|  |  |
| --- | --- |
| Document information | |
| Info | Content |
| Author(s) | Olivier Doaré |
| Department | BL-IDA, |
| Keywords | STRX, TX |
| Abstract | Design documentation of the TX and TXBIST digital control IP |
| SGI | <value> |
| ECCN | <code> |
| US origin | No |

|  |  |  |
| --- | --- | --- |
| Distribution information | | |
| Name | Department | Address |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

| Revision history | | |
| --- | --- | --- |
| Rev | Date | Description |
| 0.1 | 01 April 2021 | Initial version |
| 0.2 | 16 April 2021 | Added DAC linearity and PR settling time (Paul W) |
|  |  |  |
|  |  |  |

Contents

[1. Introduction 5](#_Toc69457298)

[2. Product variant 5](#_Toc69457299)

[3. TX Power mode 6](#_Toc69457300)

[3.1 Power up, down , stand by 6](#_Toc69457301)

[3.2 Sequence based dynamic power 6](#_Toc69457302)

[3.3 Chirp based dynamic power 6](#_Toc69457303)

[3.4 TX enable control 7](#_Toc69457304)

[4. Clock and reset 9](#_Toc69457305)

[5. TX mode of operations 10](#_Toc69457306)

[6. TX application mode 11](#_Toc69457307)

[6.1 Tx phase 11](#_Toc69457308)

[6.2 TX phase init 12](#_Toc69457309)

[6.3 TX gain 12](#_Toc69457310)

[6.4 TX profile 12](#_Toc69457311)

[7. Phase rotator calibration 13](#_Toc69457312)

[7.1 Introduction 13](#_Toc69457313)

[7.2 Block diagram 13](#_Toc69457314)

[7.3 Calibration FSM states 14](#_Toc69457315)

[7.3.1 Start\_CAL FSM state 15](#_Toc69457316)

[7.3.2 ADC\_AVE FSM state 16](#_Toc69457317)

[7.3.3 VGA\_INIT FSM state 17](#_Toc69457318)

[7.3.4 VGA\_Offset FSM state 18](#_Toc69457319)

[7.3.5 VGA\_Gain FSM state 19](#_Toc69457320)

[7.3.6 PR\_OFFSET\_CAL FSM state 20](#_Toc69457321)

[7.3.7 GPATH\_SEL FSM state 21](#_Toc69457322)

[7.3.8 PR\_GAIN\_CAL FSM state 22](#_Toc69457323)

[7.3.9 PR\_PHASE\_CAL FSM state 22](#_Toc69457324)

[7.3.10 END\_CAL 23](#_Toc69457325)

[7.4 Automatic calibration 23](#_Toc69457326)

[7.5 Manual calibration 23](#_Toc69457327)

[7.5.1 Phase rotator DACs input force 23](#_Toc69457328)

[7.5.2 Compensation force 24](#_Toc69457329)

[7.6 Calibration time 25](#_Toc69457330)

[8. TX safety 26](#_Toc69457331)

[8.1 Real time monitoring 26](#_Toc69457332)

[8.1.1 RTM description 26](#_Toc69457333)

[8.1.2 RTM Mask 27](#_Toc69457334)

[8.1.3 RTM FIT 27](#_Toc69457335)

[8.2 Ball break detector 28](#_Toc69457336)

[8.2.1 BBD description 28](#_Toc69457337)

[8.2.2 BBD Mask 29](#_Toc69457338)

[8.2.3 BBD FIT 29](#_Toc69457339)

[8.3 CRC REGISTER 30](#_Toc69457340)

[8.3.1 CRC REGISTER description 30](#_Toc69457341)

[8.3.2 CRC Mask 30](#_Toc69457342)

[8.4 CRC MISO 30](#_Toc69457343)

[8.4.1 CRC MISO description 30](#_Toc69457344)

[8.4.2 CRC MISO MASK 30](#_Toc69457345)

[8.5 CRC MOSI 31](#_Toc69457346)

[8.5.1 CRC MISO description 31](#_Toc69457347)

[8.5.2 CRC MISO MASK 31](#_Toc69457348)

[8.6 TX INL/DNL 31](#_Toc69457349)

[8.6.1 TXINL description 31](#_Toc69457350)

[8.6.2 TX INL MASK 32](#_Toc69457351)

[8.6.3 TX INL FIT 33](#_Toc69457352)

[8.7 TX to TX delta phase variation (TXBIST) 33](#_Toc69457353)

[8.7.1 TX to TX delta variation description 33](#_Toc69457354)

[8.7.2 TX to TX delta variation Mask 34](#_Toc69457355)

[8.7.3 TX to TX delta variation FIT 34](#_Toc69457356)

[9. ADC communication 35](#_Toc69457357)

[9.1 TX ADC /ATB ADC selection 35](#_Toc69457358)

[9.1.1 TX ADC measurement. 35](#_Toc69457359)

[9.1.2 ATB ADC measurement 36](#_Toc69457360)

[9.2 TXADC communication 36](#_Toc69457361)

[9.2.1 During Phase rotator calibration 36](#_Toc69457362)

[9.2.2 Outside from Phase rotator calibration 37](#_Toc69457363)

[9.2.3 Timeout flag 37](#_Toc69457364)

[10. DFT 38](#_Toc69457365)

[10.1 PR DAC linearity 38](#_Toc69457366)

[10.1.1 P-channel current measurement-test 38](#_Toc69457367)

[10.1.2 N-channel current measurement-test 38](#_Toc69457368)

[10.1.3 Disabled DFT linearity test 39](#_Toc69457369)

[10.2 PR settling time 39](#_Toc69457370)

[10.3 IF generation (UPN) 42](#_Toc69457371)

[10.4 FSM STOP 42](#_Toc69457372)

[10.5 ADC measurement in FSM state 42](#_Toc69457373)

[10.6 GPIO 43](#_Toc69457374)

# Introduction

TX digital module is companion of the TX and TXBIST RF module.

* Tx DIG module translate the 7 bits phase from the timing engine to 256 unitary weighted bit input of the TX I and Q phase rotator DAC.
* Tx DIG module improves the TX INL through a self-autonomous compensation mechanism of the Phase rotator impairments.
* The TXDIG with its companion TXANA contains safety mechanism
* Tx DIG module contains DFT features to measure the Phase rotator settling time, check the phase rotator DAC linearity

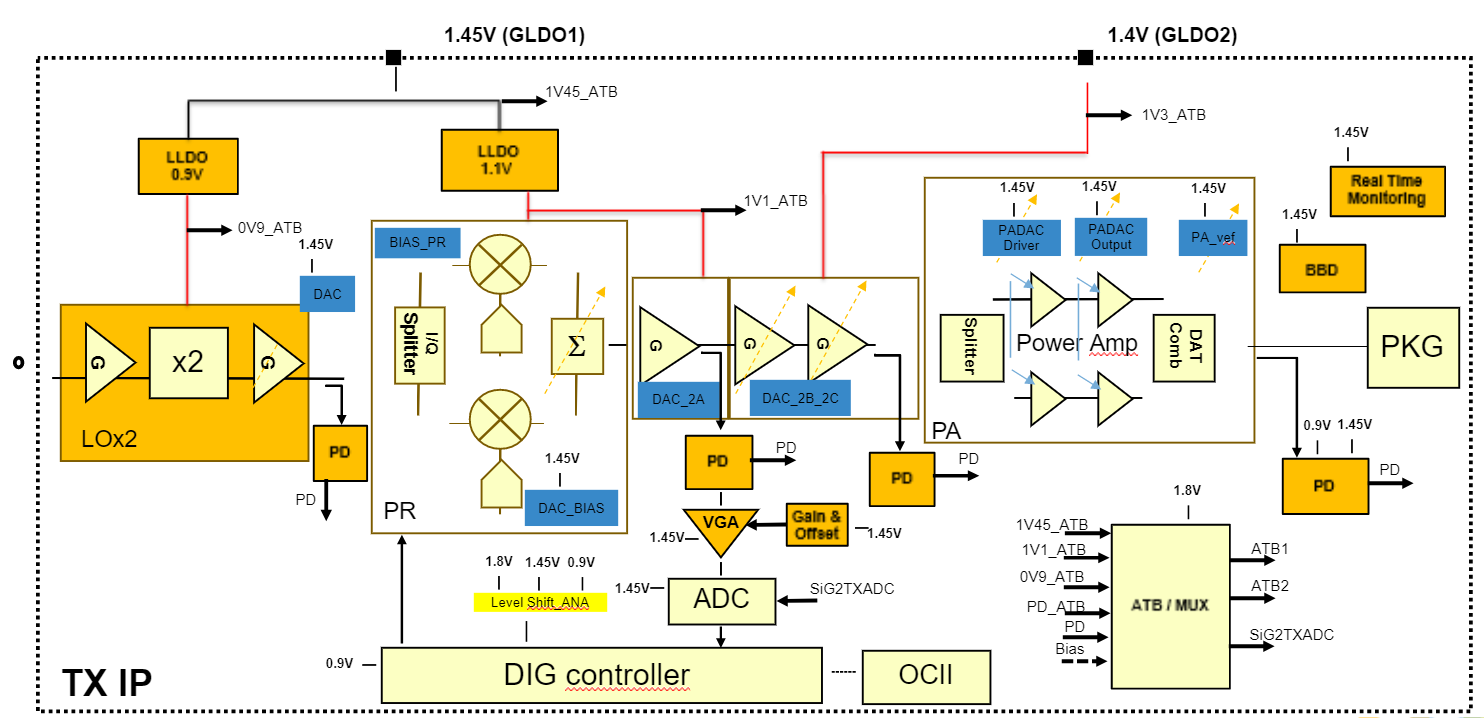


Figure 1: STRX TX

# Product variant

Tx can be configurated using OTP bit

It shall be possible to enable/disable each individual TX-channel via OPT control.

It shall be possible to enable/disable each individual phase rotator controller via OPT control.

It shall be possible to control the maximum output power via OTP control

# TX Power mode

The TX Power sequence shall be initiated by the Timing Engine. Nevertheless it shall be possible to enable through SPI the TX and TXBIST modules.

The TXBIST power sequence shall only be controlled by SPI

## Power up, down , stand by

* The TX power down mode shall be active when level shifter and LDO are OFF
* Standby mode shall correspond to Level shifter ON and LDO are OFF
* The TX power up mode shall be active when Level shifter are On, LDOs ON and TX enabled

## Sequence based dynamic power

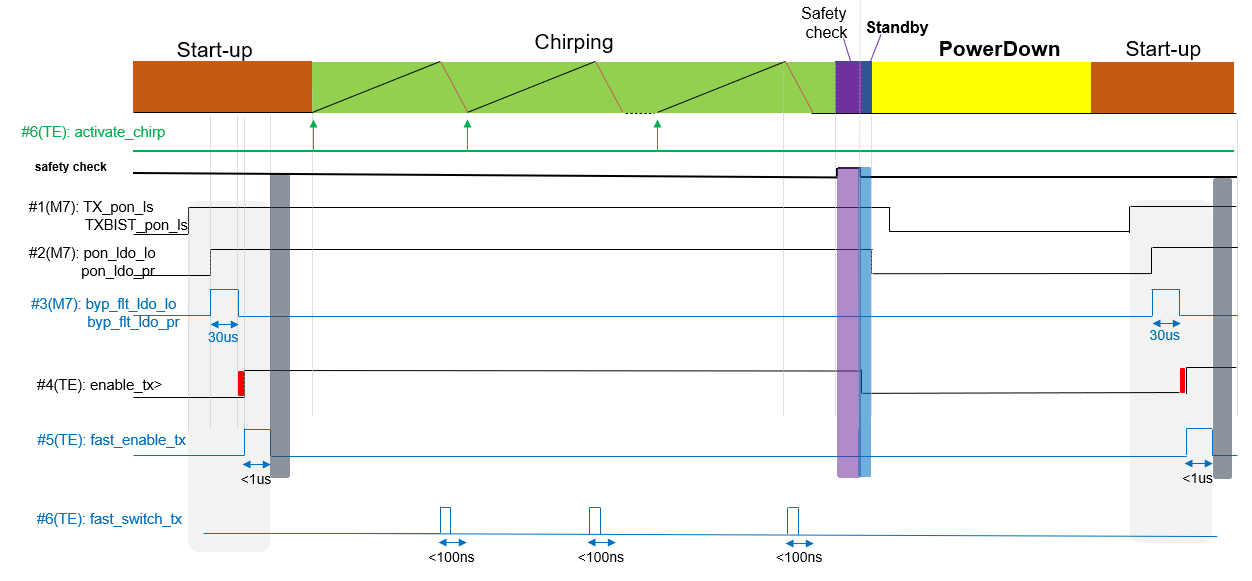


Figure 2: TX sequence based dynamic power

## Chirp based dynamic power

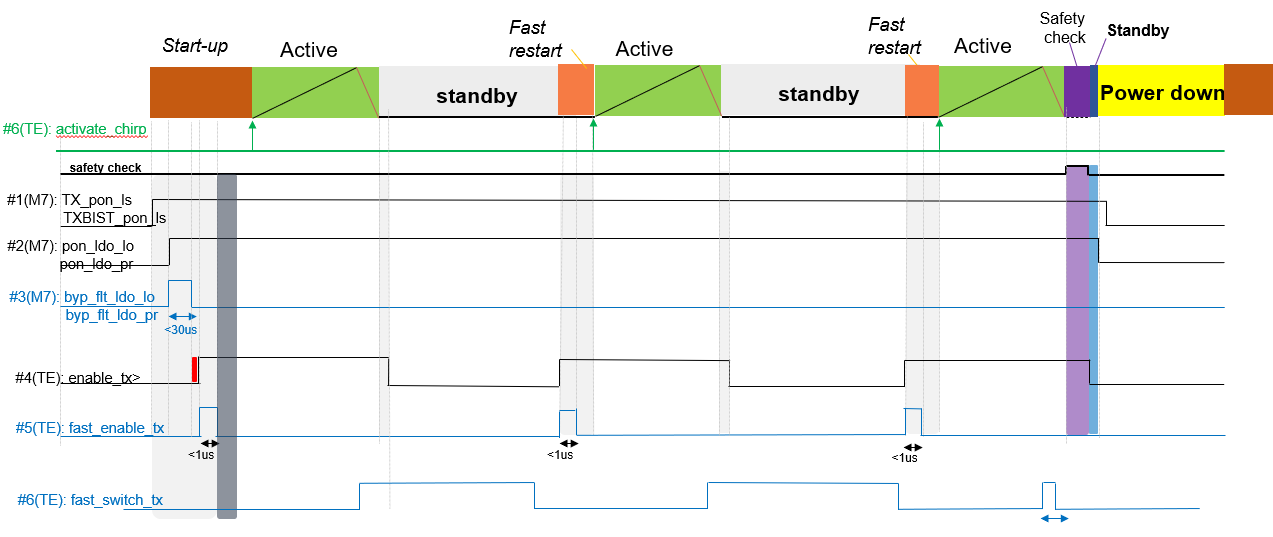


Figure 3:TX chirp based dynamic power

## TX enable control

TX has 2 types of enables.

