

Programmes After Market Services
NPC-1 Series Transceivers

Troubleshooting Instructions

Table of Contents

	Page No
Troubleshooting Instructions	1
Transceiver Troubleshooting	4
Baseband Troubleshooting	4
General about testing.....	4
Troubleshooting	5
Power up faults.....	6
Flash Programming does not work.....	10
Charging.....	12
Audio failures.....	13
Accessory detection.....	15
Sleep Clock	16
Display/Keyboard lights do not work	18
Display does not work.....	19
Buzzer does not work.....	20
Vibra does not work	20
Keyboard does not work	20
Infrared does not work	21
Receiver Troubleshooting	22
General instructions for RX troubleshooting	22
Path of the received signal	22
Fault finding charts for receiver chain	22
Transmitter Troubleshooting	26
General instructions for TX troubleshooting	26
Path of the transmitted signall.....	26
Fault finding charts for the transmitter.....	26
Synthesizer Troubleshooting	30
19.44 MHz reference oscillator.....	30
RX VHF VCO.....	31
TX VHF VCO	32
UHF SYNTHESIZER	33

List of Figures

	Page No
Fig 1 Power up sequence	7
Fig 2 Fault tree, phone does not power up.....	9
Fig 3 Fault tree, charging	13
Fig 4 Fault tree, earpiece fail	14
Fig 5 Fault tree, microphone fail	15
Fig 6 Fault tree, accessory detection.....	16
Fig 7 Fault tree, sleep clock	17
Fig 8 Fault tree, Display/Keyboard lights.....	18
Fig 9 Fault tree, Display does not work.....	19
Fig 10 Fault tree, infrared	21
Fig 11 Fault tree. AMPS	23
Fig 12 Fault tree. Transmitter troubleshooting	27
Fig 13 Fault tree: Power control loop	29
Fig 14 Detected voltages.....	30
Fig 15 Fault tree: 19.44 MHz oscillator.....	31
Fig 16 Fault tree: RX VHF VCO.....	32
Fig 17 Fault tree: TX VHF VCO	33
Fig 18 Fault tree: UHF Synthesizer	34

Transceiver Troubleshooting

The Troubleshooting instructions are divided into following sections:

- 1 How to check/fix the system/sleep clock.
- 2 How to check/fix the power supplies.
- 3 Contact service case.
- 4 How to check/fix the SIM faults.
- 5 How to check/fix the Audio faults.
- 6 How to check/fix the charger faults.

The first thing to do is to carry out a thorough visual check of the module. Make sure that:

- there are no mechanical damages
- the solder joints are OK

Note: Before changing anything, ALL SUPPLY VOLTAGES AND THE SYSTEM CLOCK / SLEEP CLOCK should be checked.

Baseband Troubleshooting

General about testing

The phone has three different modes for testing and/or repairing the phone. The modes can be selected with suitable resistors connected to BSI- and BTEMP- lines as illustrated in the table below:

Table 1: Phone operating mode

Mode	BSI-resistor	BTEMP resistor	Remarks
Normal	68k	47k	
Local	560_ (<1k_)	Whatever	
Test	> 1k	560_ (<1k_)	Recommended for baseband testing. Same as local mode, but making a phone call is possible.

If the corresponding resistors are connected, the MCU software automatically enters into the local or test mode when the supply voltage is connected to the phone.

The power can be switched on by:

- 1 Pressing the power key.
- 2 Connecting the local/test mode resistors to the bsi/btemp lines and connecting the battery voltage to the phone.
- 3 Connecting a charger.
- 4 Phone's internal functions (Real time clock alarm)

In the local and test mode, the baseband can be controlled through MBUS or FBUS (FBUS is recommended) connections using Phoenix service software.

Troubleshooting

Note: Most of the baseband repair actions require removing the baseband shield. The shield must not be removed unless authorized by PAMS.

Note: ESN has to be re-written onto the phone, if the UEM is replaced. This can be done only in Central Service.

The BB troubleshooting instructions consist of the following topics:

- 1 Power up
- 2 Flash programming
- 3 Audio
 - Earpiece
 - Microphone
- 4 Charging
- 5 Accessory detection
- 6 Sleep clock
- 7 User interface
 - Display
 - Lights
 - Buzzer
 - Vibra
 - Keyboard

8 Infrared

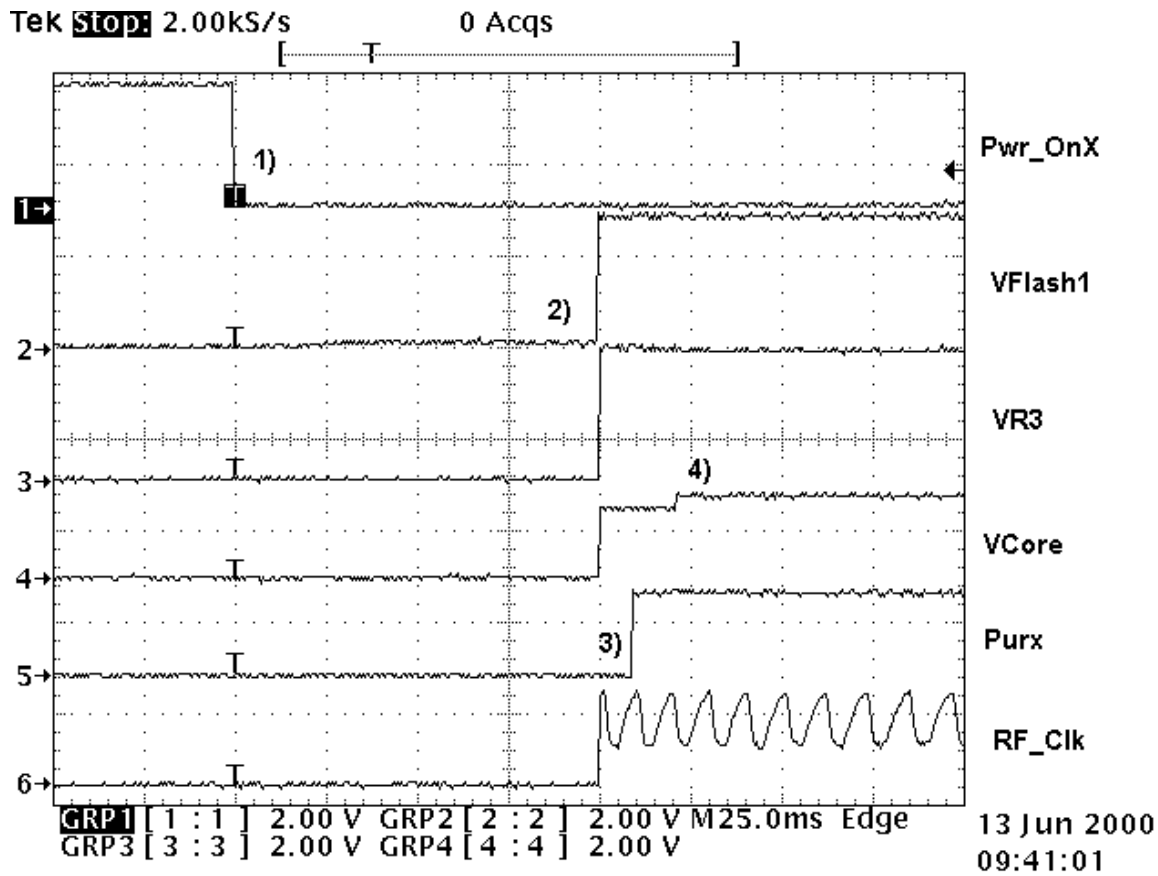
Power up faults

Power up sequence

- 1 The UEM acts as a HW master during start up
 - Vbatt limits: 2.1V for internal state machine, 3V triggering whole startup
 - Regulator sequencing
 - HW "core" regulators "on": Vio, Vcore, VR3, Vflash1
 - These regulators supply the processors, memory, chip interfaces and clock source in RF
 - Reset releasing delay
 - Supply voltages stabilize to their UEM HW default values
 - RFCLK grows to full swing
 - The core is ready to run but waiting for the PURX release
 - Reset releasing
 - The UPP releases the SLEEPX up to the "non sleep" -state to prevent the UEM switching the regulators "OFF"
- 2 MCU starts running the Bootstrap Code
 - written in stone/ UPP internal ROM
 - the program checks if there is any reason for the FDL mode (Flash Down Load)
 - If there is an executable code in FLASH and there is no reason for FDL, the MCU starts running the MCU program from FLASH.
- 3 MCU runs the FLASH MCU code
 - the phone initialization, user interfaces, internal blocks etc.
 - Core regulator voltage setting for required DSP speed
 - Initializes the DSP and concerning HW
 - Releases DSP reset -> DSP starts running

Note: In figure below, the RF_Clk frequency appears to be lower than 19.44MHz because of a too low oscilloscope sampling frequency (2kS/s).

Figure 1: Power up sequence



1 Power key pressed

- After 20ms UEM enters RESET MODE if VBAT>Vmstr+
- VFLASH1, Vana, Vcore, Vio and VR3 goes high.
- VCTCXO enabled by VR3 -> RFCIk 19.44 MHz running.

2 Purx released

- Purx released by UEM, UEMINT goes high for 100 ms, SleepX goes high and UEM starts feeding SleepClk (32 KHz) to UPP.

Note: The sleepclock starts running in the crystal circuitry (B200) immediately when the battery voltage is supplied.

3 Software running

- Default value for Vcore is 1.5 volts and, if the software is running, Vcore will rise to 1.8 volts.
- Cbus (1.08MHz) clock starts running.

