# Purpose:

Quick technical description of the HC30 PCBA used in HC30RF handsets.

# Overview:

The PCBA is divided into the following sub circuits:

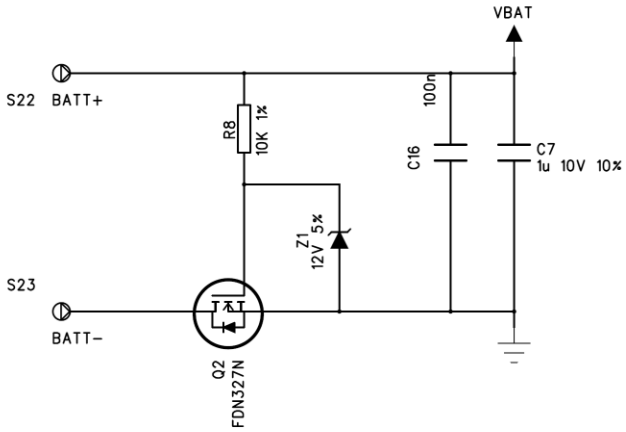
1) Battery connection
2) Micro controller
3) RF transmitter + antenna
4) Key-matrix
5) Programming interface
6) Backlight

# Ref:

- Circuit diagram for HC30 board – 10907584-B.
- Block Diagram

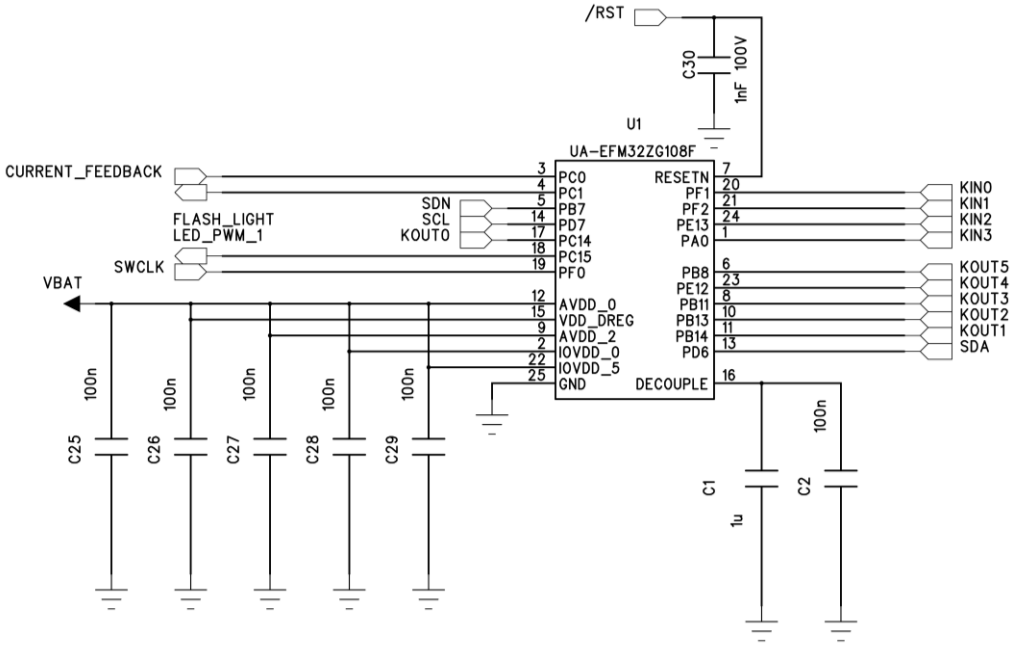
# Sub circuit description

## *Battery connection*



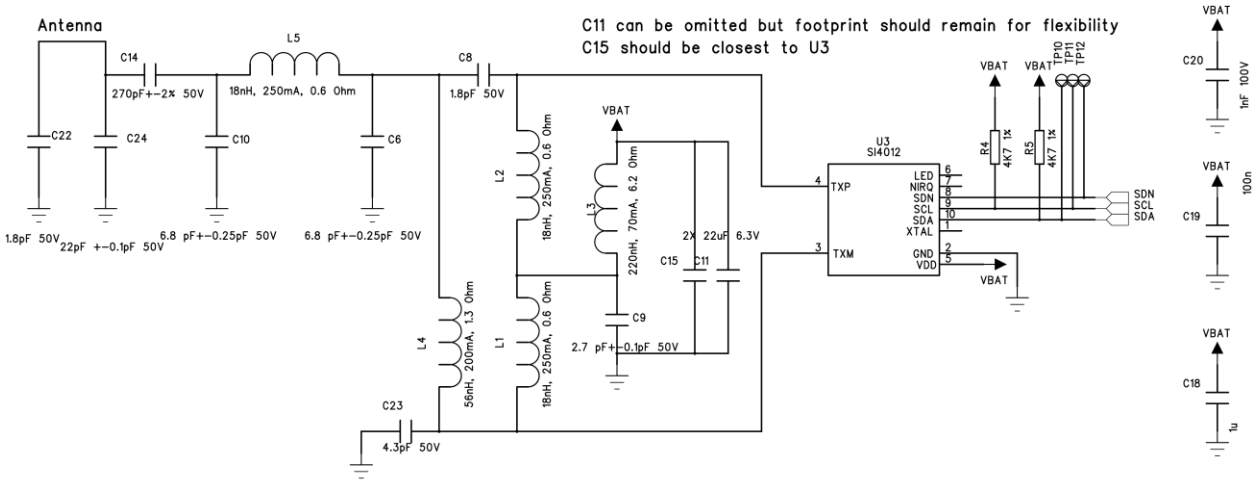

2 x AAA (LR03) batteries are connected in series to S24(+) and S19(-). Q2 and R8 work as a protection against wrong polarization. Z1 protect the gate of Q2 against voltage pulses above 12V (ESD)

## *Micro controller*




The uC handles all key scan, Backlight and LED operation and of cause setup and data stream to the RF transmitter.
When the handset is not used, the microcontroller (uC) enters a low-power mode where all external clock frequencies are switched off. It wakes up and scans the key matrix to see if a button have been pressed and then power down again.
If a button is pressed down, the uC wakes up and start communicating with the RF transmitter on the I2C bus. It starts by initializing the transmitter and afterwards it start transmitting the actual data packages to the buffer of the RF transmitter.