Enables which are needed for application mode are controlled by a OR gate which input is either the tx\_enable from the Timing engine or a dedicated SPI bit.

Other enables which are needed for other purpose such as calibration , safety , DFT only are managed only by SPI bit.

Table 1: TX enable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***TX\_ADC*** |  |  |  |  |
|  | POWER\_ENABLE | TXADC | FUNC | SPI |
| ***PON\_TXBIST*** |  |  |  |  |
|  | BIAS\_EN | TXBIST | SAFETY | SPI |
|  | TEST\_EN | TXBIST | SAFETY | SPI |
|  | VGA\_EN | TXBIST | SAFETY | SPI |
|  | MIXER\_CORE\_EN | TXBIST | SAFETY | SPI |
| ***TX\_LO*** |  |  |  |  |
|  | EN\_LOX2 | TX | FUNC | TE OR SPI |
|  | EN\_PPD\_LOX2 | TX | FUNC | TE OR SPI |
|  | EN\_FAST\_START\_LOX2 | TX | FUNC | TE OR SPI |
|  | EN\_LOX2\_BIAS | TX | FUNC | TE OR SPI |
| ***TX\_PR*** |  |  |  |  |
|  | EN\_PR | TX | FUNC | TE OR SPI |
|  | EN\_PR\_DAC | TX | FUNC | TE OR SPI |
|  | EN\_LBIAS\_PR | TX | FUNC | TE OR SPI |
|  | EN\_BUF2A | TX | FUNC | TE OR SPI |
|  | EN\_DAC\_BUF2A | TX | FUNC | TE OR SPI |
|  | EN\_PPD\_BUF2A | TX | FUNC | TE OR SPI |
|  | EN\_PPD\_BUF2A\_VGA | TX | FUNC | SPI |
|  | EN\_VGA\_PR | TX | FUNC | SPI |
|  | EN\_VGA\_DAC | TX | FUNC | SPI |
|  | EN\_DAC\_VGA\_IBIAS | TX | FUNC | SPI |
|  | EN\_VGA2\_VBIAS\_VCM | TX | FUNC | SPI |
|  | EN\_VGA\_DAC\_OFST | TX | FUNC | SPI |
|  | FAST\_EN\_PR | TX | FUNC | TE OR SPI |
|  | FAST\_SW\_PR | TX | FUNC | TE OR SPI |
| ***TX\_2B2C*** |  |  |  |  |
|  | EN\_BUF\_2B | TX | FUNC | TE OR SPI |
|  | EN\_BUF\_2C | TX | FUNC | TE OR SPI |
|  | EN\_DAC\_BUF2B2C | TX | FUNC | TE OR SPI |
|  | EN\_PPD\_BUF2C | TX | FUNC | TE OR SPI |
|  | FAST\_EN\_BUF | TX | FUNC | TE OR SPI |
|  | FAST\_SW\_BUF | TX | FUNC | TE OR SPI |
| ***TX\_PA*** |  |  |  |  |
|  | EN\_PA\_CORE | TX | FUNC | TE OR SPI |
|  | EN\_PA\_DAC\_ST1\_C | TX | FUNC | TE OR SPI |
|  | EN\_PA\_DAC\_ST2\_C | TX | FUNC | TE OR SPI |
|  | EN\_PA\_DAC\_VREF | TX | FUNC | TE OR SPI |
|  | EN\_PPD\_PA | TX | FUNC | TE OR SPI |
|  | EN\_RTM\_DAC | TX | SAFETY | SPI |
|  | EN\_RTM\_COMP | TX | SAFETY | SPI |
|  | EN\_RTM\_BUFX1 | TX | SAFETY | SPI |
|  | EN\_RTM | TX | SAFETY | SPI |
|  | PPD\_PAOUT\_VGA\_EN | TX | FUNC | SPI |
|  | FAST\_EN\_PA | TX | FUNC | TE OR SPI |
|  | FAST\_SW\_PA | TX | FUNC | TE OR SPI |
| ***BBD\_CTL*** |  |  |  |  |
|  | EN | TX | SAFETY | SPI |
| ***BBD\_DFT*** |  |  |  |  |
|  | EN\_BALL\_CURRENT | TX | SAFETY | SPI |
| ***ADC\_DFT*** |  |  |  |  |
|  | PPD\_BUF\_EN | TX | DFT | SPI |
| ***TXBIST\_CTRL*** |  |  |  |  |
|  | STLING\_TIME\_EN\_N | TXBIST | DFT | SPI |
|  | STLING\_TIME\_EN\_P | TXBIST | DFT | SPI |
| ***PRC\_IF\_TEST*** |  |  |  |  |
|  | TX\_IF\_EN | TX | DFT | SPI |
| ***PR\_STLING\_TIME*** |  |  |  |  |
|  | EN |  | DFT | SPI |
| ***ATB\_CTL*** |  |  |  |  |
|  | ATB1\_EN | TX/TXBIST | DFT | SPI |
|  | ATB2\_EN | TX/TXBIST | DFT | SPI |
| ***VSTRESS*** |  |  |  |  |
|  | TX\_HVST\_EN | TX | DFT | SPI |
|  | TXBIST\_HVST\_EN | TXBIST | DFT | SPI |

# Clock and reset

Clock and reset are managed through 2 internal TXDIG module.

CTRL\_TOP provide to the TX functional TOP a 40MHZ clock and a reset. The detail of CTRL\_TOP shall be defined in a separate document.

Only 1 reset is used in the TX functional Top.

Clk40\_tx\_cal\_clk drives the phase rotator calibrations digital modules .

Clk40\_tx\_cal\_clk is disabled when SPI bit power\_down =1 or when otp\_tx\_pr\_disable=1 or when otp\_tx\_disbale=1

The clock is active when enable\_TX and PR\_CAL\_RUN=1 indicating the calibration is running. clk\_free\_run SPI bit allows to bypass the enable\_tx and pr\_cal\_run

Clk40\_tx\_prc\_clk drives the phase rotator controller digital module .

Clk40\_tx\_cal\_clk is disabled when SPI bit power\_down =1 or when otp\_tx\_pr\_disable=1 or when otp\_tx\_disable=1.

The clock is active when enable\_tx from the Timing engine or clk\_free\_running =1

Clk40\_tx\_clk drives the safety and DFT. The clock is enabled when enable\_tx=1 or clk\_free\_run =1

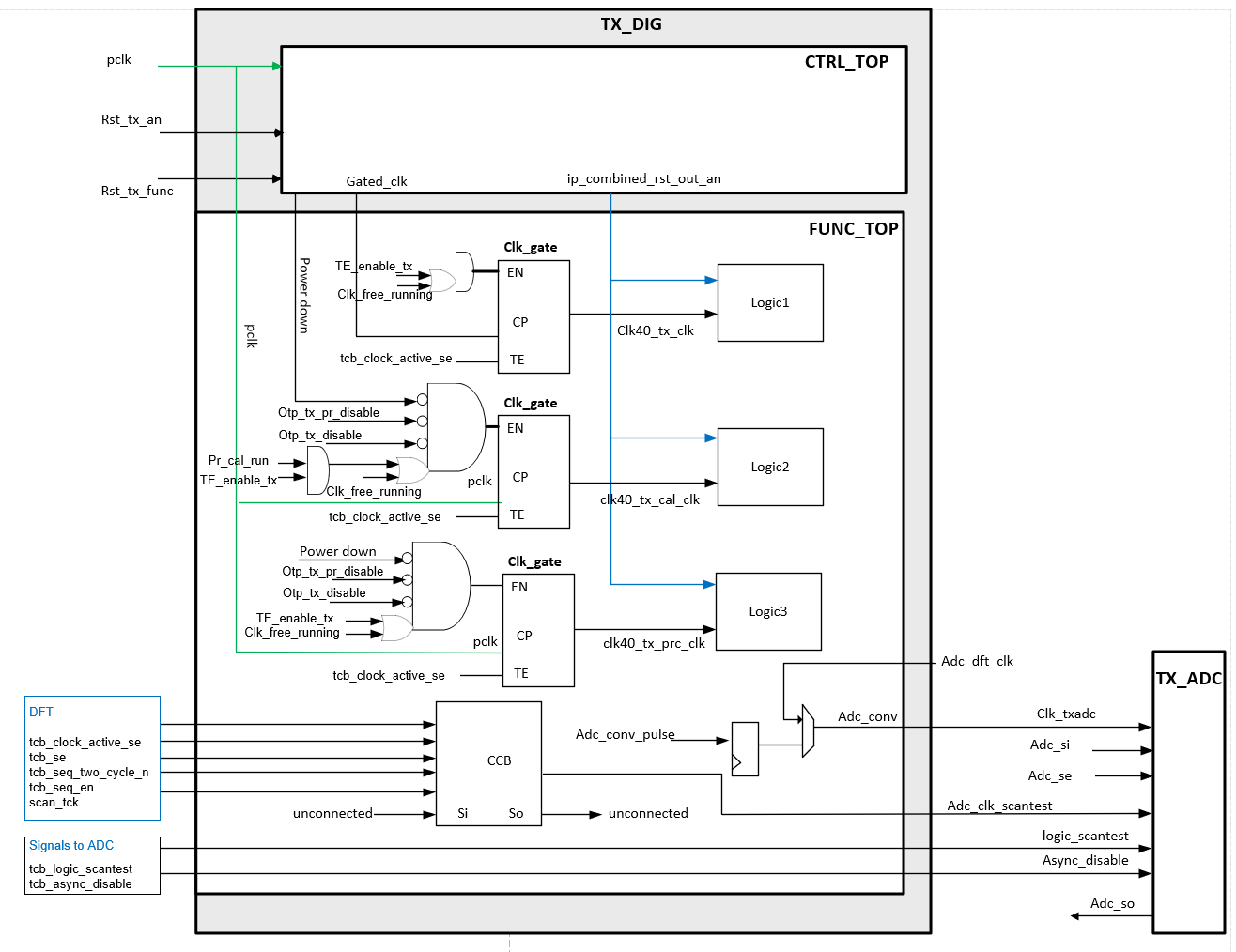


Figure 4: Clock and reset

# TX mode of operations

TXDIG has 3 modes of operations

1. **Application mode**: Phase is transmitted from the timing engine to the phase rotator
2. **Phase rotator calibration mode**: The Phase rotator is calibrated and self-compensated
3. **Safety mode**: 7 safety mechanisms guaranty the behavior of the transmitter

Tx is driven by an internal FSM. The application mode and the safety mode shall operate when the FSM is in IDLE.

All others state than IDLE shall be dedicated to the Phase rotator calibration and shall be depicted chapter 7.3.

# TX application mode

The application mode correspond to the IDLE state of the FSM

The main functionality of the TXDIG is to translate the 7 bits phase from the timing engine to 256 unitary weighted bit input of the TX I and Q phase rotator DAC. The translation is done through a Cordic algorithm and a thermometer encoder.

## Tx phase

During the application mode, the phase is a 7 bits signal from the Timing Engine module. The rotation in degree shall be defined by  TE\_phase\_code[6:0]\*360/2^7 where TE\_phase code is a SPI bits within Timing Engine digital module.

A debug mode exists where the phase is directly driven by a register d\_force\_te\_phase in the TX. The phase d\_force\_te\_phase is a 7 bits SPI word that apply the phase at cordic input when SPI bit d\_force\_te\_phase\_sel =1

The rotation in degree shall be defined with d\_force\_te\_phase[6:0]\*360/2^7.

Table 2: Phase binary

|  |  |  |
| --- | --- | --- |
| phase ⁰ | Phase decimal | Phase binary |
| 0 | 0 | 0000000 |
| 45 | 16 | 0010000 |
| 90 | 32 | 0100000 |
| 135 | 48 | 0110000 |
| 180 | 64 | 1000000 |
| 225 | 80 | 1010000 |
| 270 | 96 | 1100000 |
| 315 | 112 | 1110000 |

A thermometer encoder shall convert signed bits to unary binary code to control PR DACs. The thermometer encoder input shall be thermin\_i and thermin\_q respectively for I and Q path. Both thermin\_i and themin\_q shall be 9 signed bit format

* case thermin\_i/q > or = 128 the output shall be clamp to 256 ‘0’ unary bits
* case thermin\_i/q = 127 the output shall be 255 consecutive bits at ‘0’ , 1bit @ ‘1’
* case thermin\_i/q = 126 the output shall be 254 consecutive bits at ‘0’ , 2 consecutives bit @ ‘1’
* case thermin\_i/q = 125 the output shall be 253 consecutive bits at ‘0’ , 3 consecutives bit @ ‘1’
* .
* case thermin\_i/q = -126 the output shall be 2 consecutive bits at ‘0’ , 254 consecutives bit @ ‘1’
* case thermin\_i/q = -127 the output shall be 1 bits at ‘0’ , 255 consecutives bit @ ‘1’
* case thermin\_i/q = -128 the output shall be 0 bits at ‘0’ , 256 consecutives bit @ ‘1’

all intermediate code (from 124 to -125) shall follow the same rules "

## TX phase init

The TX phase init shall be programmable using SPI bit d\_phi\_init\_ci and d\_phi\_init\_cq.