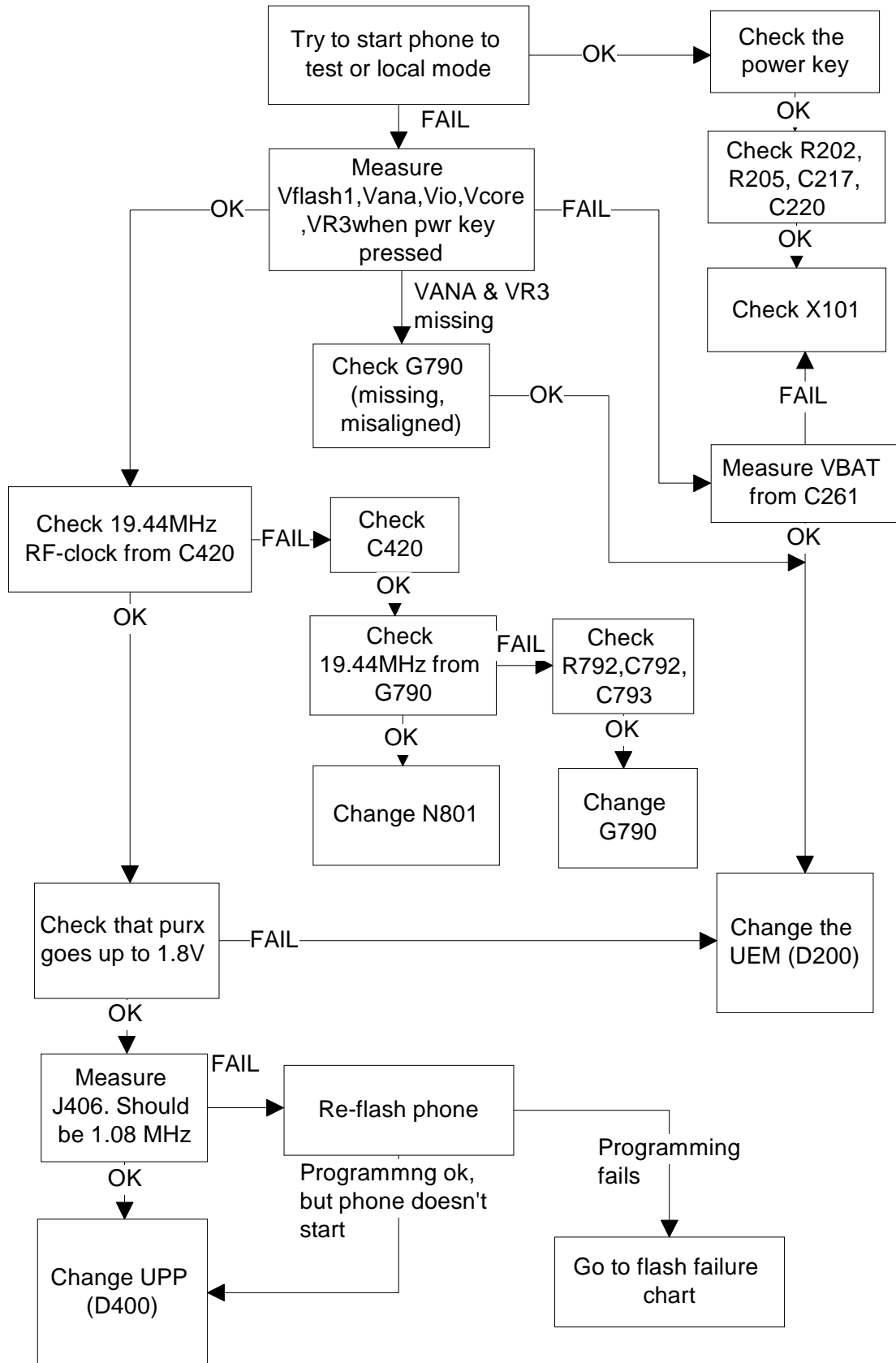
Phone does not start up normally or does not stay on

Note: In case of power up faults, it is not possible to force the phone on by disabling the watchdog. Instead, measurements should be taken immediately when the power key is pressed or when the battery voltage is connected to the phone (local/test mode).

The easiest way to check if the software is running when the phone takes an abnormal amount of current is to measure the *cbusclk* and *Vcore*.

The Dbus clock (programmable 9.72MHz) is not automatically visible in the test and local modes.

Figure 2: Fault tree, phone does not power up



Flash Programming does not work

Flash programming procedure

- 1 The phone communicates with the prommer via the production test pattern, using the following signals:
 - FBUSTX(serial data **to** phone)
 - FBUSRX(serial data **from** phone)
 - MBUS(serial clock for FBUSRX)
 - VPP(External flashing voltage for speed up flashing)
 - The BSI line is also used when initializing flashing (battery connector).
- 2 When the phone is powered ($V_{BAT} > 3V$), the MBUS and FBUSTX lines are pulled up internally by the phone.
- 3 The prommer sends a command to the UEM, using FBUSRX, to enter the Flash-mode. During the sending of this command, the prommer keeps the BSI line high and MBUS is used as a serial clock.
- 4 When the Flash-mode command is acknowledged, **UEM** enters the Flash-mode and releases reset (PURX) to MCU.
- 5 After reset is released, UPP checks if there is a request for the Bootstrap code (that resides in the UPP ROM).
- 6 The request for Bootstrap is the MBUS pulled down by the prommer (if the bootstrap is not requested, the bootstrap code jumps to FLASH SW).
- 7 If the Bootstrap code is requested, **UPP** enters the Flash-mode and sets FbustX to '0' as an acknowledgement to the prommer. This is an indication that UPP can run, at least, the fixed Bootstrap code – although it is not able to run the FLASH code. UPP then sends an UPP-ID to the prommer via the FBUSTX line.
- 8 After the prommer has received the UPP-ID, it sends a corresponding Secondary Boot Code to the phone via FBUSRX. The Secondary Boot Code, when run in UPP, requests UPP to send information to the prommer about the flash type and other HW-related parameters about the device to be flashed.
- 9 Prommer then sends the Algorithm Code corresponding to the HW parameters, and this algorithm, when run in UPP, takes over handling the MCUSW transfer to Flash.
- 10 12 volts can be supplied to Vpp (by the prommer) to speedup flashing.

- 11 The FLASH Program includes a package of MCU and DSP software and all default parameters for the phone. The tuning values will be added/rewritten during the Flash/Alignment phase.

Flash programming error codes

The various error codes can be seen from the "FPS-8 Flash" in Phoenix.

The underlined text in the table below means that the item under consideration is being used for the first time in the flashing sequence.

Table 2: Table 2. Flash programming error codes.

Error	Description	Not working properly
C101	"The Phone does not set FbusTx line high after the startup."	<u>Vbatt</u> <u>Vflash1</u> <u>Vcore</u> <u>VIO</u> <u>BSI and FbusRX from prommer to UEM.</u> <u>FbusTx from UPP->UEM->Prommer(SA0)</u>
C102	"The Phone does not set FbusTx line low after the line has been high. The Prommer generates this error also when the Phone is not connected to the Prommer."	<u>PURX(also to Safari)</u> <u>VR3</u> <u>Rfclock(VCTCXO->Safari->UPP)</u> <u>Mbus from Prommer->UEM->UPP(MbusRx)(SA1)</u> <u>FbusTx from UPP->UEM->Prommer(SA1)</u>
C103	"Boot serial line fail."	<u>Mbus from Prommer->UEM->UPP(MbusRx)(SA0)</u> <u>FbusRx from Prommer->UEM->UPP</u> <u>FbusTx from UPP->UEM->Prommer</u>
C104	"MCU ID message sending failed in the Phone."	<u>FbusTx from UPP->UEM->Prommer</u>
C105	"The Phone has not received Secondary boot codes length bytes correctly."	<u>Mbus from Prommer->UEM->UPP(MbusRx)</u> <u>FbusRx from Prommer->UEM->UPP</u> <u>FbusTx from UPP->UEM->Prommer</u>
C106	"The Phone has not received Secondary code bytes correctly."	<u>Mbus from Prommer->UEM->UPP(MbusRx)</u> <u>FbusRx from Prommer->UEM->UPP</u> <u>FbusTx from UPP->UEM->Prommer</u>
C107	"The Phone MCU can not start Secondary code correctly."	<u>UPP</u>
A204	" The flash manufacturer and device IDs in the existing Algorithm files do not match with the IDs received from the target phone."	<u>Flash</u> <u>Signals between UPP-Flash</u>
A387	"The MCU ID in the MCUSW."	<u>UPP</u>
C601	"The prommer has detected that Vpp voltage level has dropped below the limit"	<u>Vpp from prommer - >Flash</u>

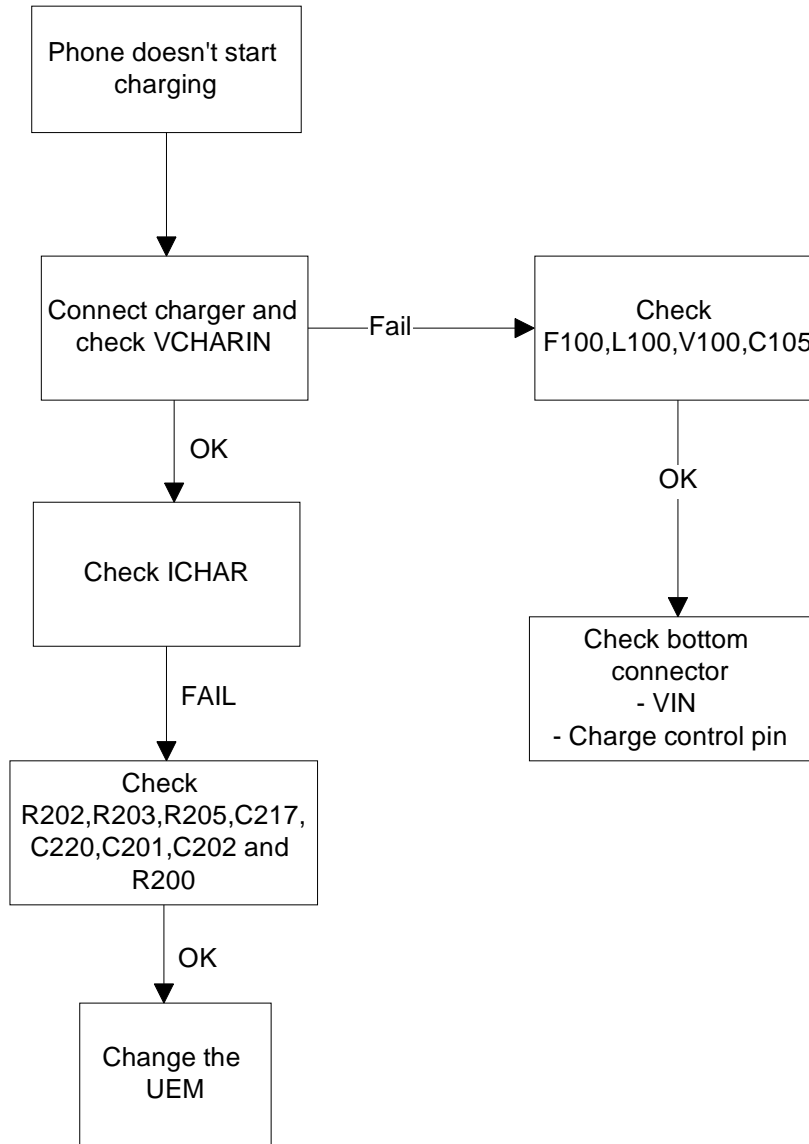
C383 C583 C683	"The Prommer has not received Phone acknowledge to the message." (C383-during algorithm download to target phone) (C583-during erasing) (C683-during programming)	Flash UPP Signals between UPP-Flash
C384 C584 C684	"The Phone has generated NAK signal during data block transfer." (C384-during algorithm download to target phone) (C584-during erasing) (C684-during programming)	Flash UPP Signals between UPP-Flash
C585 C685	Data block handling timeout" (C585-during erasing) (C685-during programming)	Flash UPP Signals between UPP-Flash
C586 C686	"The status response from the Phone informs about fail." (C586-during erasing) (C686-during programming)	Flash

Charging

Note: The charging voltage and current can be checked by connecting the phone to the service software and reading the ad- converter values of the vchar and ichar.

