse Einsprung	
				INT-Nr. in der Sprache C :
Externer INT0	Port 3.2	IE0	0003h	Interrupt 0
Zähler 0 Überlauf Takt	Port 3.4.	TF0	000Bh	Interrupt 1
Externer INT1	Port 3.3	IE1	0013h	Interrupt 2
Zähler 1 Überlauf Takt	Port 3.5.	TF1	001Bh	Interrupt 3
Serieller Port		RI oder TI	0023h	Interrupt 4
Zähler 2 oder externer INT2 (8032)		TF2 oder EXF2	002Bh	Interrupt 5

Es gibt verschiedene Interrupt Quellen

- Externer Interrupt 0 (Port 3_2)
- Externer Interrupt 1 (Port 3_3)
- Überlauf Zähler 0
- Überlauf Zähler 1
- Überlauf Zähler 2
- Serieller Port

Welche Bedingung muss erfüllt sein?

Es muss festgelegt werden bei welchem Zustand der Interrupt auslösen soll.

- Low Pegel
- High Pegel
- Negative Flanke
- Postive Flanke



Wo steht meine Interrupt – Routine?

Jeder Interrupt hat seine feste Einsprungsadresse zu welcher der Controller automatisch beim Auftreten des Interrupts hin springt. Da zwischen diesen Adressen zu wenig Platz für eine Programmroutine ist steht dort meist ein Sprungbefehl an eine andere Stelle im Codespeicher. Über den Befehl „reti“ beendet man dann die Interruptroutine und kehrt wieder ins Hauptprogramm zurück.

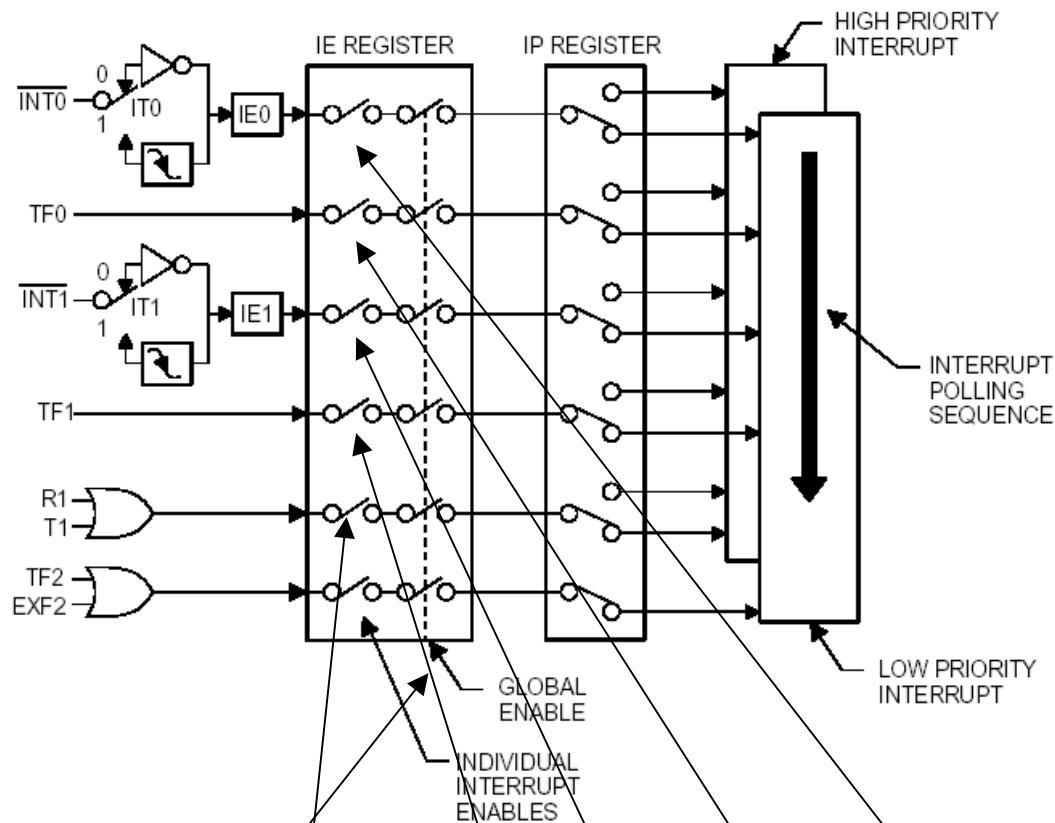
Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{\text{INT0}}$ (external interrupt 0)
P3.3	$\overline{\text{INT1}}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{\text{WR}}$ (external data memory write strobe)
P3.7	$\overline{\text{RD}}$ (external data memory read strobe)



Mit den Portpins P3.2 und P3.3, an die auch jeweils ein Taster angeschlossen ist, können externe Interrupts ausgelöst werden.

