

OsmoTRX - Bug #4468

"RSSI offset" default of 0 is not useful

03/21/2020 08:18 PM - laforge

Status: Feedback	Start date: 03/21/2020
Priority: Normal	Due date:
Assignee: Hoernchen	% Done: 80%
Category:	
Target version:	
Spec Reference:	
Description	
<p>The "rssi offset" value that can be configured via the VTY and command line arguments is initialized to a default of 0.</p> <p>This does not work out at all, at the very least it's proven to be bogus on the popular USRP B2xx hardware in DCS 1800 (See #4467). I would actually be surprised if it's correct on any hardware at all.</p> <p>In an ideal world, every GPSDR vendor would ship every unit together with a proper calibration table over frequency, so that a power level as seen in the I/Q samples can be translated into an absolute value in dBm.</p> <p>However, the world is far from ideal, and the best we can do is try to measure this offset for the few commonly used devices we have available (B200, B210, N200, LimeSDR-mini, LimeSDR USB) and then use that value instead of '0'.</p> <p>We may need a value per band (particularly on LMS where different bands go through different RF paths), and it will of course also change depending on how the hardware receive gains/LNAs are configured. Also, it will drift over frequency within the band, and it will of course have a spread across different units of a given device type.</p> <p>However, I guess anything is better than "0" at this point.</p>	
Related issues:	
Related to OsmoBTS - Bug #4467: bad voice quality in current osmo-bts-trx master	In Progress 03/20/2020
Related to OsmoTRX - Bug #3949: osmo-trx-lms: improve runtime gain setting (m...	New 04/23/2019
Related to OsmoTRX - Feature #4583: generate tx-power correlation table value...	Resolved 06/05/2020

Associated revisions

Revision 4ffdca10 - 04/14/2020 05:16 PM - dexter

doc: apply an rssi-offset of 28 by default.

Set an rssi offset of 28 in the example configs to make sure that the power control loop gets RSSI values that match at least half way the reality when the 1800 Mhz band is used. For other bands the value will be different (See also related osmocom ticket)

Change-Id: I62725fe454f54e2c7cb7550dadb1e6fc94337d78
Related: OS#4468

Revision e91544d7 - 10/14/2020 10:53 AM - pespin

Calculate RSSI offset based on RxGain configuration

Prior to this patch, osmo-trx relied totally on proper VTY configuration being set in "rssi-offset" together with the RxGain set through TRXC in order to provide correct Uplink RSSI measurements to bts-trx.

With this patch, RSSI is now by default calculated (in LMS and UHD backends) based on the currently set RxGain, by providing empirically discovered values. Still, for backward compatibility, the old "rssi-offset" command will overwrite completely the per-default calculated rssi offset.

A new optional parameter "relative" is added at the end of the "rssi-offset" VTY command to flag the value as relative to the newly per-default calculated value. This way specific setups (like adding a LNA / RF fronted) can still be expressed while still keeping the automatic per-default offset.

History

#1 - 03/21/2020 08:19 PM - laforge

- Related to Bug #4467: bad voice quality in current osmo-bts-trx master added

#2 - 03/21/2020 08:19 PM - laforge

- Related to Bug #3949: osmo-trx-lms: improve runtime gain setting (missing calibration) added

#3 - 03/22/2020 04:03 PM - laforge

I've done some initial measurements on a B210 at ARFCN 871, using my Racal 6113 BTS tester.

Using the default RxGain value of 38 (half of the maximum value 76), a RSSI offset of 28 renders correct RxLev vlaues.

It needs to be determined how much this is influenced by frequency, gain, and spread across B2xx units.

#4 - 03/22/2020 06:44 PM - laforge

- File `osmo-trx-calib.gnumeric` added

- Status changed from New to In Progress

- % Done changed from 0 to 20

I took some more measurements with two different B210 units and one LimeSDR-USB at different sides of the 900 MHz and 1800 MHz bands.

- USRP spread between B210 units is < 1 dB, i.e. neglectible
- rssi-offset for B210 in 1800 MHz should be "rxGain - 11"
- rssi-offset for B210 in 900 MHz should be "rxGain - 7.5"
- rssi-offset for LimeSDR-USB in 1800MHz should be "rxGain - 17" (assuming LNAW)
- rssi-offset for LimeSDR-USB in 900MHz should be "rxGain - 6" (assuming LNAL)

Attaching detailed measurements as gnumeric spreadsheet.

So the best approach would probably be to dynamically adjust the rssi-offset every time the RxGain is being set.

The question is a bit how to do this properly in a way that

1. we have sane defaults
2. the user can still add an additional offset to express e.g. that he's added an external LNA / RF frontend
3. we remain backwards-compatible with previous config file / command line argument semantics

So what about:

- if the existing rssi-offset is given via command line or vty, behavior remains as is
- we introduce a new vty command `rssi-offset mode (absolute|relative)`
 - absolute is the old behavior, where the user-provided value is used as-is, irrespective of gain
 - relative is the new behavior, where the user-provided value (default:0) is applied relative to the device+band specific default values (see my maasurements for 900+1800 above)

#5 - 06/05/2020 01:03 PM - laforge

- Related to Feature #4583: generate tx-power correlation table values for different sdr boards added

#6 - 06/05/2020 01:04 PM - laforge

[pespin](#), would be great to get this wrapped up

#7 - 10/13/2020 06:30 PM - pespin

- Status changed from In Progress to Feedback

- % Done changed from 20 to 80

I submitted a patch implementing what was described in this ticket:

<https://gerrit.osmocom.org/c/osmo-trx/+20639>

#8 - 10/14/2020 11:28 AM - pespin

- Assignee changed from pespun to Hoernchen

Patch implementing the described approach have been merged.

Still, it's still unclear what's the best way to implement that in osmo-trx-ipc. Either we retrieve the rssiOffset from the IPC Driver or we expected the received bursts from it to always be rssiOffset=0. I'd welcome some feedback from @Hoernchen on this topic.

#9 - 10/14/2020 02:40 PM - laforge

On Wed, Oct 14, 2020 at 11:28:29AM +0000, pespun [REDMINE] wrote:

Still, it's still unclear what's the best way to implement that in osmo-trx-ipc. Either we retrieve the rssiOffset from the IPC Driver or we expected the received bursts from it to always be rssiOffset=0. I'd welcome some feedback from @Hoernchen on this topic.

the radio heads which are the current target for osmo-trx-ipc all have calibrated radios, unlike a GP-SDR device.

So IMHO all that is needed to know is a single value defining the fixed relationship of 'how many dBm (absolute received power level) corresponds to full scale of your I/Q samples, i.e. the highest value you can receive from the ADC'.

Files

osmo-trx-calib.gnumeric	7.96 KB	03/22/2020	laforge
-------------------------	---------	------------	---------