If charging fails when the ACP-9 or the LCH-9 is used, and works with ACP-7, ACP-8 and LCH-8, check that the charge control pin of the system connector is connected to the ground.

Figure 3: Fault tree, charging



Audio failures

In case of audio failures, there are three possibilities to check the audio lines.

- 1 Make a phone call against tester and check audios.
- 2 In case the earpiece/XMIC fails: Feed a 1Khz signal to the XMIC line and measure the signal from the earpiece line. The Audio test box JBA-8 is needed. The Audio loop (*Audio test/Ext In Hp Out*) must be switched on by the Phoenix service software. This loop will connect audios only through the UEM. In this case, the UPP is not used.

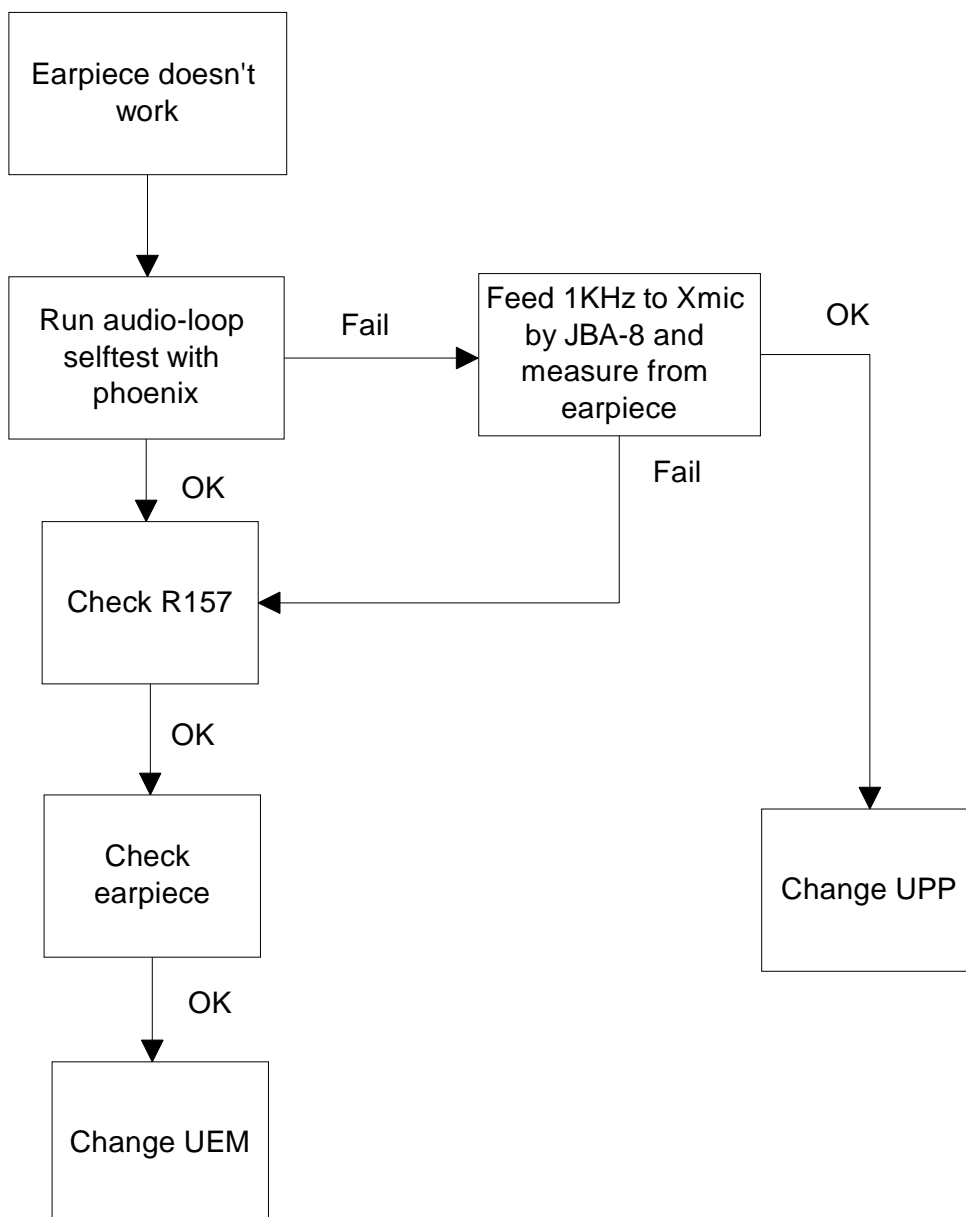
In case the internal microphone/XEAR fails: Feed the tone to the microphone and measure the signal from the XEAR line. The Audio test box JBA-8 is needed. The

Audio loop (*Audio test/Hp In Ext Out*) must be switched on by the Phoenix service software. This loop will connect audios only through the UEM. In this case, the UPP is not used.

- 3 Run the audio-loop self-test with Phoenix (*BB Self Tests / ST_EAR_DATA_LOOP_TEST*). This loop will test the ear-data/mic-data lines between the UPP and the UEM.

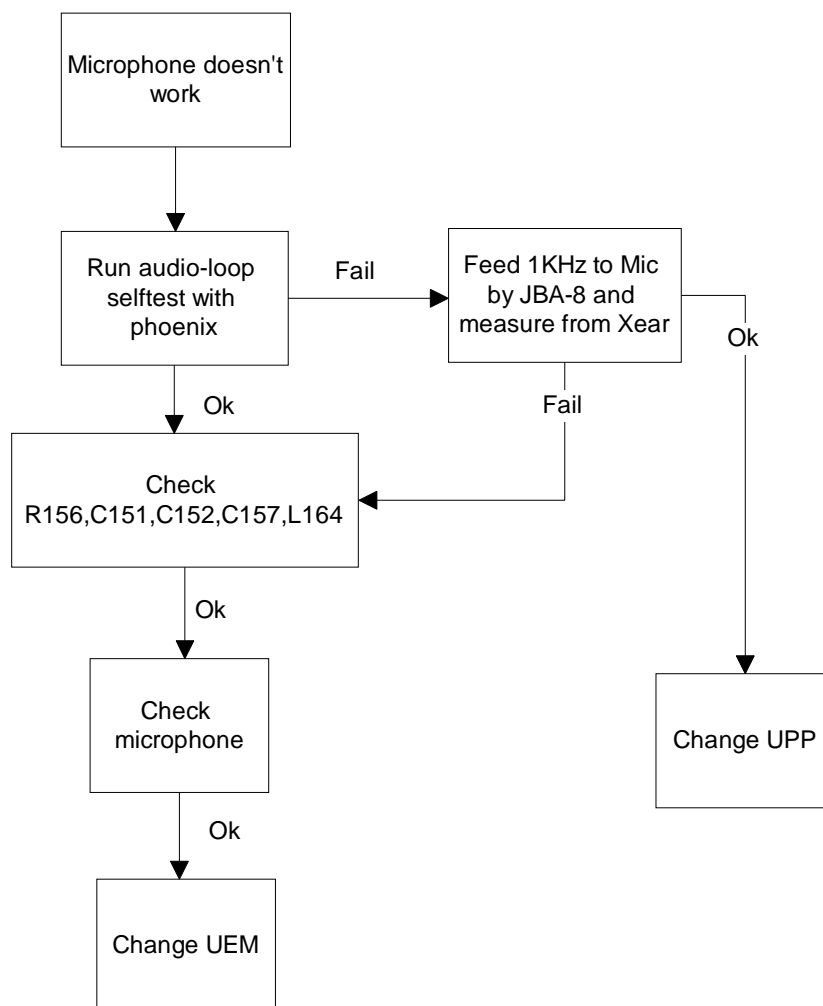
Earpiece or external microphone line does not work