## *RF transmitter+antenna*

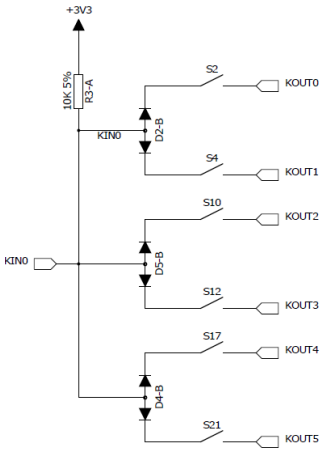



The RF transmitter is actually a SI4010 but preloaded with a specific supplier software and therefor named SI4012.

The modulation scheme is the simplest form of ASK modulation called OOK. The data bit rate is 4 kbit/s (each bit having a length of
250 us). The coding scheme is Manchester so each bit consist of a high period of 125 us and a low period of 125 us. Thus the signal will always have a duty cycle of 50% (constant DC component) regardless of the data content. Data is transmitted in packets.
For a more detailed description of the coding and data scheme refer to the “RF-protocol description”.

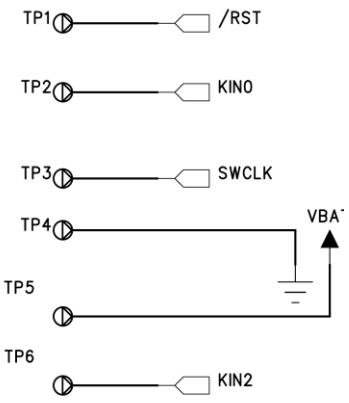
When a button is pressed and the micro controller wakes up from sleep state it starts setting up the RF transmitter. First it send the setup package telling the transmitter the transmitting frequency (433.92MHz), transmit power, modulation type, to use internal oscillator, how the impedance network on the output is configured ect. Second it send a “clear FIFO” command to clear the internal transmit buffer.
When the setup is complete, the microcontroller send the actual data package according to the RF-protocol.