## TX gain

The TX gain shall be programmed through several bits from GLDO to TX profile bits. The gain shall also be controlled through GAIN\_INIT which is 13-bit SPI that controls the amplitude of PR DAC input signal. By default the gain is programmed 0.2dB below the DAC saturation.

## TX profile

TX is using 9 profiles.

TXDIG profile are managed by the TXDIG register bits:

Table 3: TX profile content

|  |  |  |
| --- | --- | --- |
| LOX2\_DAC\_CTRL |  | LO POWER: Dac control of the LOX2(buf1B) (PR input) |
| ST1\_C |  | PA POWER :DAC current control PA first stage |
| ST2\_C |  | PA POWER :DAC current control PA second stage |
| ST1\_V |  | PA POWER :DAC voltage control PA first stage |
| ST2\_V |  | PA POWER :DAC voltage control PA second stage |
| DAC\_SFTY\_LEVEL |  | PA\_SAFETY:DAC control for safety level settings |
|  |  |  |

# Phase rotator calibration

## Introduction

The goal of the calibration is to ensure that the phase rotator has +-2.8 degrees phase accuracy as required Req: 1098668 of the TX Requirement specifications.

The calibration can be done through an autonomous mode or through a software mode. The autonomous mode shall be initiated through the pr\_cal\_start signal coming from TE or SPI or OCII. In case of SW calibration no start are needed, and steps of the calibration shall be individually done through SPI access.

## Block diagram

The calibration is a closed loop starting at the phase rotator output including the first buffer(BUF2A). A peak power detector measures the amplitude of the RF signal, which is then amplified through a first VGA , then a second VGA then an ADC. The second VGA is calibrated using 2 DACs, to optimize the signal according to the ADC Full-scale .The digital data is used to take decision and digital compensation are applied to improve the phase rotator performance

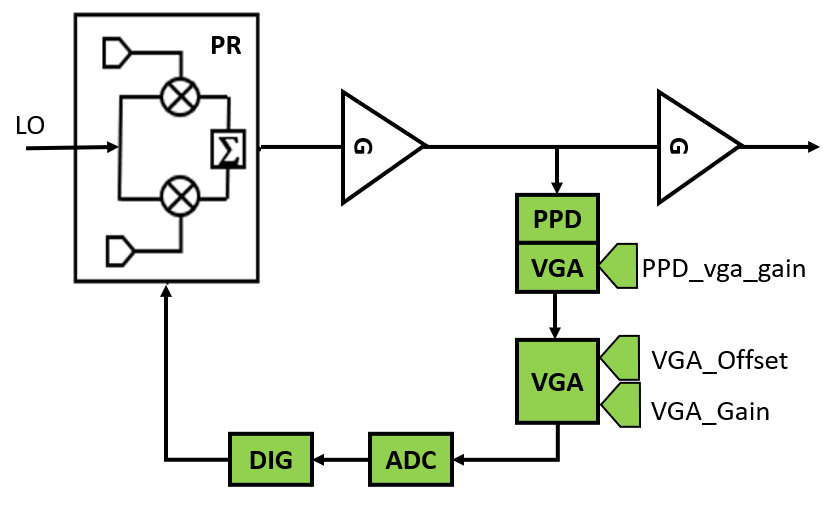


Figure 5:PR calibration block diagram

PPD VGA is used for phase rotator calibration and also for other measurement such as TX power calibration. During Phase rotator calibration the PPD\_VGA gain is controlled by SPI bit VGA\_PPD\_PHASE. Outside from the PR calibration, the PPD VGA gain is controlled by SPI bit VGA\_PPD\_POWER. The PPD VGA gain shall be forced through VGA\_PPD\_GAIN\_FORCE when VGA\_PPD\_GAIN\_FORCE\_SEL=1

The second VGA shall be Gain and offset controlled. The VGA shall only be used during the Phase rotator calibration.

The offset of the second VGA shall be defined during the PR autonomous calibration . the offset shall nevertheless be force by SPI bits VGA\_OFFSET\_FORCE when VGA\_OFFSET\_FORCE\_SEL=1

The gain of the second VGA shall be updated along the Phase rotator calibration. When in VGA\_INIT state the gain shall be controlled by VGA\_INIT\_VGA\_GAIN. The gain shall be updated autonomously in VGA\_Gain state. the gain shall nevertheless be forced by SPI bits VGA\_GAIN\_FORCE when VGA\_GAIN\_FORCE\_SEL=1

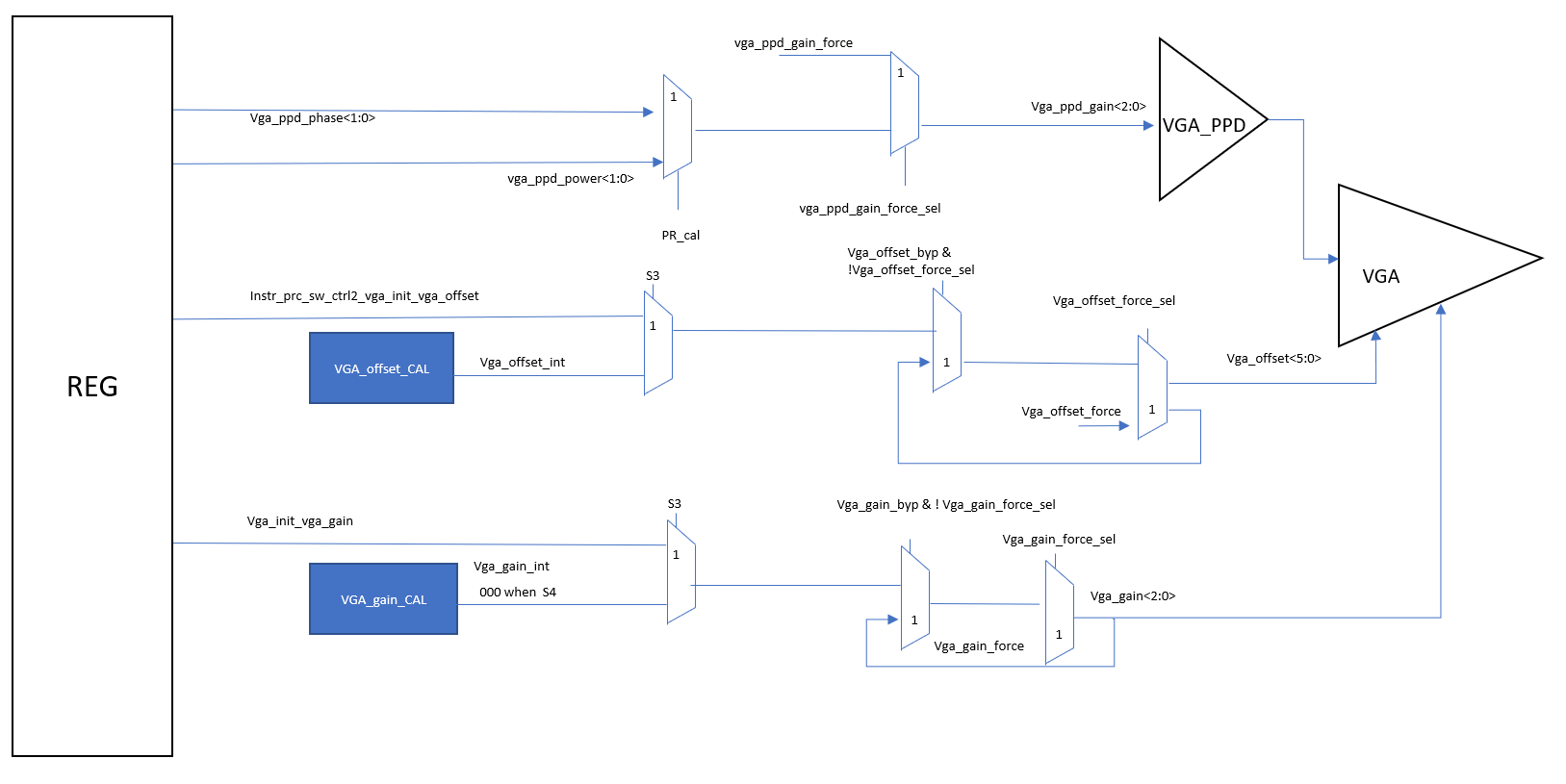


Figure 6:VGA control on PR calibration path

## Calibration FSM states

The Phase rotator calibration is driven by an internal FSM. The calibration needs 10 states to work properly. Each of the state can be bypassed. In case the state is bypassed the result of the calibration is unchanged.

The calibration starts when the timing engine activate the start\_pr\_cal signal. The start of the calibration can also be done through SPI and OCI. The calibration start is managed by SPI start\_cal\_force when SPI bit starts\_cal\_force\_sel=1

At the end of the calibration a signal end\_pr\_cal indicates to the timing engine than the calibration is over.

A SPI bit “pr\_cal\_run” indicate that the calibration is in progress.

Diagram

Description automatically generated

Figure 7: PR calibration FSM

### Start\_CAL FSM state

#### Overview

This state allows to power up the analog module before running the calibration. Some of the analog and RF module may need some time to wake up. This time is programmable using SPI bit “analog\_startup\_timer”

Analog\_startup\_timer shall be programmed by step of 25ns

Analog\_startup\_timer shall be programmed accordingly to the needed analog power up time of the analog and RF module on the Phase rotator calibration path.

#### State bypass

There Is no bypass for this state, in case no need to use this timer to power up analog, it is recommended to program the “analog\_startup\_timer” to the minimum value

#### State duration

The state duration shall be 25ns\*(Analog\_startup\_timer+2)

### ADC\_AVE FSM state

#### Overview

ADC\_AVE is a state of the FSM during which a calibration is done .

ADC averaging consists in determining the number of ADC measurement "cal\_out\_rd" needed to get an accurate ADC measurement.

The averaging is needed to decrease the noise contribution of the RF chain.

The result of the calibration is reported with cal\_out\_rd SPI register.

The result of the calibration shall be 1, 2, 4, 8 or 16 which indicate the number of needed ADC measurements.

Other states of the FSM related to the phase rotator calibration shall perform "av\_cal\_out\_rd" number of ADC measurement each time a measurement is needed.

By default "ADC\_AVE" state is bypassed to avoid variation in the calibration time along the radar process. The default value is 8.

By default this state is bypassed and the number of ADC measurement averaging is 8. The state is bypass to guaranty a constant time of the full calibration

#### Calibration forced result

The calibration result shall be forced when SPI bit av\_force\_sel=1 .

When av\_force\_sel=1 the result of the calibration is programmed thought SPI register av\_calout\_force

#### State bypass

This calibration shall be bypassed when SPI bit byp\_ppd\_averaging=1.by default the state is bypass

When the calibration is bypassed the result of the calibration is unchanged compare to what it was before entering the "ADC\_AVE" state under the condition that the calibration is not forced.

#### State duration

When not bypassed ,the state duration shall

Duration=(timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*16\*25+300

When bypassed the state duration shall be 25ns

### VGA\_INIT FSM state

#### Overview

"“VGA\_INIT” is a state of the FSM where 16 phases codes are swept, and ADC output are processed. VGA\_INIT is a state during which 3 input phases values are defined:

* vga\_offset\_cal\_phi\_code used in next VGA\_OFFSET state
* vga\_gain\_cal\_phi\_code\_ripple\_max used in next VGA\_OFFSET state
* vga\_gain\_cal\_phi\_code\_ripple\_min used in next VGA\_OFFSET state

vga\_offset\_cal\_phi\_code shall be readable through vga\_phi\_code\_rd SPI bit.

vga\_gain\_cal\_phi\_code\_ripple\_max shall be readable through vga\_phi\_code\_max\_rd SPI bit

vga\_gain\_cal\_phi\_code\_ripple\_min shall be readable through vga\_phi\_code\_min\_rd SPI bit"

The ripple term corresponds to the variation of the ADC during VGA\_INIT when sweeping the 16 phases code.

When the state machine is stop at the end of VGA\_INIT state the 16 ADC outputs shall be accessible using SPI read bit ppd\_meas1, ppd\_meas2, ppd\_meas3, … ppd\_meas16

vga\_phi\_code\_rd shall correspond to 1 of the 16 phase code that have been swept

vga\_phi\_code\_rd shall correspond to the phase code that give the ADC value the closest to the mid ripple during VGA\_INIT

vga\_phi\_code\_min\_rd shall correspond to 1 of the 16 phase code that have been swept

vga\_phi\_code\_min\_rd shall correspond to the phase code that give the ADC min value during VGA\_INIT state

vga\_phi\_code\_max\_rd shall correspond to 1 of the 16 phase code that have been swept

vga\_phi\_code\_max\_rd shall correspond to the phase code that give the ADC max value during VGA\_INIT state

ripple\_min\_rd and ripple\_max\_rd shall give the ADC output corresponding respectively to vga\_phi\_code\_min\_rd and vga\_phi\_code\_min\_rd

vga\_init\_vga\_offset SPI bits shall control the offset of the DAC of the VGA during VGA\_INIT state

vga\_init\_vga\_gain SPI bits shall control the gain of the DAC of the VGA during VGA\_INIT state

vga\_ppd\_phase SPI bits shall control the gain of the VGA\_PPD during VGA\_init state

vga\_init\_vga\_offset , vga\_init\_vga\_gain , vga\_ppd\_phase shall be programmed so that ADC do not clamp during VGA\_INIT state

#### Calibration forced result

vga\_offset\_cal\_phi\_code shall be overwritten by SPI bit vga\_offset\_cal\_phi\_code\_force when SPI bit vga\_offset\_cal\_phi\_code\_sel shall be set to 1. The applied phase on CORDIC shall be (vga\_offset\_cal\_phi\_code\_force\*2^10)\*360/2^16 degrees.