Figure 4: Fault tree, earpiece fail



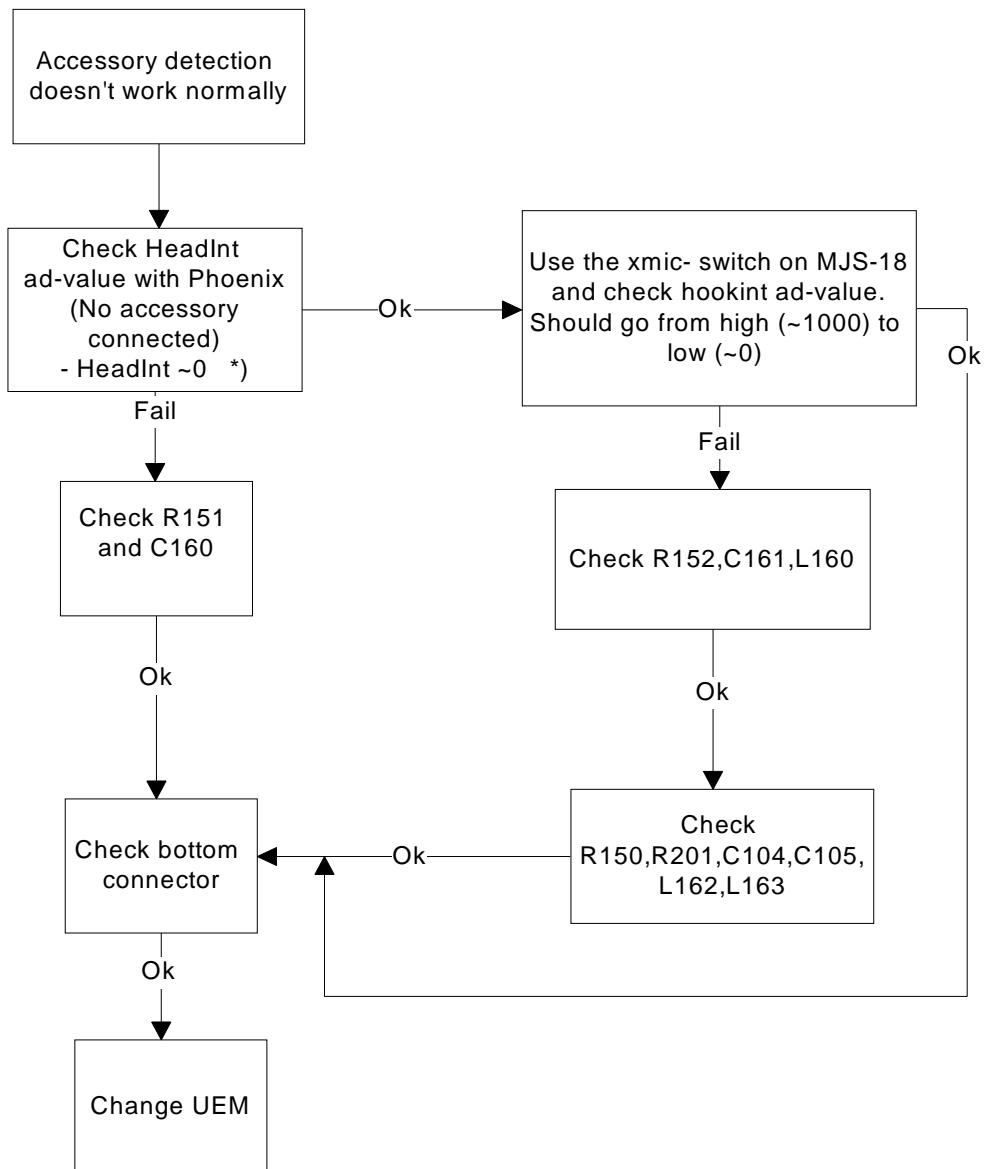
Microphone or XEAR line does not work

Figure 5: Fault tree, microphone fail

**Accessory detection**

- 1 Start the phone in the normal mode when checking for accessory detection.
- 2 The Ad- converter value of the hookint-line can be checked with the Xmic-switch on an MJS-18 test jig. Changes normally from high to low when no accessory is connected.
- 3 When the headset HDC-5 is connected, the hookint- value should be between 600 and 700.

Figure 6: Fault tree, accessory detection



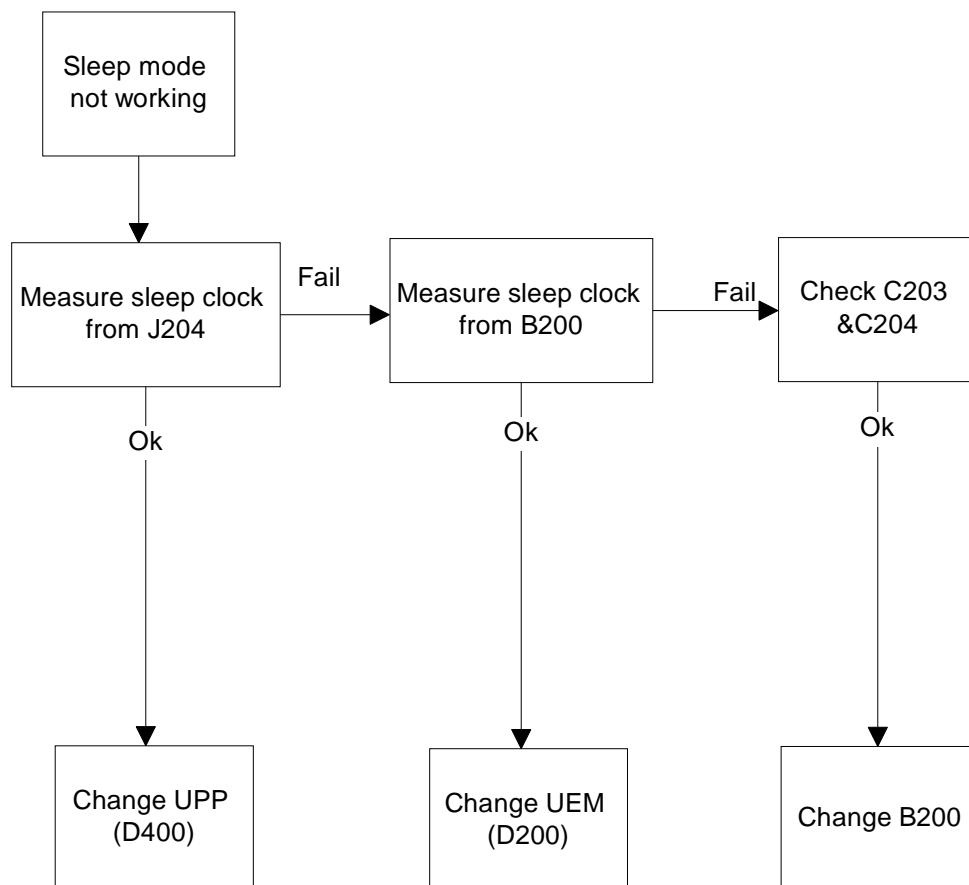
* On local or test mode
 Headint <400

Sleep Clock

Missing/non-functional sleep clock causes

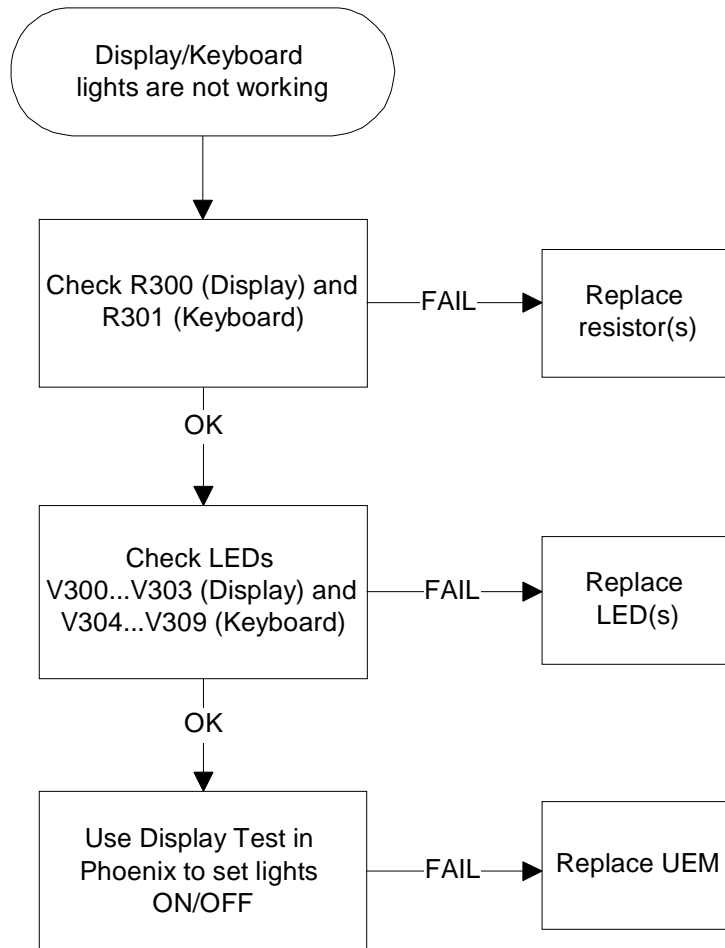
- entering sleep mode fails (higher current consumption -> shorter standby time).
The
- Baseband self tests cannot be run.
- Phone clock (on display) does not function properly.

Figure 7: Fault tree, sleep clock



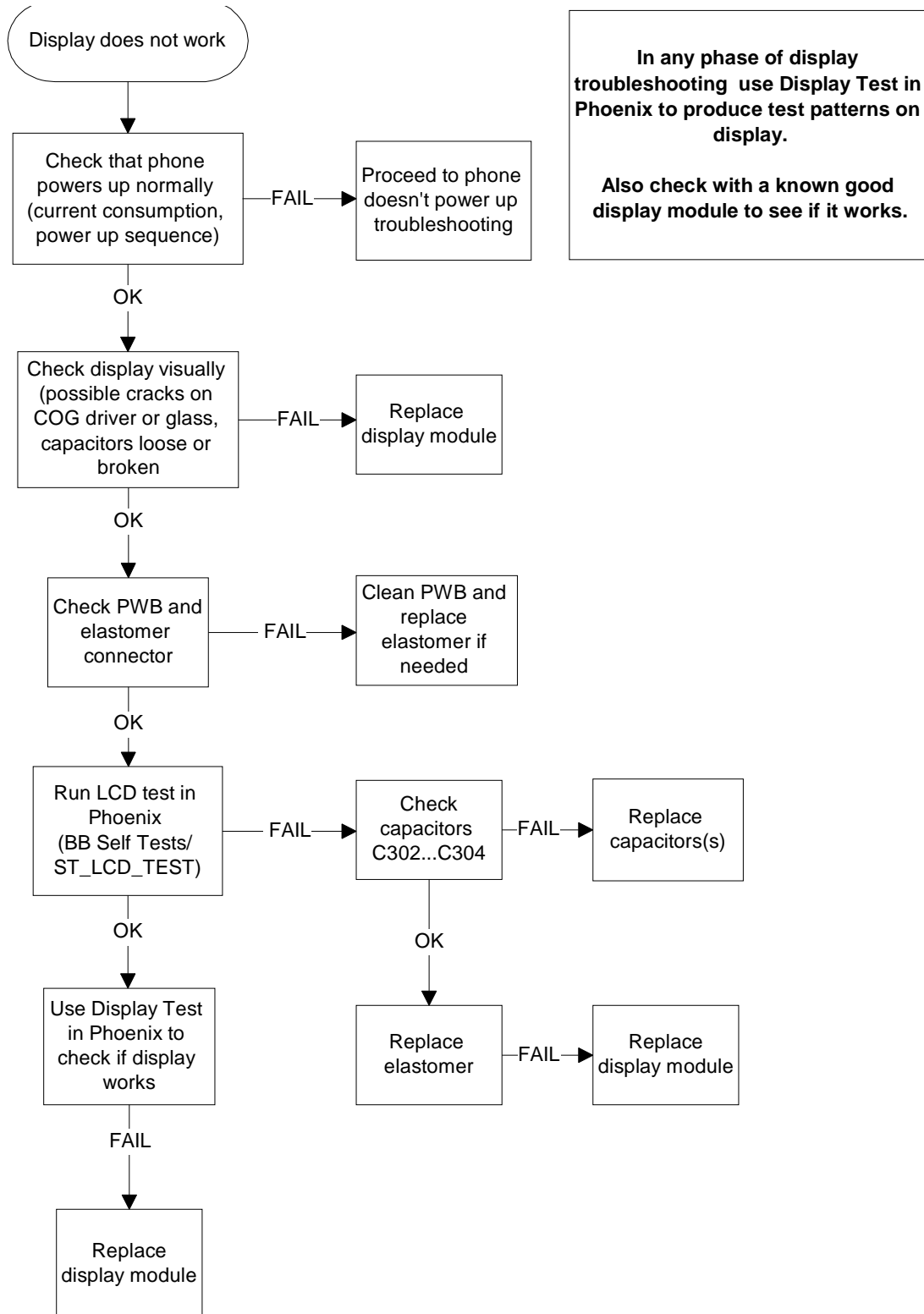
Display/Keyboard lights do not work

Figure 8: Fault tree, Display/Keyboard lights.



