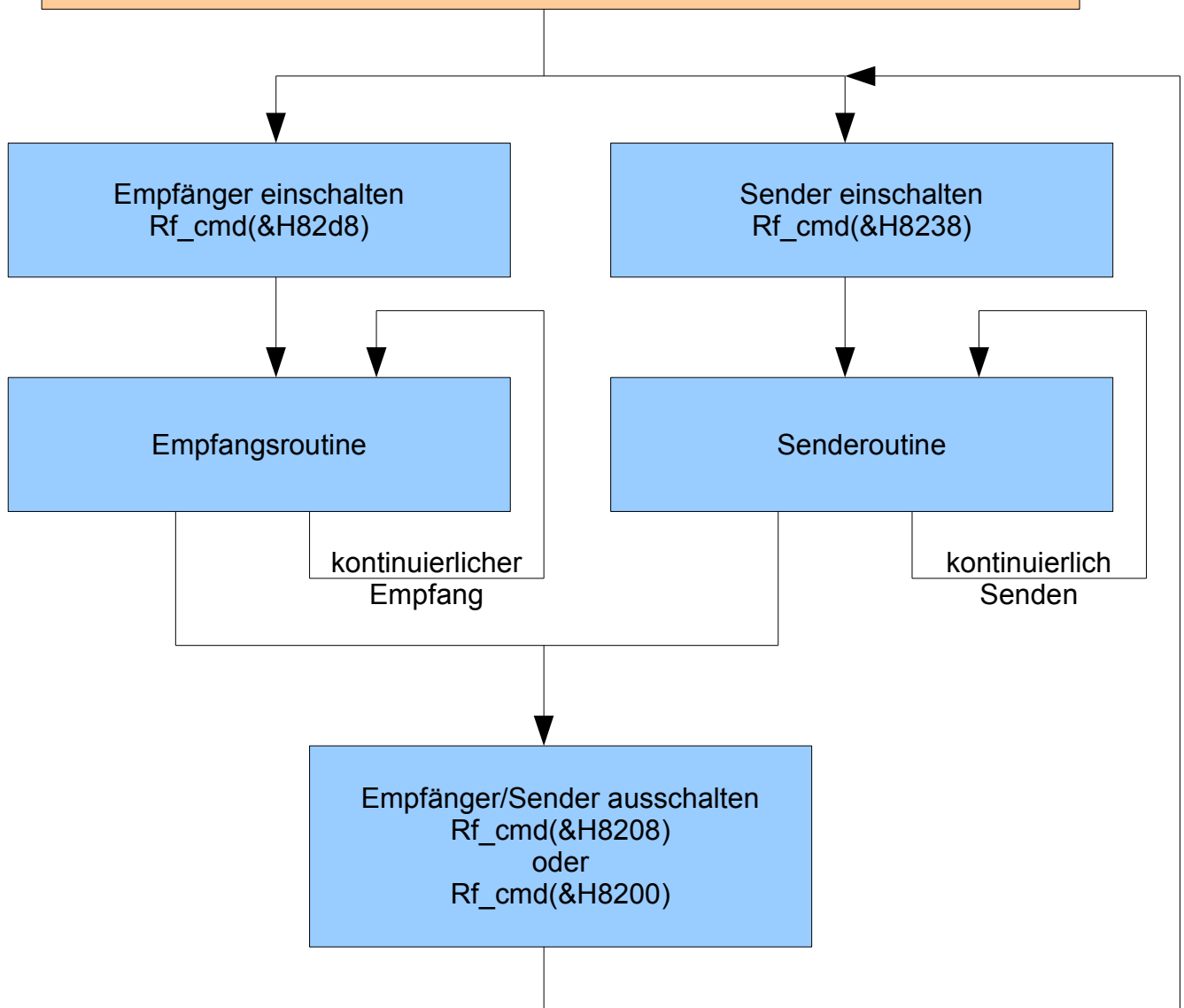
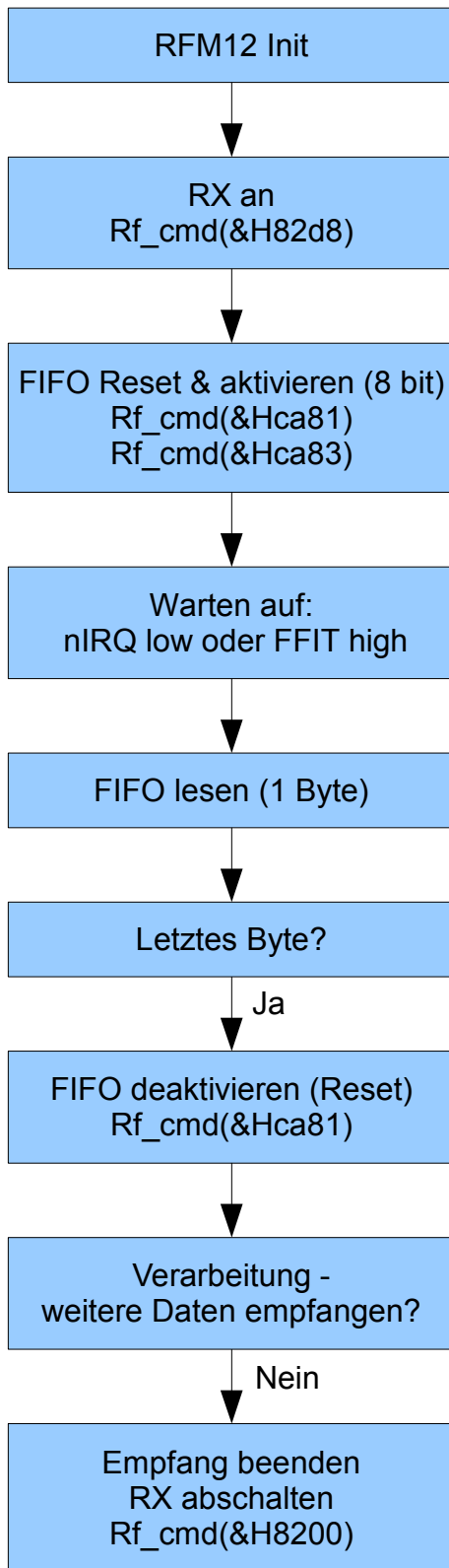