#### State bypass

The state shall be bypass when SPI bit byp\_vga\_init=1

#### State duration

When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25 +675)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\* calout\_rd \*25+675)\*15

When bypassed the state duration shall be 25ns

### VGA\_Offset FSM state

#### Overview

“VGA\_offset\_cal” is a state of the FSM during which a calibration is done . Aim of the calibration is to find optimized offset settings of the second VGA (pr\_vga\_offset interface signal) so that the ADC is used in an efficient way.

Once the vga\_offset calibration is over a second action shall done during VGA\_OFFSET state. it consists in determining the phase code vga\_gain\_cal\_phi\_code that shall be used during next state VGA\_GAIN state.

Result of the calibration shall be stored in vga\_offset\_rd and shall cover all values at least in a range [-d12:d12] LSB

At the end of the “vga\_offset\_cal” state, the ADC measurement is stored within SPI status bit meas\_vga\_offset\_rd.

The meas\_vga\_offset\_rd measurement shall be close to ADC mid value of 1023 LSB at the end of the “vga\_offset\_cal” state.

During "VGA\_offset\_cal" shall also be determined the phase (vga\_gain\_cal\_phi\_code) that shall be used during "vga\_gain\_cal" FSM state

Vga\_gain\_cal\_phi\_code shall be readable through vga\_gain\_cal\_phi\_code\_rd SPI bit

Vga\_gain\_cal\_phi\_code shall be one of the 16 phases code used during the vga\_init state calibration

Vga\_gain\_cal\_phi\_code shall be computed during vga\_offset\_cal state using results vga\_phi\_code\_max\_rd and vga\_phi\_code\_min\_rd from VGA\_GAIN\_INIT state. Vga\_gain\_cal\_phi\_code shall be the phase corresponding to the biggest ADC output variation compared to 1023 LSB(middle ADC code).

Vga\_gain\_cal\_phi\_code = ADC\_output( max ( adcout\_output( vga\_gain\_cal\_phi\_code\_max\_rd -1023, 1023-vga\_gain\_cal\_phi\_code\_min\_rd ))

#### Calibration forced result

vga\_gain\_cal\_phi\_code shall be overwritten by SPI bit vga\_gain\_cal\_phi\_code\_force when SPI bit vga\_gain\_cal\_phi\_code\_sel shall be set to 1. The applied phase on CORDIC shall be (vga\_gain\_cal\_phi\_code\_force\*2^10)\*360/2^16 degrees.

vga\_offset calibration result shall be overwritten by SPI bit vga\_offset\_force when SPI bit

#### State bypass

The calibration shall be bypassed when SPI bit byp\_vga\_offset=1

#### State duration

The "VGA\_offset" calibration duration shall be constant.

When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\* calout\_rd \*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\* calout\_rd \*25+300)\*7

### VGA\_Gain FSM state

#### Overview

“VGA\_gain\_cal” is a state of the FSM during which a calibration is done. Aim of the calibration is to find optimized Gain settings of the VGA (vga\_gain\_ctrl) so that the ADC is used in an efficient way.

The intend of the calibration is to set the output of the ADC within a defined range using information from vga\_gain\_cal\_phi\_code\_ripple\_max and vga\_gain\_cal\_phi\_code\_ripple\_min from VGA\_OFFSET state.

Result of the calibration shall be stored in vga\_gain\_rd.

The VGA gain shall amplify the corresponding PPD output to values close to the ADC full-scale

Vga\_gain\_min\_th SPI bit shall define the ADC output low limit for the VGA gain calibration

Vga\_gain\_max\_th SPI bit shall define the ADC output high limit for the VGA gain calibration

Vga\_gain\_max\_th shall be programmed greater than Vga\_gain\_min\_th

in case adcout\_output( vga\_gain\_cal\_phi\_code\_ripple\_max -1023) > adcout\_output( 1023- vga\_gain\_cal\_phi\_code\_ripple\_max ) then gain of the calibration shall be higher as possible but ADC output shall not exceed Vga\_gain\_max\_th

in case adcout\_output( vga\_gain\_cal\_phi\_code\_ripple\_max -1023) < adcout\_output( 1023- vga\_gain\_cal\_phi\_code\_ripple\_max ) then gain of the calibration shall be higher as possible but ADC\_output shall always exceed Vga\_gain\_min\_th

The maximum ADC range is defined by SPI bit Vga\_gain\_min\_th, Vga\_gain\_max\_th .

In STRX ES1 a margin of 272 LSB has been taken as the analog part is clamping on both low and high level. As consequence ¼ of the ADC full-scale cannot be used.

#### Calibration forced result

The result of the VGA calibration shall be overwritten by the vga\_gain\_force SPI register when the vga\_gain\_force\_sel SPI bit is '1'.

#### State bypass

The state shall be bypassed when SPI bit byp\_vga\_gain=1

#### State duration

When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25+300)\*3"

When bypassed the state duration shall be 25ns

### PR\_OFFSET\_CAL FSM state

#### Overview

PR\_offset\_cal” is a state of the FSM during which a calibration is done . Aim of the calibration is to compensate the offset between I and Q path . Offset may come from the DAC , mixers, branch line coupler …The compensation is digitally applied at the DAC inputs.

A mechanism has been added to avoid the saturation of the DACs .

In case the sum of the offset compensation and cordic output do not belong to [-128: 128] LSB , the gain of the cordic shall be automatically decrease so that the adder output adder\_out\_i and adder\_out\_q shall always belong to [-128:128] LSB"

Result of the calibration are stored in SPI register offset\_cal\_ioffset\_code\_rd, offset\_cal\_qoffset\_code\_rd , offset\_cal\_Igain\_code\_rd, offset\_cal\_Igain\_code\_rd

.

#### Calibration forced result

result of the calibration, offset\_cal\_Ioffset\_code shall be overwritten by the offset\_Ioffset\_code \_force SPI register when the PPD\_offset \_force\_sel SPI bit is '1'

result of the calibration, offset\_cal\_Igain\_code shall be overwritten by the offset\_Igain\_code\_force SPI register when the PPD\_offset \_force\_sel SPI bit is '1'

result of the calibration offset\_cal\_Qoffset\_code shall be overwritten by the offset\_Qoffset\_code\_force SPI register when the PPD\_offset \_force\_sel SPI bit is '1'.

result of the calibration offset\_cal\_Igain\_code shall be overwritten by the offset\_Igain\_code \_force SPI register when the PPD\_offset \_force\_sel SPI bit is '1'.

OCAL\_PPD0\_RD, OCAL\_PPD90\_RD, OCAL\_PPD180\_RD, OCAL\_PPD270\_RD SPI bit shall be used to observe the convergence of the binary search. At the end of the Offset calibration OCAL\_PPD0\_RD shall be close to OCAL\_PPD180\_RD while OCAL\_PPD90\_RD shall be close to OCAL\_PPD270\_RD

#### State bypass

The state shall be bypass when SPI bit byp\_pr\_offset

#### State duration

"When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25+300)\*27"

When bypassed the state duration shall be 25ns

### GPATH\_SEL FSM state

#### Overview

“GPATH\_SEL “ is a state of the FSM during which a calibration is done. The goal is to determine which of I and Q path has the highest gain and will be used for gain compensation.

The calibration output hall be stored in gpath\_sel\_rd

When gpath\_sel\_rd ="01" , the phase and gain imbalance of the I path shall be compensated, when gpath\_sel\_rd="10" the phase and gain imbalance of the Q path shall be compensated

If the RF Phase Rotator Gain imbalance (ratio Q/I) is > 0 the gain compensation shall be applied on Q during the PR\_GAIN\_CAL.

If the RF Phase Rotator Gain imbalance (ratio Q/I) is < 0 the gain compensation shall be applied on I during the PR\_GAIN\_CAL.

ppd measurement when phase is 0 and 90 degrees shall be respectively read back with SPI bit gpath\_sel\_ppd0\_rd, gpath\_sel\_ppd90\_rd

#### Calibration forced result

Result of the calibration shall be overwritten by gpath\_sel\_force SPI bits when gpath\_sel\_force\_sel = 1

#### State bypass

The state shall be bypassed when SPI bit byp\_gpath\_sel=1

#### State duration

When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25+300)\*1

When bypassed the state duration shall be 25ns

### PR\_GAIN\_CAL FSM state

#### Overview

PR\_gain\_cal is a state of the FSM during which a calibration is done . Aim of the calibrations is to compensate the gain imbalance between I and Q path. Gain imbalance may come from the DAC or PR.

Result of the calibration shall be stored in gain\_comp\_i\_rd and gain\_comp\_q\_rd which are respectively the compensation gain of the I and Q path

For better algorithm efficiency of the calibration , the binary search shall start using a programmable gain value. This value shall be defined by SPI bit Gain\_cal\_init . by default Gain\_cal\_init is 2.5dB lower than gain\_init which allow to look for gain imbalance compensation up to 2.5dB.

#### Calibration forced result

The d\_gain\_comp\_i  shall be overwritten by the gain\_comp\_force\_i  SPI register when the force\_gain\_comp\_i\_sel  SPI bit is '1'.

The d\_gain\_comp\_q shall be overwritten by the gain\_comp\_force\_q SPI register when the force\_gain\_comp\_q\_sel SPI bit is '1'.

#### State bypass

The state shall be bypass when SPI bit byp\_pr\_gain=1

#### State duration

"When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\* calout\_rd \*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\* calout\_rd \*25+300)\*14"

When bypassed the state duration shall be 25ns

### PR\_PHASE\_CAL FSM state

#### Overview

PR\_phase\_cal is a state of the FSM during which a calibration is done . Aim of the calibrations is to compensate the phase imbalance between I and Q path. Phase imbalance may come from the DAC or PR.

result of the calibration shall be stored in phase\_comp\_i\_rd and phase\_comp\_q\_rd SPI register which value shall be added respectively to the phase of the I and Q path.

The path to be compensated shall be defined by SPI bit phase\_branch\_sel . By default the Q path shall be compensated in phase

#### Calibration forced result

The phase\_comp\_i\_rd shall be overwritten by the phase\_comp\_force\_I SPI register when the phase\_comp\_i\_sel SPI bit is '1'

The phase\_comp\_q\_rd shall be overwritten by the phase\_comp\_force\_q SPI register when the phase\_comp\_q\_sel SPI bit is '1'.

#### State bypass

The state shall be bypassed when SPI bit byp\_pr\_phase=1

#### State duration

When not bypassed ,the state duration shall

Duration=((timer\_cdic\_conv\_ready+2)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25 +300)+((timer\_cdic\_conv\_ready+1)\*25 +(timer\_adc\_cycle+1)\*calout\_rd\*25+300)\*13

When bypassed the state duration shall be 25ns

### END\_CAL

End\_cal state is a state where the signal end\_pr\_cal is generated. This signal indicate to the timing engine than the overall calibration is over.

#### State duration

The state is 75ns.

## Automatic calibration

The self-autonomous calibration is initiated when start\_pr\_cal signal is 1.

Start\_pr\_cal is initiated by:

* The Timing engine
* Through SPI: set start\_cal\_force to 1 when SPI bit start\_cal\_force\_sel=1
* Through OCI.

## Manual calibration

The manual calibration shall be performed in IDLE mode of the Local TX FSM.

A manual calibration shall be exercised by forcing the Offset , Gain imbalance , Phase imbalance compensation and the Phase rotator DAC inputs.

### Phase rotator DACs input force

The DACs inputs shall be controlled by either forcing the thermometer encoder inputs or more easily by forcing the CORDIC input.

CORDIC has 3 input parameter, phase(phi\_code\_i/q), phase\_init(d\_phi\_init\_i/q), gain(d\_gain\_i/q) which all can be forced.

CORDIC\_I output is cdic\_out\_i signal

Cordic output shall be described by cdic\_out\_i = [d\_gain\_i / 2^6]\*sin( ( phi\_code\_i+ d\_phi\_init\_i)\*2\*pi/2^16) whatever the state of the FSM is.

CORDIC\_Q output is cdic\_out\_q signal.

Cordic output shall be described by cdic\_out\_i = [d\_gain\_i / 2^6]\*sin( ( phi\_code\_i+ d\_phi\_init\_i)\*2\*pi/2^16) whatever the state of the FSM is.

d\_gain\_i/q, phi\_code\_i/q, d\_phi\_init\_i/q shall be forced through SPI as described on diagram.