Display does not work

Figure 9: Fault tree, Display does not work



Buzzer does not work

Check spring connectors and C306.

- No sound: Try using a known good buzzer instead. If there is still no sound, change the UEM (D200)
- Weak sound: Use the Audio test in Phoenix to set the buzzer parameters (frequency, strength). If it does not effect the sound level, change the buzzer.

Vibra does not work

Check spring connectors and C307&C308.

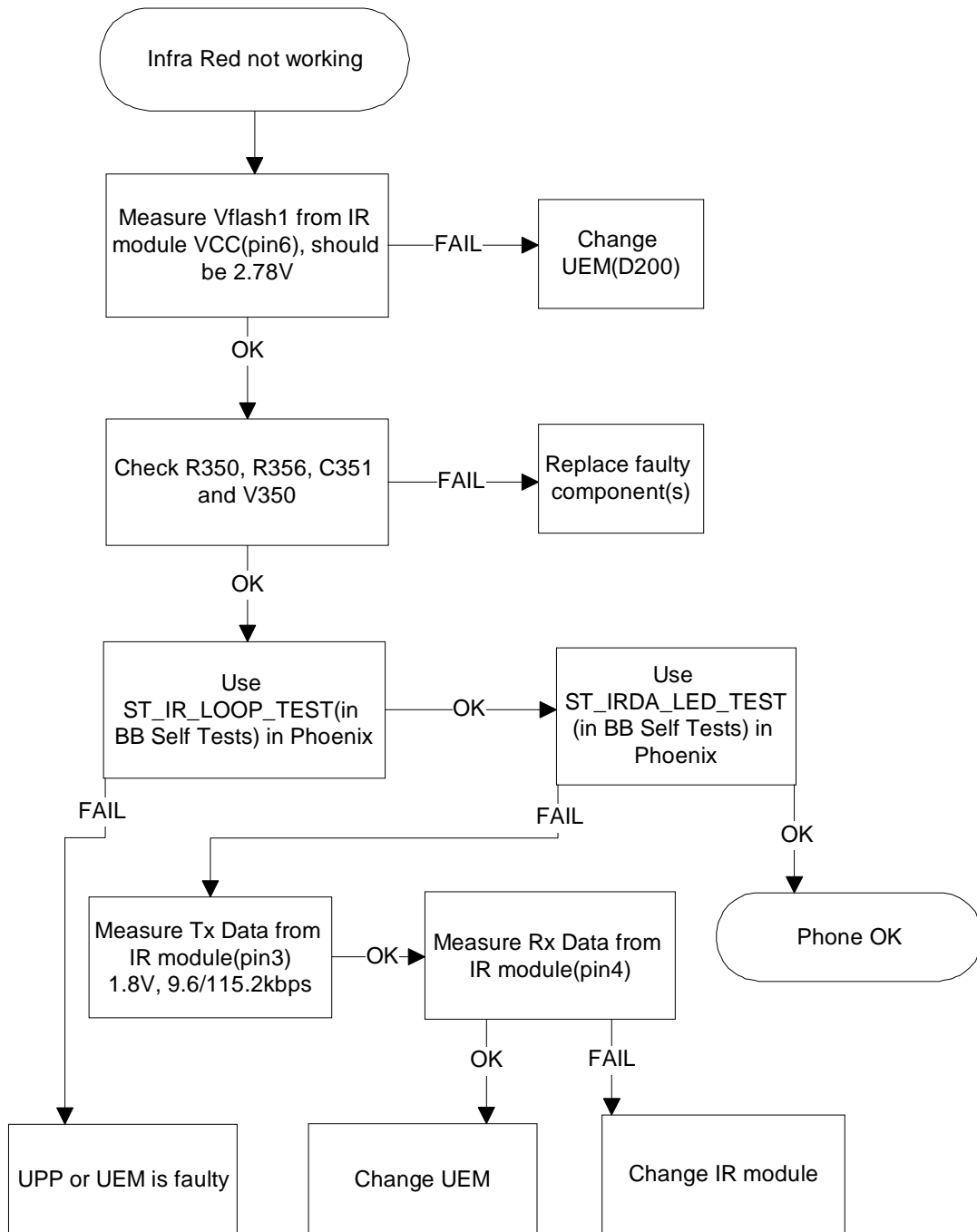
- No vibration: Try using a known good vibra instead. If there is still no vibration, change the UEM(D200)
- Weak vibration: Use the Vibra test in Phoenix to set the vibra's parameters (frequency, duty cycle). If it does not effect the magnitude of the vibration, change the vibra.

Keyboard does not work

Check that there is no dirt between the dome sheet and the PWB. If the keyboard still does not operate normally, try to use the Keyboard test in Phoenix to see if the pressed key is identified. If it is not identified, change the UPP(D400).

Infrared does not work

Figure 10: Fault tree, infrared



Receiver Troubleshooting

General instructions for RX troubleshooting

Start the Phoenix-software and use it to start the required RX-mode of the mobile phone. The troubleshooting flowchart is divided into three steps: (1) general checking, (2) local checking and (3) RX- chain checking.

Always use an RF-cable connected from an external RF-connector to the analyzer via (RF-power) attenuator. This is important to protect the analyzer against excessive RF-power and not to allow leakage of undesired RF-power into the cellular frequencies.

- 1 Start the Phoenix-software and select the TX mode under *Testing* (AMPS, DAMPS).
- 2 It is useful to select the mid channel (383 for AMPS/DAMPS) and the power level 2.

Note: Before changing ASICS or Filters, all solderings and missing components must be checked visually. After any possible component changes, the phone must be tuned with the Phoenix autotune SW.

Path of the received signal

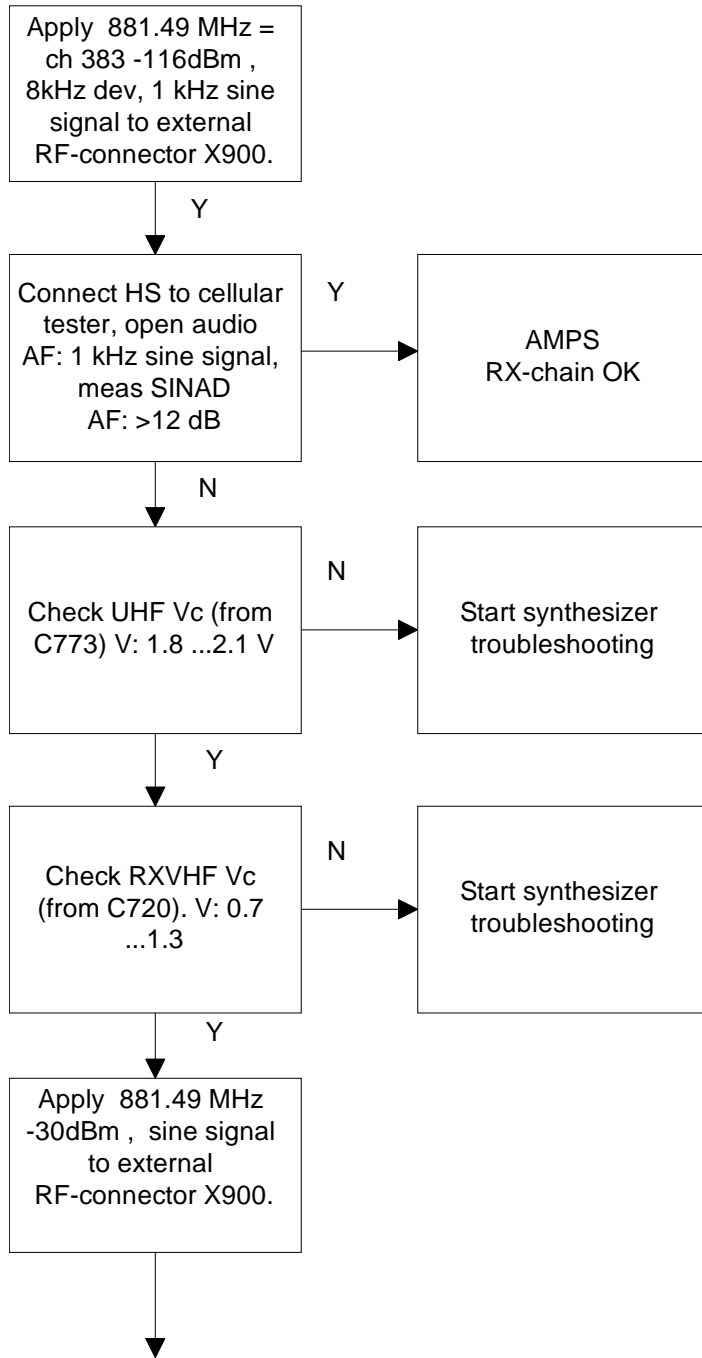
Block level description of the receiver:

(Antenna/ext RF) – Diplexer – Duplexer – Low Noise Amplifier (LNA) – RX band filter – First mixer – 135.54 MHz RX IF filter – IF-amplifier – second mixer – 14 kHz low-pass filter – adjustable IQ amplifier – Baseband.

