# Interrupt-Einstellungen

* Handler liegen in startup\_ccs.c. Dort werden sie auf aufgesetzt: GPIO-Hanlder startup\_ccs.c
* RTC-Handler liegt (noch) in rtc.c
* initRadioInts() (noch) in radio.c

# Konstanten

* Alle IO-Konstanten (wie BOARD\_IO\_DP0) board.h
* Alle GPIO-Konstanten (wie GPIO\_PIN\_29) gpio.h
* Applikations-spezififische Konstanten (WAKE\_INTERVAL, ADVLEN) config.h
* PowerDomains prc.h

# Allgemeine Einstellungen

**ioc.h** Adressen aller Peripherals (MCU, GPIO, AON, RF)

Ioc.c **IO-Controller.** Vebindet (map) Peripherals (GPIO, MCU, AON, RF) mit logischen IOs.Greift auf CFG zurück. Jedes Peripheral hat eine eigene PORTID. Verbunden wird über Peripheral-Funktionen  
 *IOCPortConfigureSet(DIOn, PORTID, PIN-CONFIG);*  Pinconfig ist das Register diese Pins. Dazu werden Defines gebraucht.

**sys\_ctrl.h**

* Buseinstellunge
* CPU Sleep mode   
  **#define** CPU\_RUN 0x00000000

**#define** CPU\_SLEEP 0x00000001

**#define** CPU\_DEEP\_SLEEP 0x00000002

* Refresh-Zyklus (SysCtrlSetRecharge..)

**rtc.c**

* initRTC() von Dario ev. In system.h  
  Aufsetzen des RTC, continuous mode, weckt nach 10 s MCU
* initRTC\_WUms(1000) PA  
  Aufsetzen RTC, normalCompare mode, wckt nach 10 s MCU ev. in system.h

**radio.h**

# Funktionen

* PowerEnable(), powerDisable(), wait(), **init()** system.h
* Radio-Funktionen (**init()**, run, bus,start(), updateData() radio.h
* RTC-Funktionen (**init()**, calculation) rtc.h