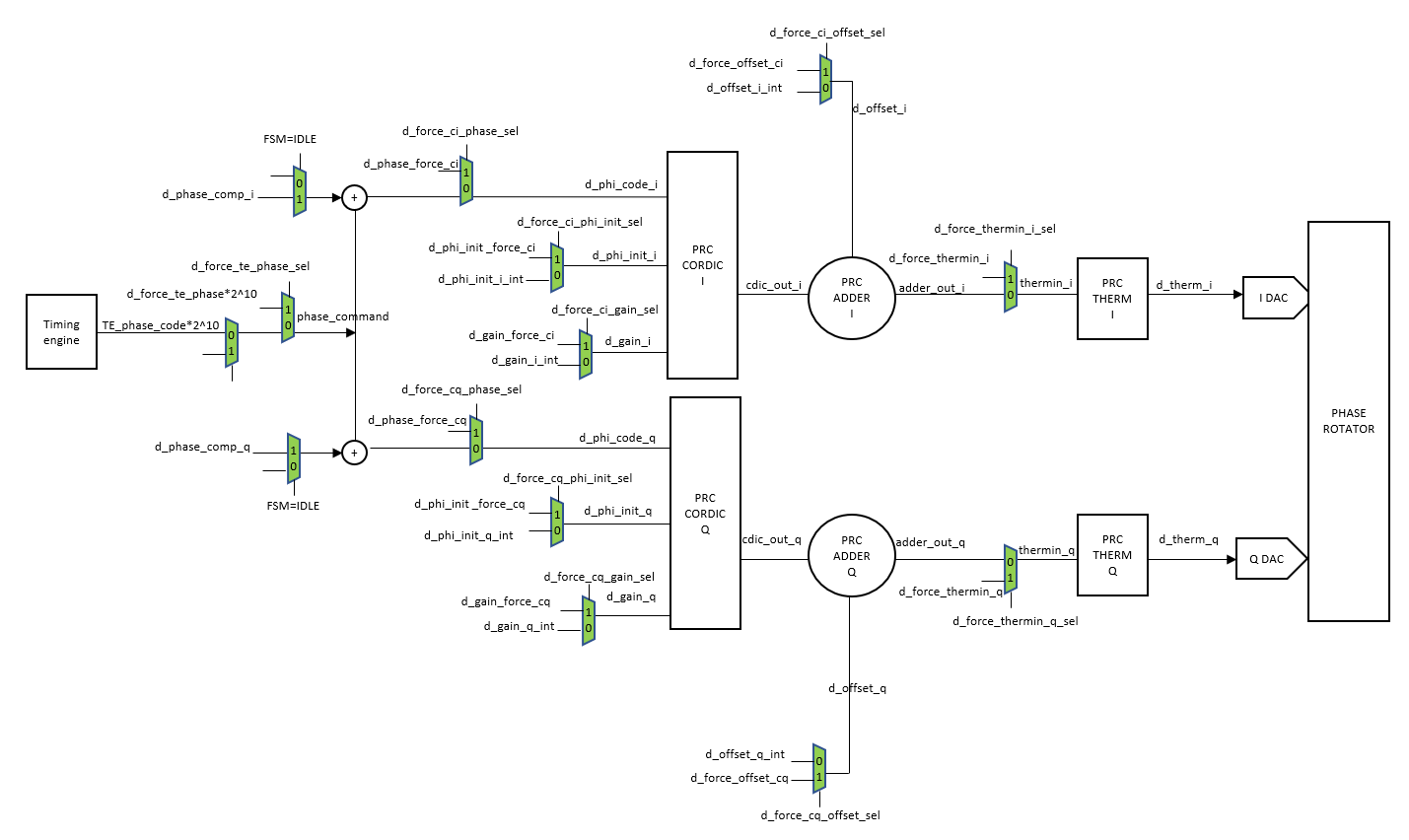


Figure 8: TXDIG forced signals

### Compensation force

All compensation results shall be forced by SPI bits . Description of the forced compensation has been described in chapter 7.3.2.2 , 7.3.3.2, 7.3.4.2, 7.3.5.2, 7.3.6.2, 7.3.7.2, 7.3.8.2, 7.3.9.2

## Calibration time

The Phase rotator calibration time shall be programmable.

SPI bit analog\_startup shall define the time to power up analog module in case they are not.

SPI bit timer\_cdic\_conv\_ready shall define the time need to pass the phase through the digital

SPI bit timer\_adc\_cycle shall define the clock period of the measurement.(so also the ADC clock period)

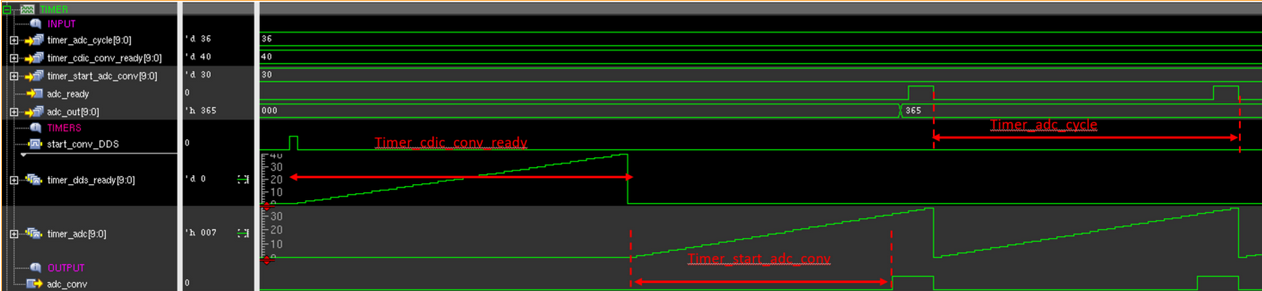


Figure 9: Calibration time parameter

The calibration time shall be defined using the document [STRX\_calibration\_time](https://www.collabnet.nxp.com/sf/docman/do/downloadDocument/projects.smarttrx/docman.root.es0.011_ic_architecture_design_and_i.ip_design.tx/doc500060/1)

A detailed description of the calibration time per state of the FSM is described in the FSM chapter

# TX safety

Safety of the TX is managed through software during IDLE state of the TX FSM

## Real time monitoring

### RTM description

This safety mechanism shall be done inside TX hardware.

A Realtime power monitoring is implemented in the transmitter to fulfill the safety requirement at system level. The error flag is sent to the inner safety monitor. The architecture is a mixed architecture with on analog side a power detector a VGA, a 6 bits reference voltage DAC to set the safety threshold and a comparator and with on digital side a mechanism to post process the comparator output. The post processing of the comparator output is done by a FIR. The latch is implemented in the digital . The FIR shall average the noise from the PPD/VGA.

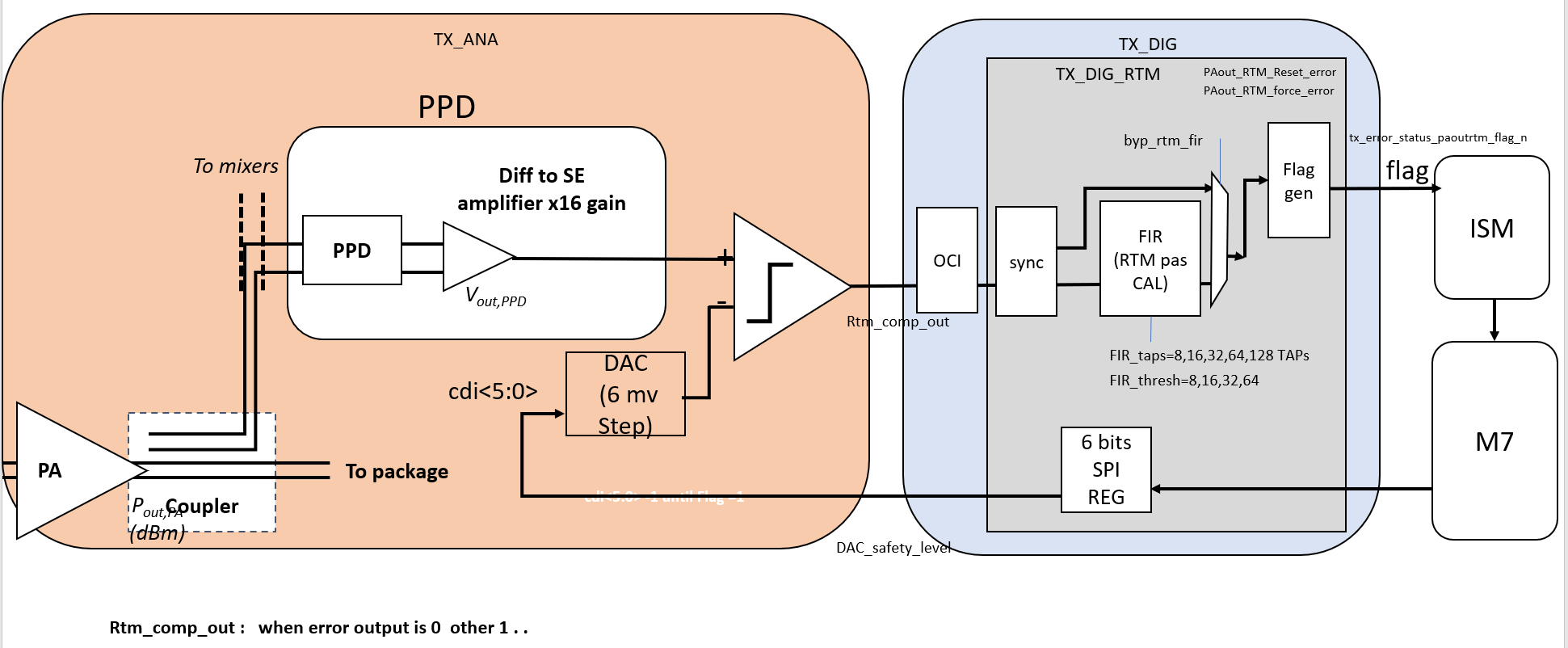


Figure 10: Real time monitoring block diagram

FIR\_TAPS shall be programmable and shall define the number of comparator outputs (sampled at 40MHz) that shall be summed for Flag generation

FIR\_TRESH shall be programmable and shall defined the threshold for safety flag generation

TX\_DIG\_RTM module shall be a TX\_DIG submodule

Aim of the TX\_DIG\_RTM shall be to process the data coming from the analog comparator at PPD output as sheet "PAOUT RTM"

The threshold shall be calibrated by SW prior to start the safety mechanism .

Threshold calibration:

• The threshold is calibrated in SW.

• THE FIR shall be bypassed byp\_fir=1

• PA shall be programmed 3dB lower than expected power.

• DAC current shall be set by SW to its max value and decrease until comparator output change.

The digital safety mechanism (post process of the comparator output) shall be enabled through EN\_RTM SPI bit.

The analog signal d\_realtime\_mon\_error is the comparator output and shall be resampled on a 40Mhz clock in TXDIG.

The FIR inside the safety mechanism shall be bypassed by settings byp\_fir=1

When byp\_fir=1 the comparator output is not averaged.

When byp\_fir=1 the latency time between the analog comparator output d\_realtime\_mon\_error and tx\_paout\_rtm\_error\_flag\_n shall have a minimum value between 0 and 1 clock cycles

When byp\_fir=0 the latency time between the analog comparator output d\_realtime\_mon\_error and tx\_paout\_rtm\_error\_flag\_n shall have a minimum value between (FIR\_TAPS+FIR\_THRESH) and (FIR\_TAPS+FIR\_THRESH+1) clock cycles

The FIR output shall be the sum of the N successive FIR input bits where N is defined by the FIR\_TAPs

The TX PAout RTM safety DAC shall be controlled by SPI bit DAC\_sfty\_level

A flag tx\_paout\_rtm\_error\_flag\_n shall be asserted ( tx\_paout\_rtm\_error\_flag\_n=0) when the sum of FIR\_TAPS comparator outputs shall be greater or equal to FIR\_thresh programmed value after a defined latency time.

The Paout RTM flag tx\_paout\_rtm\_error\_flag\_n shall be reported in PAOUT\_RTM\_ERR\_FLAG SPI bit

The Paout RTM flag shall be send to the ISM

### RTM Mask

Register MASK\_ISM\_FORCE\_ERROR/ BBD\_ERR masks the force\_error generated by the SW towards the ISM.

Register MASK\_ISM\_RESET\_ERROR/BBD\_ERR masks the reset\_error generated by the SW towards the ISM

Register MASK\_ISM\_ERROR\_FLAG/BBD\_ERR masks the flag and that not be forwarded to the ISM

### RTM FIT

The PAoutput force error shall not force the safety flag directly. the PAoutput force error signal shall force the threshold of the analog comparator to its highest value. The power shall be then below this threshold and then a flag shall be generated in the analog module

The PAoutput reset error shall force the flag and also the level of the threshold of the analog comparator to its lower value to be sure the flag is not re-asserted automatically

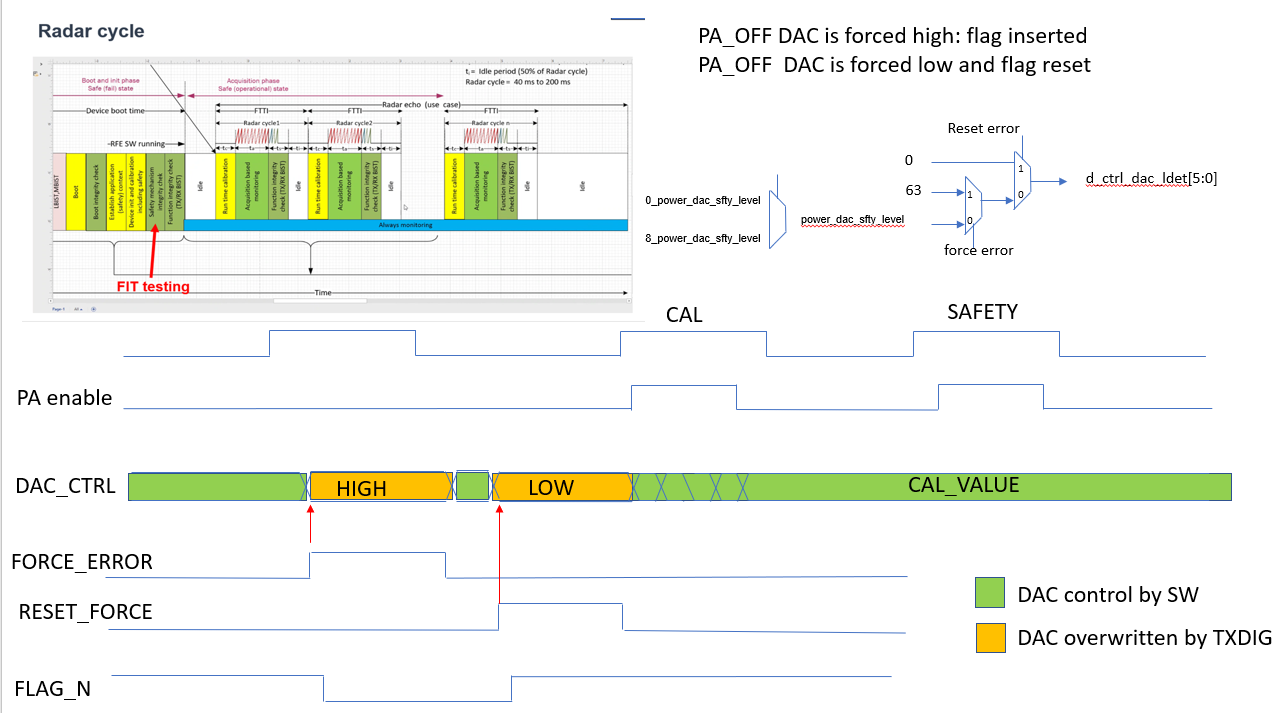
The force and reset of the PA output power safety mechanism is depicted on the link

Figure 11: RTM FIT sequence

Fusa related SPI bits within register mask\_ism\_force\_error, mask\_ism\_reset\_error, mask\_ism\_error\_flag, test\_force\_error, test\_reset\_error, error\_status and associated mechanism shall follow requirement from the CTRL\_TOP module

Force\_error, reset error shall operate as described in the CTRL\_TOP generic module

## Ball break detector

This safety mechanism shall be done inside TX hardware.