Fault finding charts for receiver chain

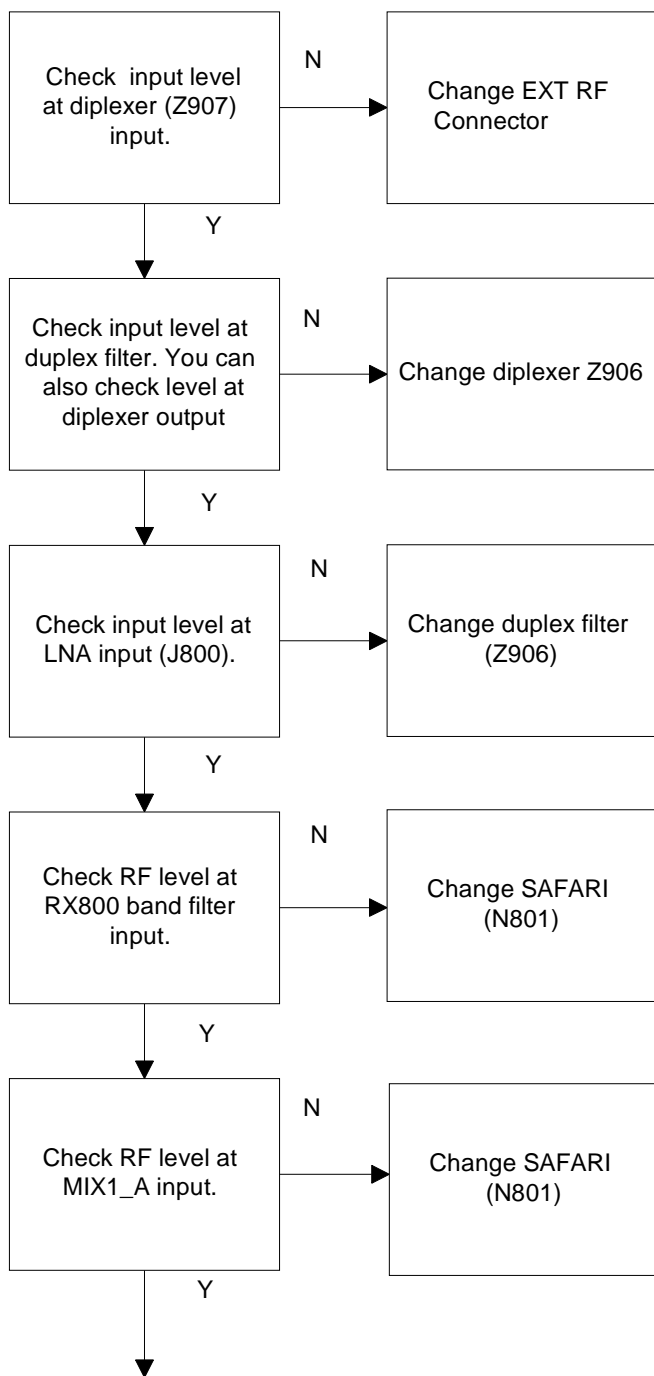
AMPS

Figure 11: Fault tree. AMPS



Note!

These tests are done in local mode by using RF Test UI.



Note!

Check all soldering and components in antenna circuit before changing

Note!

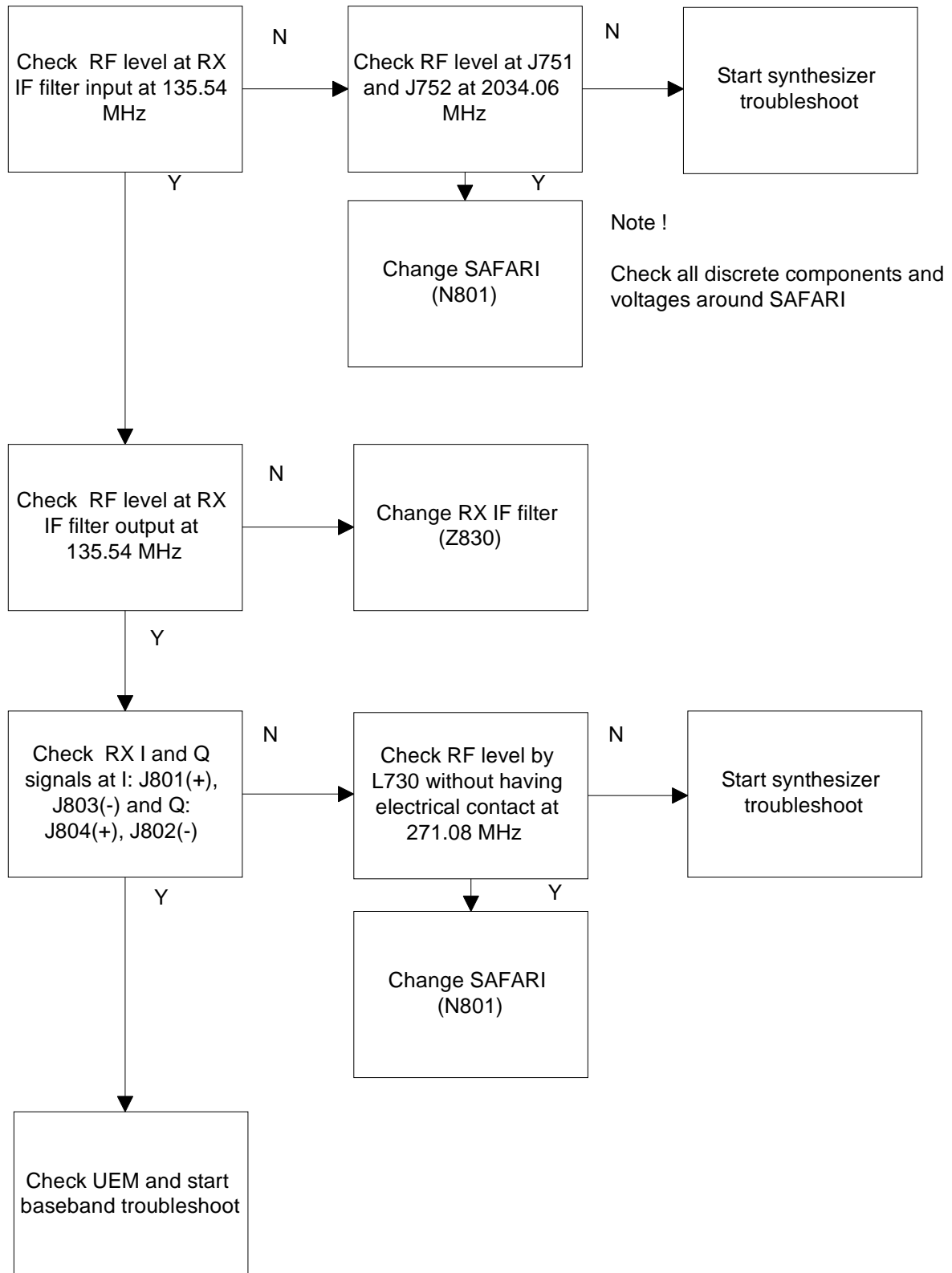
Check all soldering and discrete components of frontend.

Note!

Check all solderings and discrete components around SAFARI before changing

Note!

From the begin to MIX_1A input you should see strong RF signal level at 881.49 MHz. Exact level is difficult to define here because variable impedances in RX path. Also used probe type affects to the results.



TDMA800

Since the same physical signal path is used for both analog and digital modes at the lower band, there is no need for additional troubleshooting in the digital mode. So if the digital mode at the lower band is not working properly, start the analog mode troubleshooting.

Transmitter Troubleshooting

General instructions for TX troubleshooting

Always use an RF-cable connected from an external RF-connector to the analyzer via (RF-power) attenuator. This is important to protect the analyzer against excessive RF-power and not to allow leakage of undesired RF-power into the cellular frequencies.

- 1 Start the Phoenix-software and select the TX mode under *Testing* (AMPS, DAMPS).
- 2 It is useful to select the mid channel (383 for AMPS/DAMPS) and the power level 2.

Note: After any component change, tune the phone with the Phoenix autotune SW

Path of the transmitted signal

AMPS/DAMPS

UEM TX I/Q DA-converters -> I/Q modulator and Digital gain step amplifier (SAFARI) -> IF-BPF -> Upconverter+driver -> BPF -> PA -> Coupler (Power detector) -> Duplex-filter -> Diplexer -> EXT RF-connector -> Antenna

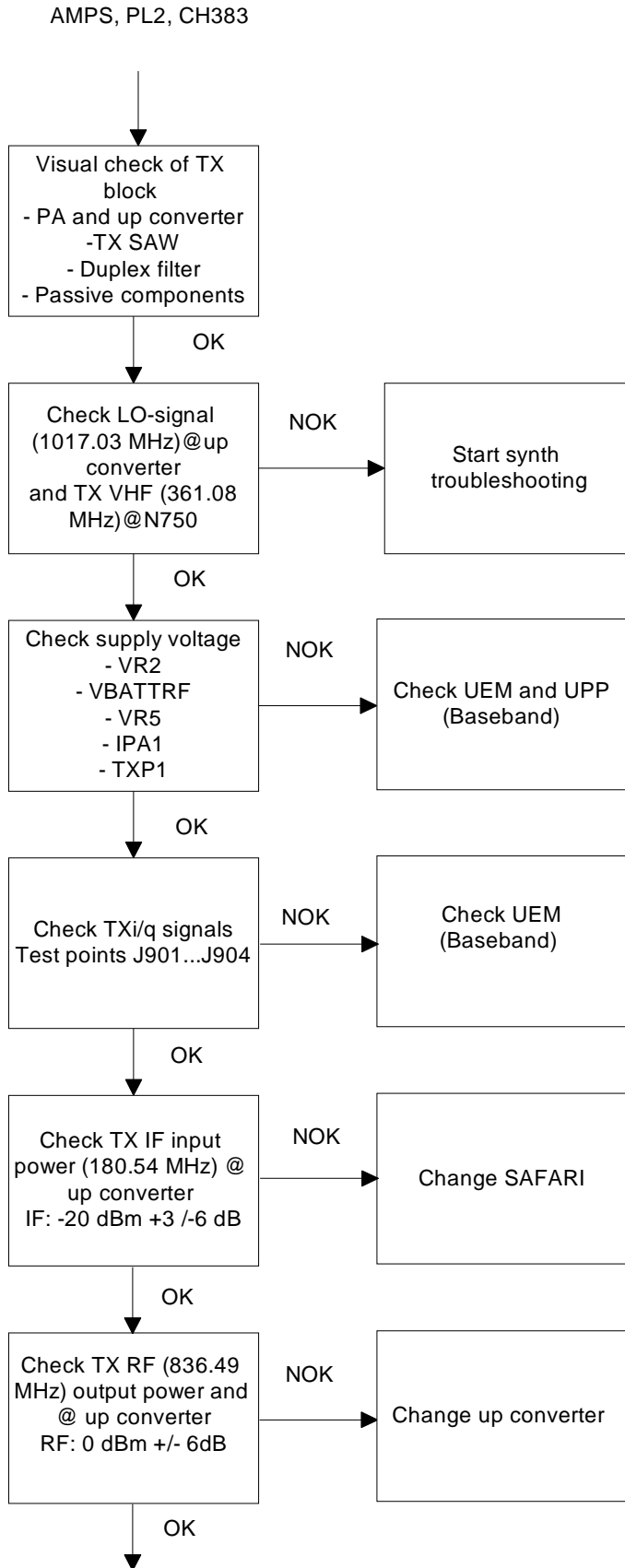
Fault finding charts for the transmitter