## *Key-matrix*

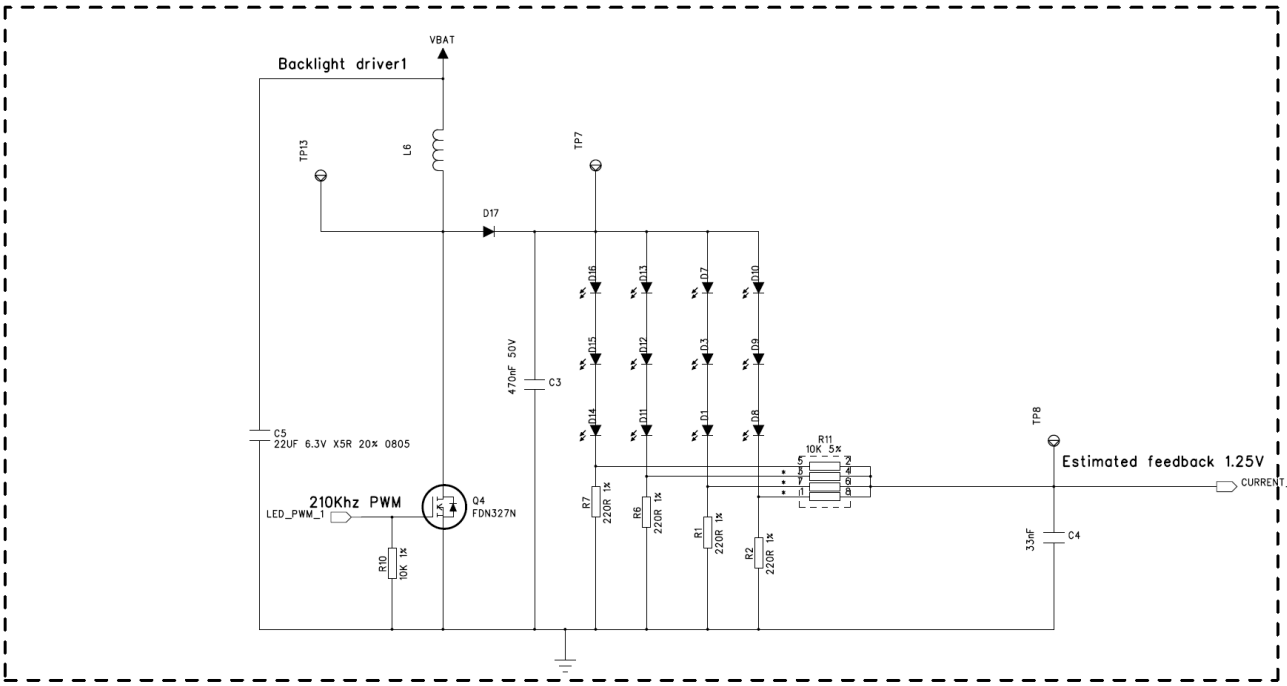



The key-matrix (example above) detects if a key is pressed. The uC has KOUTx set to 0. In turn it threstates KINx and listen. KINx is per default “HIGH” (above 0.7xVcc). If KINx is detected as being less than 0.3xVcc it measures KOUTx and thus detects which key is pressed. This information tells the software which data package to transmit.

## *Programming interface*



## *Backlight*




The backlight is controlled by the microcontroller. The voltage is boosted by pulse wide modulateing the MOSFET Q4 running about 210Khz. The current flow through the LED´s are adjusted by the summation of the current over all 4 strings measured over the 4 resistors R1, R2, R6, R7. The summation feedback is compared in the microcontroller with a comparator at 1,25V. If the voltage is below, the pulse wide modulation will increased, if above the pulse wide modulation will decreased. For safety, it is not possible to use pulse wide modulation above 90%. D17 is a schottky diode that together with C3 will stabilize the voltage for more stable feedback measurement. For brightness adjustment there is a 100hz pulse wide modulation on top of the current regulation PWM, this allow adjustment of the brightness from 0-100% with 10% steps.