### BBD description

A ballbreak detector is implemented in the transmitter to fulfill the safety requirement at system level.

The BBD is used for safety purpose in order to check the state of the solder ball contacting the package and the PCB. Correct operation of the BBD requires that the antenna at PCB side should shorted to ground through quarter-wavelength stub. The BBD is composed with two comparators, current sources and control logic. The current sources push a DC current on both differential paths of the TX (RF\_P & RF\_N) to reach the ground of the quarter-wavelength stub passing through a balun implemented in the package to convert the differential signal from the TX output to a single ended at antenna reference plane. During its flowing, the DC current will interact with the ball giving dropped voltage whose value is depending on the state of the solder ball. This value is then compared using the two compactors, the positive input of each comparator being connected to a given RF path (RF\_P or RF\_N). The negative input of the comparators is driven by a reference voltage generated internally by the BBD. The output of the comparators is combined through the logic control to provide the output flag of the BBD. When the solder ball is broken the flag will be low logic state, whereas, for normal operation the flag still at high state. The flag is latched internally by the logic control, therefore, a reset signal could be applied to the BBD in order to erase the former state.

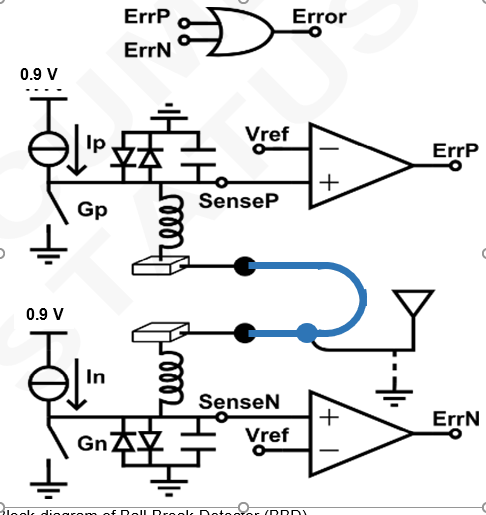


Figure 12: Ball break detector block diagram

The error flag is sent to the inner safety monitor.

A SPI bit BBD\_err\_flag shall be asserted ( bbd\_error\_flag\_n=0) when ball break is happening

### BBD Mask

Register MASK\_ISM\_FORCE\_ERROR/ BBD\_ERR masks the force\_error generated by the SW towards the ISM.

Register MASK\_ISM\_RESET\_ERROR/BBD\_ERR masks the reset\_error generated by the SW towards the ISM

Register MASK\_ISM\_ERROR\_FLAG/BBD\_ERR masks the flag and that shall not be forwarded to the ISM

### BBD FIT

Fault injection test shall be introduced in the safety BBD mechanism by activating SPI bit TEST\_FORCE\_ERROR/ BBD\_ERR\_FORCE. When BBD\_ERR\_FORCE=1,

The principle of the fault injection is to create a real fault inside the BBD. By setting SPI bit bbd\_err\_force=1, the positive input of both comparators becomes higher than the negative input (driven by the reference voltage), therefore, the comparator output is high leading to flag the BBD output (low logic state).

## CRC REGISTER

### CRC REGISTER description

Those safety mechanism shall be done inside TX hardware.

CRC Register safety check guaranty the content of the register along the time. It prevent from a bit flip by example.

A CRC reference code is generated using the internal TXDIG CRC engine at reset and each time a new SPI write is done. When CRC\_CHECK\_TR\_CTL=1, TXDIG CRC engine is generating a second code which is compared to the reference code.

A safety flag is asserted when the reference value is different from the newly calculated value indicating a flip of one or several bits.

The flag is located in SPI register ERROR\_STATUS/CRC\_ERR\_FLAG

### CRC Mask

Register MASK\_ISM\_FORCE\_ERROR/ CRC\_ERR masks the force\_error generated by the SW towards the ISM.

Register MASK\_ISM\_RESET\_ERROR/CRC\_ERR masks the reset\_error generated by the SW towards the ISM

Register MASK\_ISM\_ERROR\_FLAG/CRC\_ERR masks the flag and that shall not be forwarded to the ISM

## CRC MISO

### CRC MISO description

CRC MISO safety check guaranty the write SPI communication protocol. M7 is generating a CRC reference code which is send to through the SPI communication to TXDIG. TXDIG decode the SPI protocol and compute a CRC code using its internal CRC engine.

A safety flag is asserted when the reference value is different from the newly calculated value indicating a wrong transmission of the data through the SP write.

The flag is located in SPI register ERROR\_STATUS/MISO\_CRC\_ERR.

### CRC MISO MASK

Register MASK\_ISM\_FORCE\_ERROR/ MISO\_CRC\_ERR masks the force\_error generated by the SW towards the ISM.

Register MASK\_ISM\_RESET\_ERROR/MISO\_CRC\_ERR masks the reset\_error generated by the SW towards the ISM.

Register MASK\_ISM\_ERROR\_FLAG/MIOS\_CRC\_ERR masks the flag and that shall not be forwarded to the ISM.

## CRC MOSI

### CRC MISO description

CRC MISO safety check guaranty the read SPI communication protocol. TXDIG is generating a CRC reference code which is send to through the SPI communication to M7. M7 decodes the SPI protocol and compute a CRC code using its internal CRC engine.

A safety flag is asserted when the reference value is different from the newly calculated value indicating a wrong transmission of the data through the SP write.

The flag is located in SPI register ERROR\_STATUS/MOSI\_CRC\_ERR

### CRC MISO MASK

Register MASK\_ISM\_FORCE\_ERROR/ MOSI\_CRC\_ERR masks the force\_error generated by the SW towards the ISM.

Register MASK\_ISM\_RESET\_ERROR/MOSI\_CRC\_ERR masks the reset\_error generated by the SW towards the ISM.

Register MASK\_ISM\_ERROR\_FLAG/MOSI\_CRC\_ERR masks the flag and that shall not be forwarded to the ISM.

## TX INL/DNL

### TXINL description

This safety mechanism shall be done through SW programming.

TX phase step measurement is used to perform INL and DNL.

The TX phase step measurement is done using a second TX and the TXBIST .

The principle is described in the TXBIST Architecture specification.

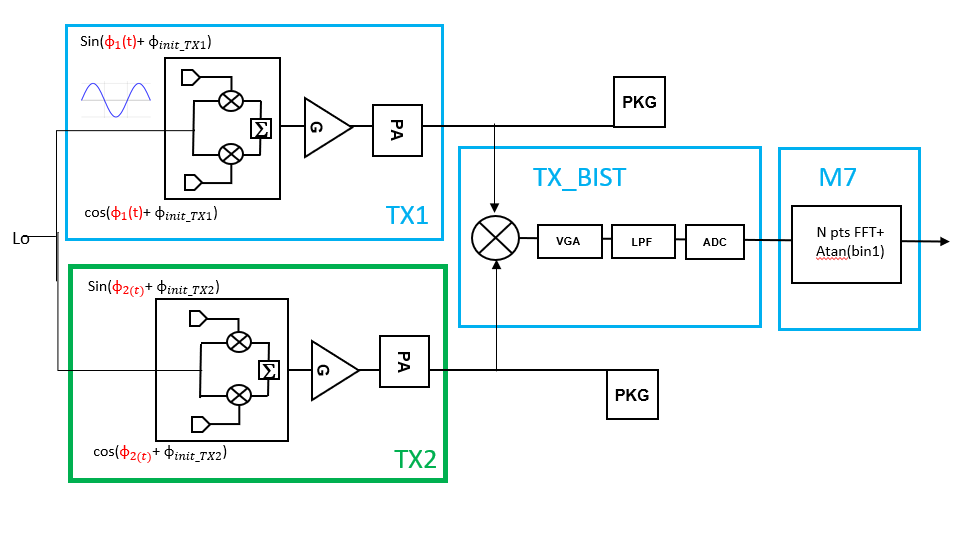


Figure 13: TX INL block diagram

TXs and TXBIST module shall be enabled using SPI registers described in the TX register map document.

8 phase shall be programmed on the TX under test . As example = .

For each of those 8 phases programmed on the TX under test , N phases shall be programmed on the second TX with as constraint to describe a periodic sinewave at the Cordic output. As example, when N=8 = so that cos(

8 FFT and post processing shall give a phase measurement in M7.

Both TX phase programmation shall be done through SPI bit d\_force\_te\_phase when SPI bit d\_force\_te\_phase\_sel=1.

Phase\_in\_degrees= d\_force\_te\_phase\*360/2^7

The post processing is done in M7 which assert a flag in case the phase step is higher than a define M7 threshold.

### TX INL MASK

No mask in the TX digital has the flag is managed through SW.

### TX INL FIT

The safety mechanism is measuring 8 output phases corresponding to 8 TX digital input phase= . Each of the 8 inputs phases are passed through a CORDIC algorithm. The CORDIC algorithm has 3 inputs. The phase (, the gain(G) and a phase init ( ). The output of the CORDIC shall be G\*cos(

A Fault injection shall be introduced by corrupting one of the phase init. By default the phase init is defined by 16 SPI bit d\_phi\_init\_ci and d\_phi\_init\_cq all zeros. Introducing a phase init of +45 degrees on one of the 8 measurement shall inject an error.

The phase init in degrees= d\_phi\_init\_ci/q[15:0]\*360/2\*16.

## TX to TX delta phase variation (TXBIST)

This safety mechanism shall be done through SW programming.

### TX to TX delta variation description

The “TX to TX delta phase variation ” safety mechanism measures the variation of the phase init between 2 TX. If the delta phase variation is higher than a defined threshold, a flag is asserted in M7.

The design under test are the 2 TXs .

The measurement mechanism is composed by the TX BIST module, the ATB ADC and post processing in the ARM7

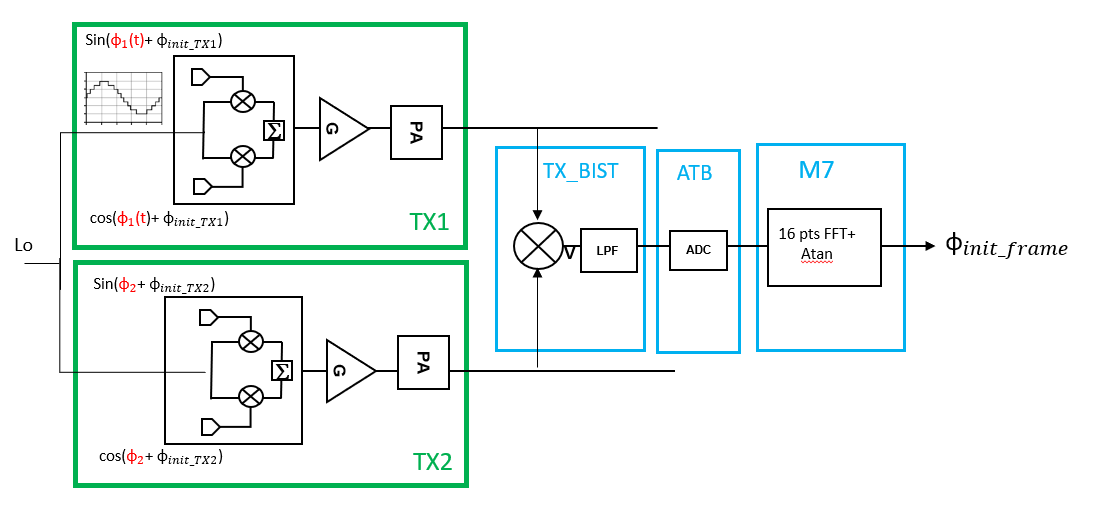


Figure 14: Tx to TX delta phase variation block diagram

Both TX phase programmation shall be done through SPI bit d\_force\_te\_phase when SPI bit d\_force\_te\_phase\_sel=1.

Phase\_in\_degrees= d\_force\_te\_phase\*360/2^7

The first TX is used to generate the IF. In the context of safety, the IF is described using 16 phase = . The FFT shall be done using the same number of points.

The fix phase of the second TX shall be selected in the range [0:360]º

phase init Φinit is computed at the end of each frame in M7

The reference phase is a phase\_init measuring stored in OTP or register.

The delta phase is defined as the difference between the phase init (Φinit) computed at the end of each radar frame and a reference phase.

The delta phase variation is defined as the variation of delta phase over Radar cycle.

### TX to TX delta variation Mask

No mask in the TX digital has the flag is managed through SW.

### TX to TX delta variation FIT

The safety mechanism is measuring the variation of the phase init from TX to TX over Radar frames.

Each of the TX has CORDIC algorithm with 3 inputs parameters. The phase (, the gain(G) and a phase init ( ). The output of the CORDIC shall be G\*cos(

A Fault injection shall be introduced by corrupting one of the TX phase init between 2 radar frames. By default the phase init is defined by 16 SPI bit d\_phi\_init\_ci and d\_phi\_init\_cq all zeros.

The phase init in degrees= d\_phi\_init\_ci/q[15:0]\*360/2\*16.

# ADC communication

## TX ADC /ATB ADC selection

TX/TXBIST can use either TX internal ADC or ATB ADC to do measurement.