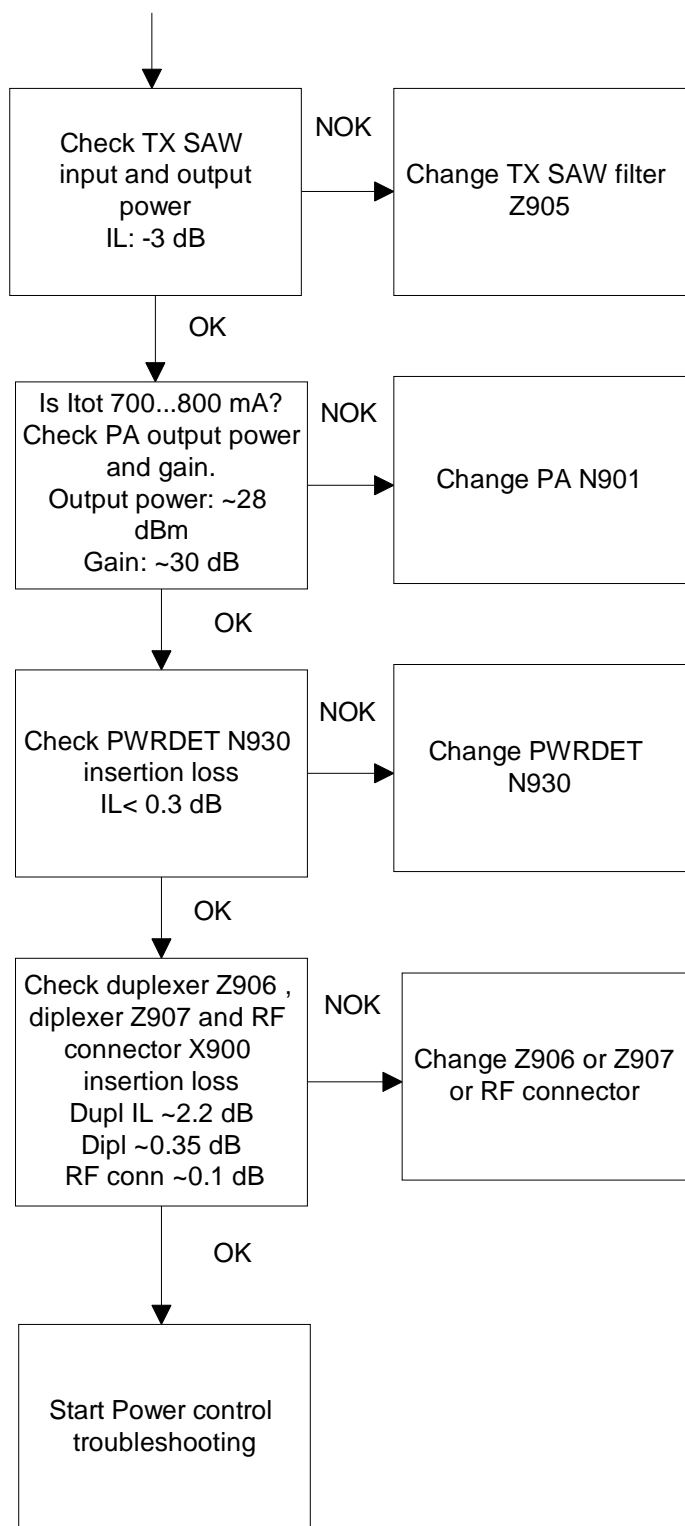
AMPS

- 1 Start the Phoenix-software and set the phone to the Analog mode by using RF Test UI.
- 2 Set the channel to 383 and the power level to 2.
- 3 Connect the RF-cable to the Ext RF connector.
- 4 Connect the cable to the Spectrum analyzer input.
- 5 Measure the RF level.

Note: Insertion loss of the cable and attenuations.

Figure 12: Fault tree. Transmitter troubleshooting





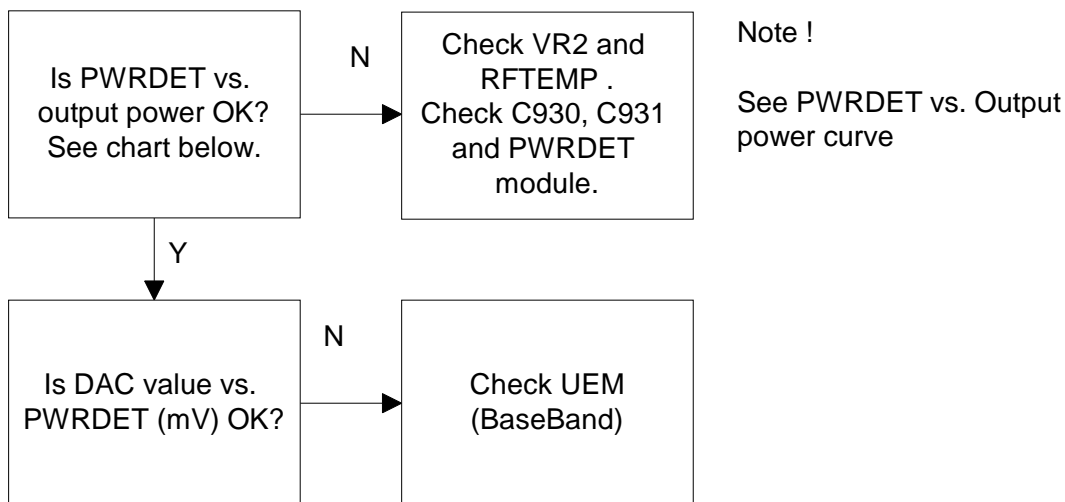
TDMA800

The transmitter chain is exactly the same as the AMPS-mode, except for the IPA current, and thus it is important that the AMPS have no faults.

Power control loop

Basically, the power detection is done with the Power detector module and power control is done inside the SAFARI. Power detection is similar for both bands, except that both bands have their own coupler.

Figure 13: Fault tree: Power control loop



The detected voltages are illustrated in the following table and diagram:

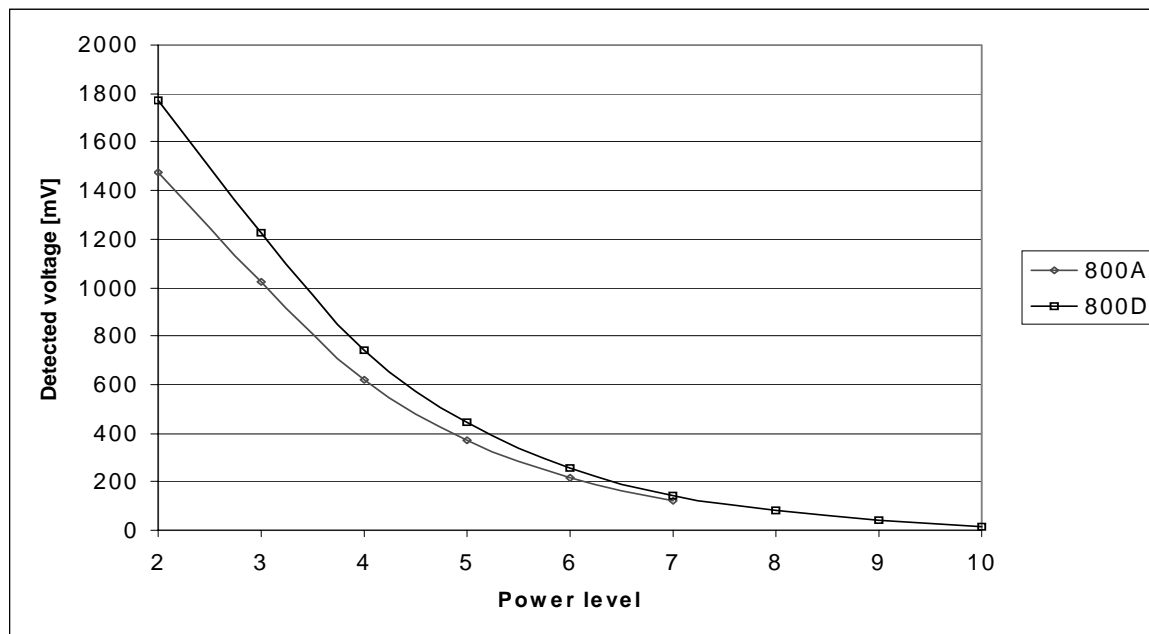
Table 3: Typical detected voltages at power levels PL2...PL10

	800A			800D		
	Pout	TXPWRDET		Pout	TXPWRDET	
PL	dBm	dac	mV	dBm	dac	Mv
2	25.5	559	1475	27.3	671	1771
3	21.5	388	1023	23.3	465	1228
4	17.5	234	618	19.3	281	742
5	13.5	140	371	15.3	168	445
6	9.5	81	215	11.3	98	258
7	5.5	45	120	7.3	54	144
8	-	-	-	3.3	31	82
9	-	-	-	-0.7	15	40
\$0	-	-	-	-4.7	5	14

Note: DAC values may vary about +- 20%

Note: TXPWRDET is the difference between TX on burst and off burst.

Figure 14: Detected voltages



Synthesizer Troubleshooting

There are four oscillators generating the needed frequencies for RF-section. 19.44 MHz reference oscillator, 2GHz UHF VCO, TX VHF cascade amplifier VCO and RX VHF VCO integrated in Safari. RX VHF frequency is fixed 271.08 MHz and TX VHF has two fixed frequencies, 361.08 MHz for lowband and 363.6 MHz for upper band. VCO's operating frequencies are controlled by PLL-circuit of SAFARI. All locals are locked to stable 19.44MHz reference oscillator.

The frequency range for 2GHz UHF VCO is: 2009.1 - 2059.02 MHz.

It is practical way to check out synthesizer status by measuring control voltage of the VCO from Integrator capacitor. If voltage is stable and reasonable, local oscillators are running correctly.

