

# Interrupts

# Inhaltsangabe

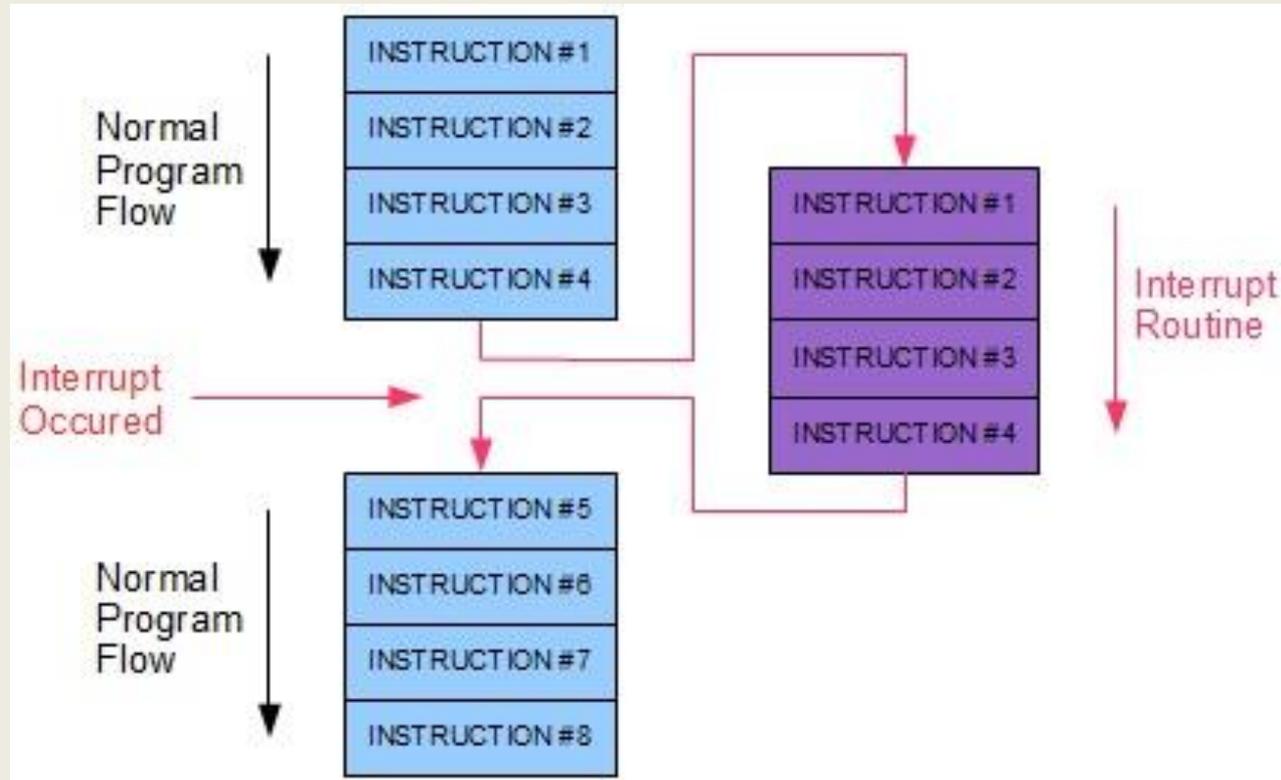
1. Einführung in Interrupts
2. Detektierung
3. Verarbeitung
4. Synthese

# Einführung in Interrupts

# Was ist ein Interrupt?

- Unterbrechung des Hauptprogramms
- Ausführung eines Unterprogramms
- Rücksprung zum Hauptprogramm

# Schema



Detektierung

# Auslösen von Interrupts:

- Änderung eines Zustandes an einem Pin
- Ablauf einer Zeitspanne
- Nach einer seriellen Übertragung
- Eine Messung eines AD-Wandlers abgeschlossen ist

# Registrierung eines Interrupts

- Voraussetzung: Globale Aktivierung (sei(),cli())
- Statusbit gesetzt (Interruptflag)
- Interrupt wird abgearbeitet

Verarbeitung

# Verarbeitung:

- Interruptflag wird gelöscht
- Interrupt Service Routine wird aufgerufen (ISR)
- Rücksprung zum Hauptprogramm

# Wichtig bei Verarbeitung:

- Jedes Interrupt braucht ein Programm
- Interrupts sind über Funktionsbausteine zu schalten
- Globalen Schalter betätigen

# Wichtig bei ISR:

- Kleine Rechnungen
- Keine langen While-Schleifen
- Vermeidung von Interrupt Verkettungen

Synthese

# Vorteile:

- Multitasking
- Zeitsparend
- Nachteil: Fehleranfällig beim Programmieren

# Quellen

<http://microautomate.com/uawordpress/wp-content/uploads/2013/12/interrupt.jpg>

<http://www.mikrocontroller.net/articles/AVR-GCC-Tutorial>

<http://www.mikrocontroller.net/articles/Interrupt>