SPI bit are controlling the muxing to route the node either on TXADC, ATB1, ATB2

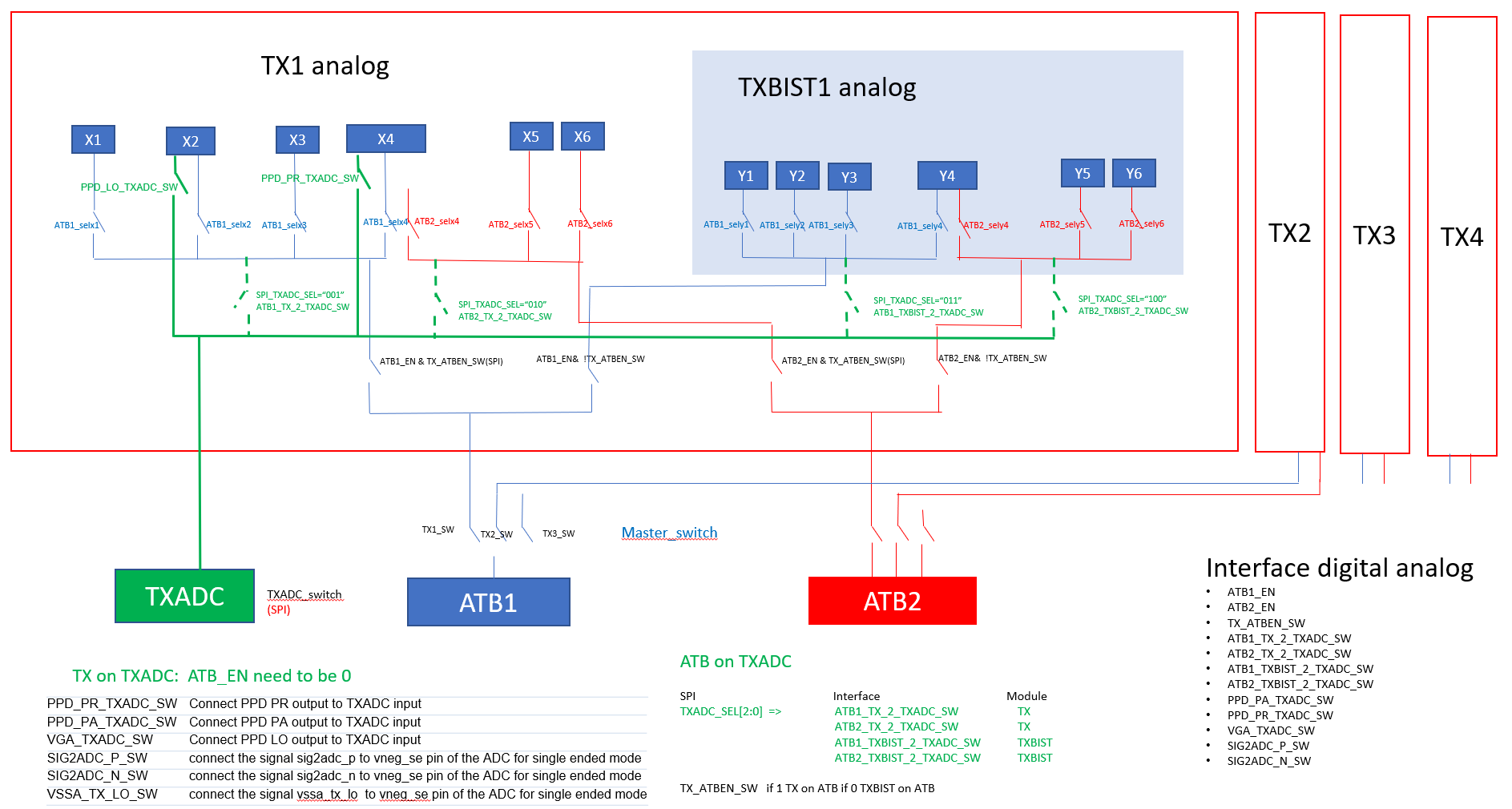


Figure 15: TXADC, TXATB connection

### TX ADC measurement.

The Following SPI bit controls the TXADC input switch.

PPD\_PR\_TXADC\_SW Connect PPD PR output to TXADC input

PPD\_PA\_TXADC\_SW Connect PPD PA output to TXADC input

VGA\_TXADC\_SW Connect PPD LO output to TXADC input

SIG2ADC\_P\_SW connect the signal sig2adc\_p to vneg\_se pin of the ADC for single

SIG2ADC\_N\_SW connect the signal sig2adc\_n to vneg\_se pin of the ADC for

VSSA\_TX\_LO\_SW connect the signal vssa\_tx\_lo  to vneg\_se pin of the ADC for

### ATB ADC measurement

The analog nodes to be observed on ATB ADCs have been classified in 3 groups

* Groups1: nodes can only be observed on ATB1
* Group2: nodes can only be observed on ATB2
* Group2: nodes can be observed on ATB1 and ATB2

**Group1: ATB1 nodes**

The TX nodes shall be observed on ATB1 when ATB1\_EN=1 and TX\_ATBEN\_SW=1

The TX nodes shall be observed on TXADC by setting TXADC\_SEL=”001”

The TXBIST nodes shall be observed on ATB1 when ATB1\_EN=1 and TX\_ATBEN\_SW=0

The TXBIST nodes shall be observed on TXADC by setting TXADC\_SEL=”011”

**Group2: ATB2 nodes**

The TX nodes shall be observed on ATB2 when ATB2\_EN=1 and TX\_ATBEN\_SW=1

The TX nodes shall be observed on TXADC by setting TXADC\_SEL=”010”

The TXBIST nodes shall be observed on ATB2 when ATB2\_EN=1 and TX\_ATBEN\_SW=0

The TXBIST nodes shall be observed on TXADC by setting to 1 TXADC\_SEL= “100”

The TX nodes shall be observed on ATB1 when ATB1\_EN=1 and TX\_ATBEN\_SW=1

The TX nodes shall be observed on ATB2 when ATB2\_EN=1 and TX\_ATBEN\_SW=1

The TX nodes shall be observed on TXADC by setting TXADC\_SEL=”001” or TXADC\_SEL=”010”

**Group3: ATB1 and ATB2 nodes**

The TXBIST nodes shall be observed on ATB1 when ATB1\_EN=1 and TX\_ATBEN\_SW=0

The TXBIST nodes shall be observed on ATB2 when ATB2\_EN=1 and TX\_ATBEN\_SW=0

The TXBIST nodes shall be observed on TXADC by setting TXADC\_SEL=”011” or TXADC\_SEL=”100”

## TXADC communication

The TX ADC shall be enabled through SPI bit TX\_ADC/POWER\_ENABLE

### During Phase rotator calibration

The TXDIG shall generate a conversion pulse to the TXADC during the Phase rotator calibration. The ADC shall answer a 11bits data and a adc\_ready signal

The adc\_conversion pulse shall be fully programmable. Period shall be programmable through SPI bit timer\_adc\_cycle and duty cycle shall be programmable through SPI bit timer\_start\_adc\_conv. Programmation step is 25ns.

Timer\_start\_adc\_conv shall be lower than timer\_adc\_cyle

Timer\_start\_adc\_conv shall be the low state of the pulse.

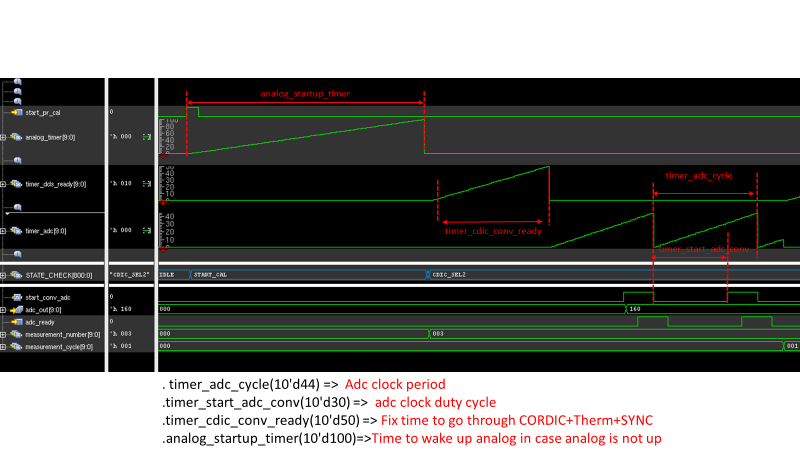


Figure 16: TX ADC perido and duty cycle

### Outside from Phase rotator calibration

TXADC conversion shall be initiated by SPI or OCI when outside the phase rotator calibration. The conversion shall happen when SPI bit ADC\_CONV\_FORCE=1 under the condition than SPI bit ADC\_CONV\_SEL=1. ADC\_CONV\_FORCE SPI bit shall remain high 200ns and shall be auto cleared.

### Timeout flag

A flag shall be asserted when the ADC shall not generate the expected ready\_signal

The flag can be observed through SPI register

When the register read back value is 1, an ADC conversion has given wrong result.

# DFT

## PR DAC linearity

The PRDACs in the TX consist of 256 elements of which each can produce a unity amount of current.

This “PR DAC linearity” feature allows for measuring this current per DAC element. Actually two currents must be measured per element, the P-channel current and N-channel current. The currents can be measured by driving the PR DAC with a special control pattern. To that end, the TX Dig can generate 2x256 patterns under the control of the registers:

Table 4: DAC linearity register

|  |  |
| --- | --- |
| Register name | Function |
| PRC\_DFT\_ DAC\_TESTMODE | 0: Normal operation, PR DAC linearity disabled  1: PR DAC linearity enabled |
| PRC\_DFT\_ DAC\_TESTMODE\_INVERTED | 0: allow measuring P-channel currents  1: allow measuring N-channel currents |
| PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I | 9 bit signed number, allows for measuring the current of the DAC\_I elements i=0…255=n+127 by programming the value n = -127…128. |
| PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_Q | 9 bit signed number, allows for measuring the current of the DAC\_Q elements i=0…255=n+127 by programming the value n = -127…128. |

See also <https://www.collabnet.nxp.com/sf/go/artf782421>.

### P-channel current measurement-test

PRC\_DFT\_ DAC\_TESTMODE = 1

PRC\_DFT\_ DAC\_TESTMODE\_INVERTED = 0

PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q = -127 … 128

For the P-channel current measurement test, the control of the PR DAC shall be a 1-hot encoded value like: “0001000…000”. So, 255 zeros and only 1 bit asserted. The asserted bit is defined by the programmed value in PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q. The programmed value n, asserts the control bit of DAC element i = n + 127.

Verification should check that the P-channel current of all DAC elements can be measured as described.

### N-channel current measurement-test

PRC\_DFT\_ DAC\_TESTMODE = 1

PRC\_DFT\_ DAC\_TESTMODE\_INVERTED = 1

PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q = -127 … 128

For the N-channel current measurement test, the control of the PR DAC shall be an inverted 1-hot encoded value like: “1110111…111”. So, 255 one and only 1 bit de-asserted. The asserted bit is defined by the programmed value in PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q. The programmed value n, de-asserts the control bit of DAC element i = n + 127.

Verification should check that the N-channel current of all DAC elements can be measured as described.

### Disabled DFT linearity test

PRC\_DFT\_ DAC\_TESTMODE = 0

PRC\_DFT\_ DAC\_TESTMODE\_INVERTED = Don’t Care

PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q = Don’t Care

Verification should check that the registers PRC\_DFT\_ DAC\_TESTMODE\_INVERTED and PRC\_THERM\_SW\_D\_FORCE\_THERMIN\_I/Q are don’t care in case the DFT linearity test is disabled.

## PR settling time

The waveforms in Figure 5: Phase rotator settling time, illustrate the principle of the measurement. The corresponding register are given in Table 4: PR settling time registers.

Table 5: PR settling time registers

|  |  |  |
| --- | --- | --- |
| Register | R/W | Description |
| PR\_STLING\_TIME\_EN | RW | 0: Disabled  1: Enabled |
| PR\_STLING\_STAT\_TIME\_RD | R | 7 bits settling-time read-out value, 25ns resolution |
| TXBIST\_CTRL\_STLING\_TIME\_EN\_N | RW | enable TXBIST out to TXBIST settling\_time comparator n input |
| TXBIST\_CTRL\_STLING\_TIME\_EN\_P | RW | enable TXBIST out to TXBIST settling\_time comparator p input |
| TX\_PA\_EN\_RTM | RW | enable RTM |

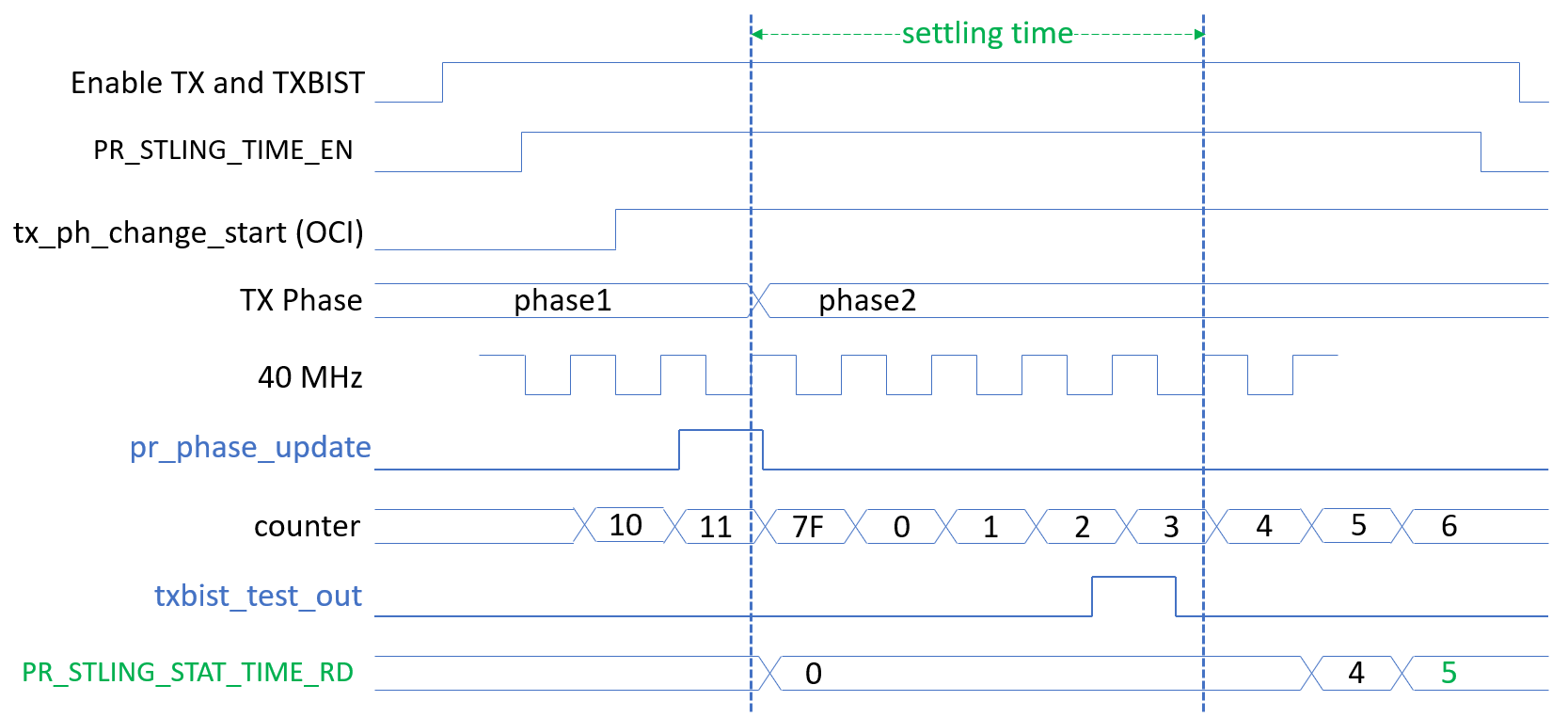
The measurement reports the time from a phase-rotator change, as defined by an up transition of pr\_phase\_update, to the last transition of comparator output txbist\_test\_out.