19.44 MHz reference oscillator

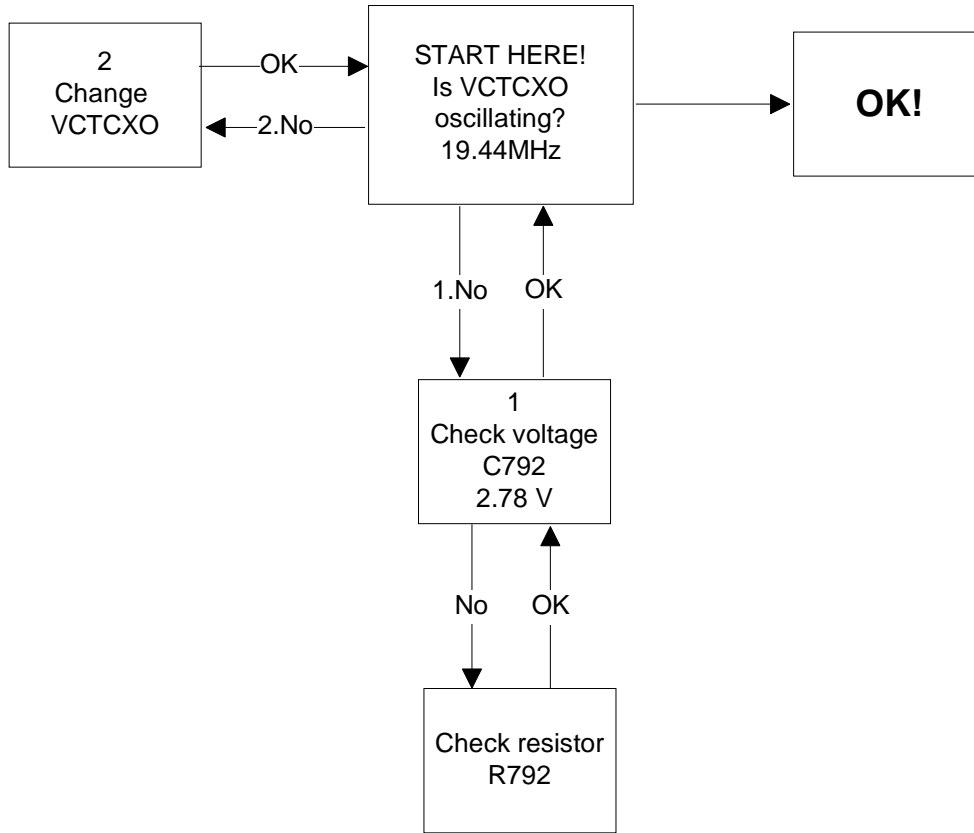
The 19.44 MHz oscillator frequency (G790) is controlled by UEM. This 19.44 MHz signal is connected to SAFARI and there in PLL-circuits and from SAFARI to UPP.

All synthesizers use divided VCTCXO signal as a reference signal for Phase locked loop to provide correct LO-frequency.

Baseband also needs a reference signal where it can generate necessary the clock signals and the VCTCXO output signal is also buffered and connected to the UPP.

Fault finding chart for 19.44 MHz oscillator

Figure 15: Fault tree: 19.44 MHz oscillator



RX VHF VCO

The RX VHF VCO signal is used to generate receiver Intermediate frequency. RX VHF VCO has one fixed frequency 271.08 MHz. Operating frequency is locked in Phase locked Loop.

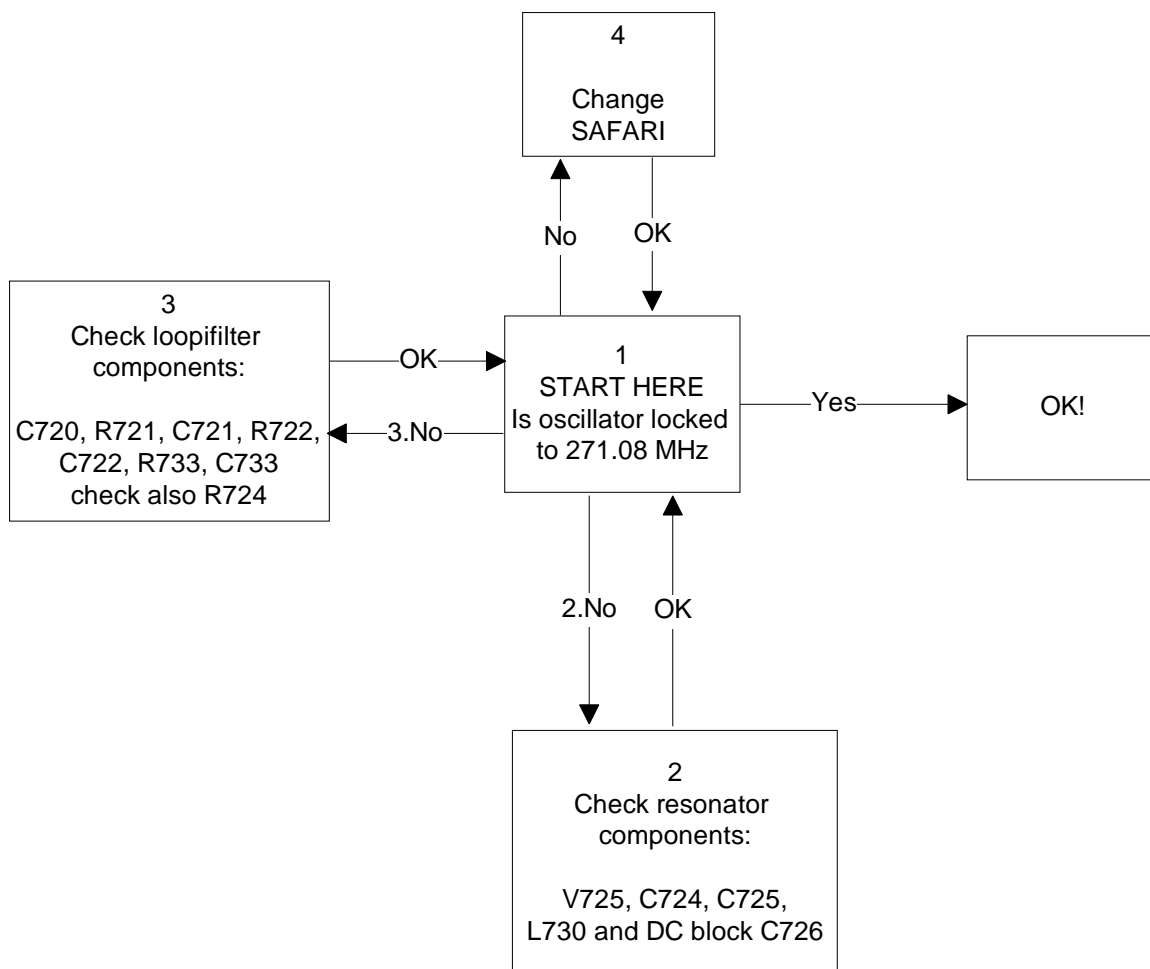
RX VHF VCO output signal is fed to SAFARI. Inside the SAFARI signal is divided for Phase detector and RX parts. Before I/Q-modulator frequency is divided by 2.

Fault finding chart for RX VHF VCO

Measure oscillator frequency over L730 coil. Do not connect probe to pads!

AMPS, CH383

Figure 16: Fault tree: RX VHF VCO



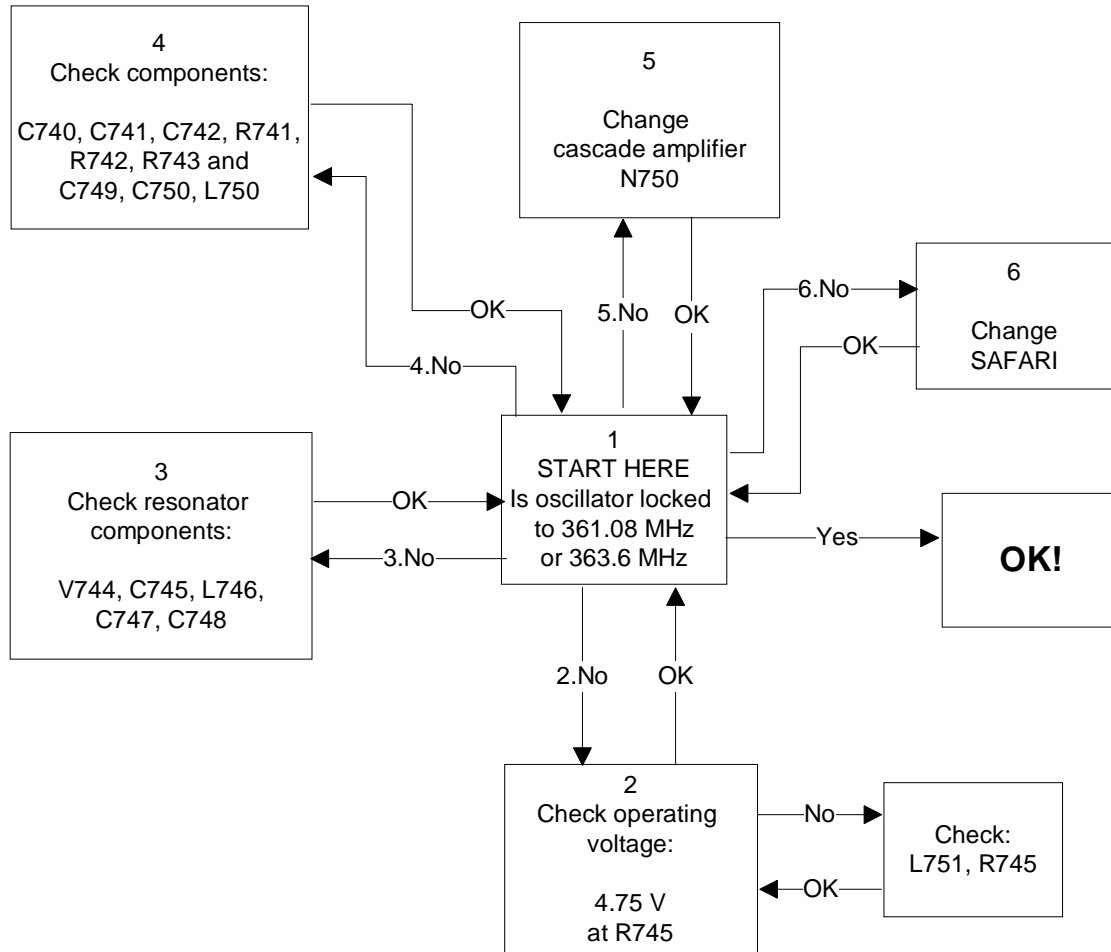
TX VHF VCO

The TX VHF VCO signal is used to generate transmitter Intermediate frequency. The frequency of the TX VHF VCO is 361.08MHz. The operating frequency is locked in the Phase locked Loop and the frequency is divided by two before modulator.

Fault finding chart for TX VHF VCO

AMPS, CH383 -> TX VHF frequency 361.08 MHz

Figure 17: Fault tree: TX VHF VCO



UHF SYNTHESIZER

The UHF synthesizer consists of a external 2GHz UHF VCO, loop filter and integrated PLL in Safari RF-IC. Safari T IC contains stages like counters, pre-scaler, divider by two, phase and frequency comparator and a charge pump circuit which allows two operational modes analog and digital. The UHF VCO Oscillation frequency is from 2009.1MHz to 2059.02MHz.

In AMPS & TDMA800, two GHZ UHF VCO frequency signal is first divided by two inside SAFARI and then that divided signal is fed to mixers. The output frequency of the VCO depends on the DC-control voltage which is controlled by PLL-circuit inside SAFARI.

Fault finding chart for UHF SYNTHESIZER

Figure 18: Fault tree: UHF Synthesizer