RF 12 Init für 868 Mhz Band und TX/RX FIFO

&Hc0e0 'low battery 2,2V, clock output 10 MHz
&H80e7 'Configuration: 868MHzband, 12pf, enable TX & RX FIFO
&H8208 'power management: er, ebb, et, es, eb, ew aus; ex an
&Hc2ac 'data filter command
&Ha686 'Frequency: 868,35MHz
&Hc611 'Datarate: 19,2 kbit
&H94a1 'receiver setting: bw 134kHz, RSSI -97dbm, LNA 0 dB, VDI fast
&Hc483 'AFC:
&H9850 'TX control: 0dB output power, 90 kHz
&He000 'wake-up: 0 ms
&Hc800 'low duty-cycle: none
&H0000 'read status, clear IRQ's



Empfangsroutine



&H82d8:
Enable: receiver chain, baseband,
Synthesizer, crystal oscillator

FIFO Reset Prozedur:
'FIFO fill' deaktivieren & aktivieren,
Danach wartet FIFO auf sync word H2DD4

Nach Empfang des letzten Bytes sofort FIFO
ausschalten, sonst wird weiter Datenmüll
empfangen und FIFO-IRQ erneut ausgelöst

Daten müssen kontinuierlich (ohne Pause)
gesendet werden, am Ende eines Blocks
FIFO Reset ausführen.
FIFO wartet wieder auf Sync word

FIFO lesen:

mit „Status read command“: config SPI „Spiin = 0“ (default)
„Spiin Fifo(1) , 3“ – es werden 3 Byte (00) an den RFM gesendet, RFM sendet 3 Byte zurück,
Bytes 1 & 2 enthalten Statusbits, Byte 3 ist das Datenbyte vom FIFO

mit „FIFO read command“: config SPI „Spiin = &HB0“
„Spiin Fifo(1) , 2“ - 2 Byte (hB0) werden an RFM gesendet, RFM sendet 2 Byte zurück,
Byte 2 ist das Datenbyte

Senderoutine

RFM12 Init

TX an:
Rf_cmd(&H8238)

&H8238:
Enable: PLL, PA, TX, Synthesizer, crystal Osc.

Sende Sync Bytes (h2DD4)
Sdi_send(&H2d)
Sdi_send(&Hd4)

Preamble (hAA) muss am Anfang nicht
gesendet werden, da im TX Register schon
vorhanden (siehe RF12.pdf)

Sende Daten (n * Byte)
Sdi_send(Byte)

Sende 2 Dummy Bytes
Sdi_send(&Haa)
Sdi_send(&Haa)

Am Ende 2 Dummy Bytes senden, damit die
letzten 2 Datenbytes aus dem TX Register
gesendet werden.

TX aus
Rf_cmd(&H8200)
Oder
Rf_cmd(&H8208)

&H8200:
disable: PLL, PA, TX, Synthesizer, crystal Osc.
&H8208:
disable: PLL, PA, TX, Synthesizer

SDI-send(xx):

- wartet auf TX FIFO IRQ
- sendet „Txregister write command“ (hB8..) und
ein Byte an den RFM