Figure 17: Phase rotator settling time

IT shall be possible to select the N or P path of the comparator output

When TXBIST\_CTRL\_STLING\_TIME\_EN\_N=1 txbist\_test\_out shall be the N path output of the comparator

When TXBIST\_CTRL\_STLING\_TIME\_EN\_P=1 txbist\_test\_out shall be the P path output of the comparator

To perform a measurement, the TX, TXBIST and the tx\_pr\_settling\_time module must be enabled. The latter is done by setting register PR\_STLING\_TIME\_EN.

The settling time is measured by a counter that is clocked at 40MHz. Hence the resolution of the measurement time is 25ns. Once finished, the measurement time is stored in register PR\_STLING\_STAT\_TIME\_RD. The value in this register is copy of the counter at every detection of a transition on txbist\_test\_out.

At the start of the measurement, namely at rising transition of pr\_phase\_update, both the counter and register PR\_STLING\_STAT\_TIME\_RD are reset to 0xFF and 0x00 respectively.

When the module is disabled (PR\_STLING\_TIME\_EN = 0), both the counter and PR\_STLING\_STAT\_TIME\_RD are kept constant.

The counter has a width of 7 bits, so a settling time up to 128\*25ns = 3.2us can be measured.

Note that register TX\_PA\_EN\_RTM shall be programmed to ‘0’, otherwise txbist\_test\_out gets disconnected.

The circuit for detecting transitions on txbist\_test\_out is given in Figure 6: PR settling time transition detection

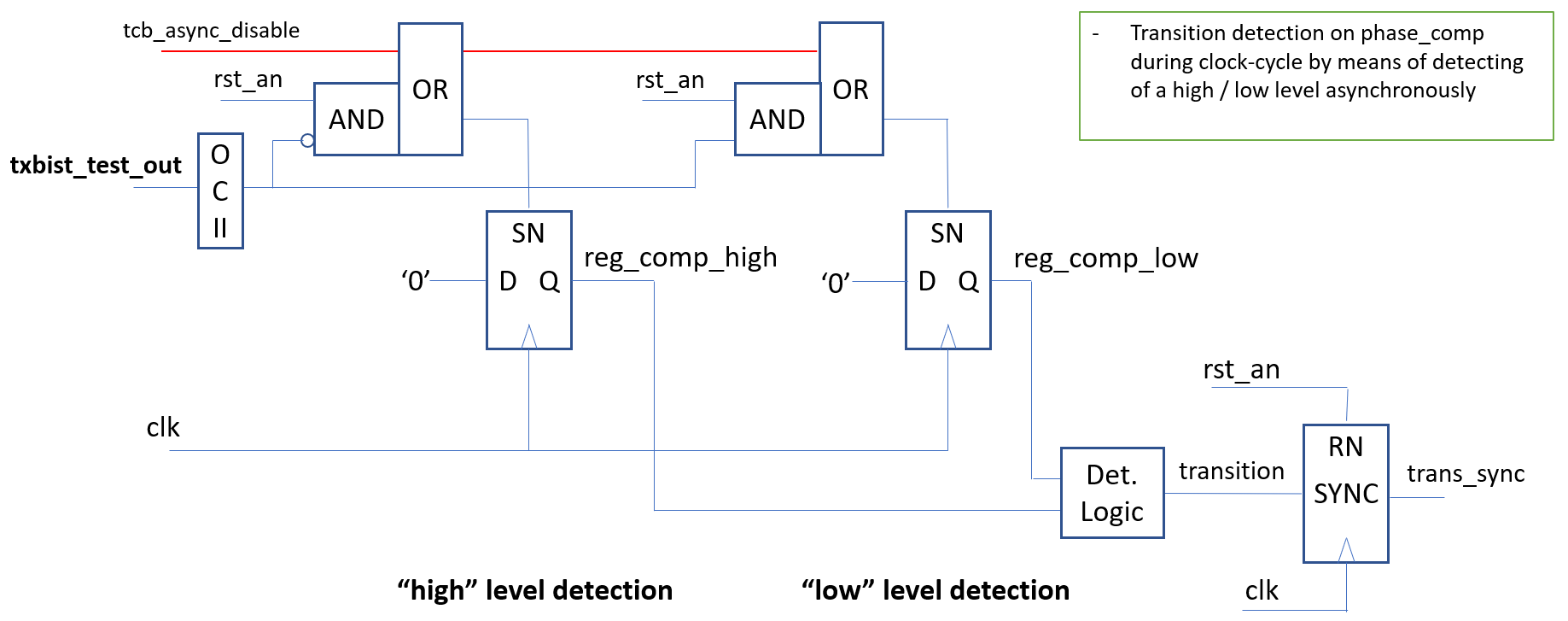


Figure 18: PR settling time: Transition detection mechanism

This circuit can detect transitions and pulses that are asynchronous to the 40MHz clock. To that end, two flip-flops are present which respectively detect the presence of a “high” level and/or a “low” level during the present clock cycle. In case such conditions are being detected, the signals det\_high and det\_low will get asserted correspondingly.

Based on these signals, a (rising and falling) transition is being detected in the block called “Det. Logic”. This block can detect a transition in three situations as depicted in Figure 7:PR settling time: Transition detection casesIn this situation both a high and low level have been seen during the current clock cycle

1. In this situation, a low level has been seen during the current clock cycle and a high level during the past clock cycle
2. In this situation, a high level has been seen during the current clock cycle and a low level during the past clock cycle
3. In this situation, a high level has been seen during the current clock cycle and a low level during the past clock cycle

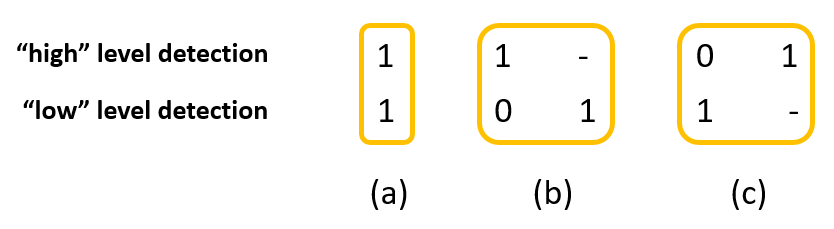


Figure 19:PR settling time: Transition detection cases

In this way, also a transition of txbist\_test\_out exactly at the rising clock edge is being detected.Figure 8:PR settling time: Transition detection example, illustrates two of the three possible detection cases (namely a, and c).

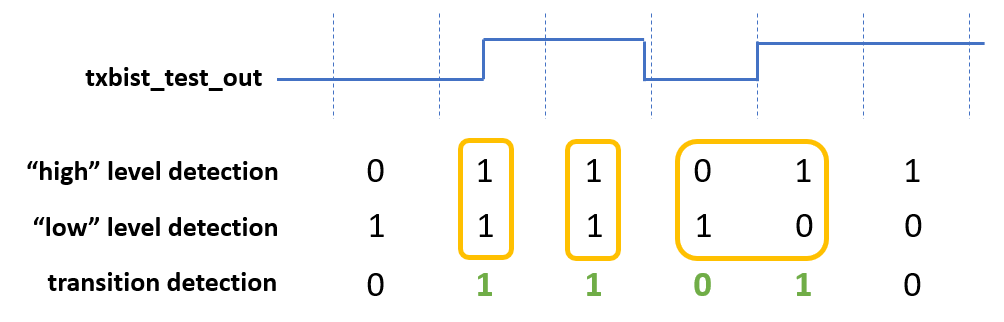


Figure 20:PR settling time: Transition detection example

The transition detection module is fully combinatorial. To use the result by the clocked state machine of the module, the output signal must be synchronized to the 40 MHz clock which is done by a synchronizer, resulting in the signal trans\_sync.

For the purpose of DFT, the asynchronous reset signals of the two level detection flip-flops can be blocked by signal tcb\_async\_disable.

The comparator output signal txbist\_test\_out is passed through a OCII slice in cluster 43. By doing so:

1. Isolation is realized during Scan-test
2. Allows testing of the paths from txbist\_test\_out through the logic towards the receiving registers
3. No need for dedicated DFT observation logic at the interface of the tx\_pr\_settling\_time module for testing the path from analog IP towards the digital IP

## IF generation (UPN)

A phase ramp shall be generated at the cordic input when SPI bit tx\_if\_en =1

The step of the ramp is defined by tx\_if\_phaseinc. Tx\_if\_phaseinc is a 12 bits SPI register

" The phase ramp shall be converted to sinewave signal at CORDIC output

Cordic I output (cdic\_out\_i) shall frequency of f0=40e6\*tx\_if\_phaseinc\*16/2^16 "

"The phase ramp shall be converted to sinewave signal at CORDIC output

Cordic Q output (cdic\_out\_q) shall frequency of f0=40e6\*tx\_if\_phaseinc\*16/2^16 "

CORDIC\_I and CORDIC\_Q output shall be in quadrature

CORDIC\_I and CORDIC\_Q output signal shall have both a SNDR>30dB in a [0-20MHz] bandwidth with f0 in [100K,10MHz]

## FSM STOP

"The FSM shall be stopped at the end of a state by setting to '1' it’s corresponding stop bit.

(in this case, end of the state means that the operations are over, but the end signal has not been asserted)

The stop shall be asserted prior to the start of the calibration .

This stop shall be used to read back the latest ADC measurement of the completed state.

Once the analysis/debug are done, the stop signal shall be de-asserted, which will unlock the end signal of the state and will move the FSM to next state"

The state machine shall restart by writing the SPI stop bit back to 0.

It has to be notified that the first ADC sample output after the restart may be corrupted.

STOP\_START\_CAL stops in the FSM in “start\_cal” state.

STOP\_ADC\_AVE stops in the FSM in “adc\_ave” state.

STOP\_VGA\_INIT stops in the FSM in “vga\_init” state

STOP\_VGA\_OFFSET\_CAL stops in the FSM in “vga\_offset\_cal” state.

STOP\_VGA\_GAIN\_CAL stops in the FSM in “vga\_gain\_cal” state.

STOP\_PR\_OFFSET\_CAL stops in the FSM in “pr\_offset\_cal” state.

STOP\_GPATH\_SEL stops in the FSM in “gpath\_sel” state.

STOP\_PR\_GAIN\_CAL stops in the FSM in “pr\_gain\_cal” state.

STOP\_PR\_PHASE\_CAL stops in the FSM in “pr\_phase\_cal” state.

STOP\_END\_CAL stops in the FSM in “end\_cal” state.

## ADC measurement in FSM state

When stop signal is asserted in a state of the FSM, SPI bits ppd\_meas1, ppd\_meas2, ppd\_meas3, ppd\_meas4, ppd\_meas5,ppd\_meas6, ppdmeas7, ppd\_meas8, ppd\_meas9, ppd\_meas10, ppd\_meas11, ppd\_meas12, ppd\_meas13, ppd\_meas14, ppd\_meas15, ppd\_meas16 shall store the 16 first ADC measurement of the current state."

## GPIO

the TX digital controller shall provide signals through TX debug[7:0] interface signal going to GPIO interface

"the following signals are connected to the GPIO

tx\_debug[0]=tx\_fast\_sw;

tx\_debug[1]=tx\_fast\_en ;

tx\_debug[2]=tx\_en; is the output of a OR gate on which inputs are all the enable of the TX

tx\_debug[3]=pr\_phase\_update; indicating that the phase of the PR has been updated

tx\_debug[4]=pr\_cal\_run ; indicating the calibration is running

tx\_debug[5]=txbist\_test\_out ; txbist\_test\_out is the output of the TXBIST comparator

tx\_debug[6]= tx\_comp\_out; tx\_comp\_out is the output of the comparator used for real time PA output power measurement

tx\_debug[7]= instr\_error\_status\_bbd\_err\_flag\_n . instr\_error\_status\_bbd\_err\_flag\_n is the signal from RF ball break detector module"

PR settling time shall be the time difference between the pr\_phase\_update and txbist\_test\_out