Die Interrupts werden nur ausgelöst, wenn sie durch Setzen der Bits EX0 bzw. EX1 freigegeben sind.

Wählen Sie unbedingt die Flankensteuerung, da sonst solange Interrupts ausgelöst werden, wie an den Pins Low-Signal anliegt.



Wie steuert man die Interrupts?

Interrupt Enable Register (IE) Bit adressierbar

Adresse: A8

Die Steuerung der Interrupts erfolgt über die Special Funktion Register IE, IP, TCON.

Aufbau:

EA	-	ET2	ES	ET1	EX1	ET0	EX0
----	---	-----	----	-----	-----	-----	-----

- EA Setzt man dieses Bit schaltet man damit Interrupts generell erstmal frei
- Reserve
- ET2 Setzt man dieses Bit wird beim Überlauf des Timer 2 ein Interrupt ausgelöst
- ES Setzt man dieses Bit wird der Interrupt der seriellen Schnittstelle frei geschaltet
- ET1 Setzt man dieses Bit wird beim Überlauf des Timer 1 ein Interrupt ausgelöst
- EX1 Setzt man dieses Bit schaltet man den externen Interrupt am Port 3_3 frei
- ET0 Setzt man dieses Bit wird beim Überlauf des Timer 0 ein Interrupt ausgelöst
- EX0 Setzt man dieses Bit schaltet man den externen Interrupt am Port 3_2 frei

Interrupt Priority Register (IP) Bit adressierbar

Adresse: B8

Aufbau:

-	-	PT2	PS	PT1	PX1	PT0	PX0
---	---	-----	----	-----	-----	-----	-----

-	Reserve
-	Reserve
PT2	Setzt man dieses Bit hat Zähler 2 Interrupt höhere Priorität
PS	Setzt man dieses Bit hat der Interrupt des seriellen Ports höhere Priorität
PT1	Setzt man dieses Bit hat Zähler 1 Interrupt höhere Priorität
PX1	Setzt man dieses Bit hat der externe Interrupt 1 höhere Priorität
PT0	Setzt man dieses Bit hat Zähler 0 Interrupt höhere Priorität
PX0	Setzt man dieses Bit hat externe Interrupt 0 höhere Priorität

Hinweis:

Treten Interrupts mit gleicher Priorität gleichzeitig auf werden sie nach der folgenden Reihenfolge abgearbeitet.

- Externer Interrupt 0
- Überlauf Zähler 0
- Externer Interrupt 1
- Überlauf Zähler 1
- Serieller Port
- Überlauf Zähler 2

Timer Control Register (TCON) Bit adressierbar

Adresse: 88

Aufbau:

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-----	-----	-----	-----	-----	-----	-----	-----

IE1	Wird automatisch vom Controller gesetzt wenn der externe Interrupt 1 auftritt (Hardwarelöschung/Softwarelöschung).
IT1	Dieses Bit entscheidet ob der externe Interrupt 1 bei einer negativen Flanke (1) oder einem negativen Pegel (0) ausgelöst werden soll.
IE0	Wird automatisch vom Controller gesetzt wenn der externe Interrupt 0 auftritt (Hardwarelöschung/Softwarelöschung).
IT0	Dieses Bit entscheidet ob der externe Interrupt 0 bei einer negativen Flanke (1) oder einem negativen Pegel (0) ausgelöst werden soll.

Bei den externen Interrupts 0/1 wird das entsprechende Erkennungsbit nur dann vom Controller wieder hardwaremäßig zurückgesetzt wenn auf eine negative Flanke abgefragt wurde. Wurde auf einen negativen Pegel abgefragt so muss die Löschung des Bits softwaremäßig erfolgen.

Hardwarelöschung:

Der Controller löscht das Bit nach dem er die Interruptroutine abgearbeitet hat.

Softwarelöschung:

Der Programmierer muss daran denken das Bit über einen Befehl zurück zu setzen.