SmartLEWIS ™ RX+
TDA5240/35/25 Explorer

Configuration and Evaluation Software B12.6.33-51

High Sensitivity Receiver with
Digital Baseband Processing (TDA5240/35) /
Digital Slicer (5225)

Register Value Calculations

Addendum to Data Sheet
Released, 2011-03-11

Wireless Control

Rev0.1 2011-03-11

Published by
Infineon Technologies AG
81726 Munich, Germany

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.

Information

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

Warnings

Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.

Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

Explorer Calculation
Revision History: 2011-03-11, Released

Previous Revision:

Page	**Subjects (major changes since last revision)**
	Update of formulas

Table of Contents

Introduction

This document describes all calculations for registers implemented by the TDA5240/35/23 Explorer software. Calculations that recover input values from register values are not covered by this document. Some calculations are valid for multiple configuration pages (Config A,B,C,D) of the chip. Affected register names contain the placeholder %Page%.

The crystal oscillator frequency is changeable in the Explorer software. However, for this document a constant input of 21.948717 MHz is used for the system frequency.

Arguments of a function are separated by a comma. Digits of a floating point number are separated by a dot.

Definitions for rounding functions are:

- Math rounding: $Round(value, [decimals])$ Rounds to the next higher integer away from zero number if the number was at the half way point
  Example: $Round(-3.5) = -4$; $Round(3.5) = 4$
- Bankers rounding: $Round_{toEven}(value, [decimals])$ Rounds to the even number if the number was at the half way point
  Example: $Round_{toEven}(2.5) = 2$; $Round_{toEven}(3.5) = 4$
- $Floor(x) = \lfloor x \rfloor$ is the largest integer not greater than x
- $Ceiling(x) = \lceil x \rceil$ is the smallest integer not less than x
- $Abs(x) = |x|$ is the numerical value of x without regard to its sign

Definitions for logical functions are:

- $Or(x, y) = x \lor y$ is true if at least one of the conditions evaluates to true
- $And(x, y) = x \land y$ is true if both conditions evaluates to true

Definitions for notations in the set theory are:

- $\mathrm{Union(X, Y)} = X \cup Y$ is the collection of points which are in X or in Y (or in both)
- $\emptyset = \{\}$:= The empty set is the set having no elements

1 Wizard Page 2 – RF PLL

Input variables and constants:

- System frequency; $\mathrm{fsys} = 21948717\ [\mathrm{Hz}]$
- RF channel frequency for channel ch: rf(ch) [MHz]; ch ∊ Channels; Channels = {1, 2, 3}
- Single/Double conversion selection: ifconv = (SFR_%Page%_IF1.SDCSEL)
- RXRF Receive Side Band Select: sbsel = (SFR_%Page%_IF1.SSBSEL)

Intermediate variables:

- Single/Double conversion frequency: conv [MHz]

Output variables and registers allocation:

- ISM band: ismb(ch) = (SFR_%Page%_PLLINTC1.BANDSEL)
- PLL Fractional Division Ratio: fracdiv(ch) = (SFR_%Page%_PLLFRAC2C(ch),SFR_%Page%_PLLFRAC1C(ch),SFR_%Page%_PLLFRAC0C(ch))
- Fractional Spurii Compensation: fraccomp(ch) = (SFR_%Page%_PLLFRAC2C(ch).PLLFCOMPC(ch)) := 0
- PLL Multi Modulus Divider Integer Offset: mmdiv(ch) = (SFR_%Page%_PLLINTC1.PLLINTC(ch))

$$ismb = \begin{cases} 3, & if\ rf(x) > 299.999\ \wedge rf(x) < 320.0001\ \forall\ x\ \in Channels \\ 2, & if\ rf(x) > 424.999\ \wedge rf(x) < 450.0001\ \forall\ x\ \in Channels \\ 1, & if\ rf(x) > 862.999\ \wedge rf(x) < 928.0001\ \forall\ x\ \in Channels \\ 0, & else \end{cases}$$

$$conv = \begin{cases} 10.7, & if\ ifconv = 0 \\ \frac{10.7}{39}, & else \end{cases}$$

$$fracdiv(ch) = \left\lfloor (2^{21} - 0.5) * \left(\frac{rf(ch) - \big(((sbsel * 2) - 1) * -conv\big)}{\frac{fsys}{1000000}} - \left\lfloor \frac{rf(ch) - \big(((sbsel * 2) - 1) * -conv\big)}{\frac{fsys}{1000000}} \right\rfloor \right) - 0.5 \right\rfloor$$

Explorer version > B12.6.37 uses $fracdiv2$ instead of $fracdiv$

$$fracdiv2\ (ch) = \max\left(\left\lfloor (2^{21} - 0.5) * \left(\frac{rf(ch) - \big(((sbsel * 2) - 1) * -conv\big)}{\frac{fsys}{1000000}} - \left\lfloor \frac{rf(ch) - \big(((sbsel * 2) - 1) * -conv\big)}{\frac{fsys}{1000000}} \right\rfloor \right) - 0.5 \right\rfloor, 0 \right)$$

$$mmdiv(ch) = \left\lfloor \frac{rf(ch) - \big(((sbsel * 2) - 1) * -conv\big)}{fsys/1000000} \right\rfloor$$

1.1 Spur warning

Input variables and constants:

- EMI Source: fEmi [MHz] (fEmi can be an external EMI source or the crystal itself)
- Maximal spur frequency: fMax [MHz] := 3000
- Wanted bandwith: fWantedBW [MHz] := 0.3

Intermediate variables and sets:

- Local oscillator harmonics for low side injection: $fLOharmonic_{low}$ [MHz]
- Local oscillator harmonics for high side injection: $fLOharmonic_{high}$ [MHz]
- Mixer sideband: nSideband
- The set of spurs when using low side injection: $FDelta_{Low}$
- The set of spurs when using high side injection: $FDelta_{High}$

Output variables:

- The value the crystal oscillator frequency must be shifted to avoid spurs in the actual channel: fDeltaMax [MHz]
- Spur warning: spurWarning ∈ SpurWarnings; SpurWarnings = {“No spur detected”, “Crystal frequency shift”, “Low side injection recommended”, “High side injection recommended”}

$$nSideband = \begin{cases} -1, & if\ nLo - 2\left\lfloor \frac{nLo}{2} \right\rfloor \neq 0 \\ 1, & else \end{cases}$$

$$fDelta_{Low} = \frac{fWantedBW}{2} - \big||nEmi * fEmi - (2 * nLo + 1) * (rf(ch) - conv)| - conv\big|$$

$$fDelta_{High} = \frac{fWantedBW}{2} - \big||nEmi * fEmi - (2 * nLo + 1) * (rf(ch) + conv)| - conv\big|$$

$$FDelta_{Low} = \left\{ fDelta_{Low}: |nSideband * (nEmi * fEmi - (2 * nLo + 1) * rf(ch) - conv)| < \frac{fWantedBW}{2}\ \forall nEmi \in \mathbb{N}: 1 \le nEmi < \frac{fMax}{fEmi}\ \forall nLo \in \mathbb{N}: 1 \le \frac{fMax - rf(ch) - conv}{2 * (rf(ch) + conv)} + 1 \right\}$$

$$FDelta_{High} = \left\{ fDelta_{High}: |nSideband * (nEmi * fEmi - (2 * nLo + 1) * rf(ch) + conv)| < \frac{fWantedBW}{2}\ \forall nEmi \in \mathbb{N}: 1 \le nEmi < \frac{fMax}{fEmi}\ \forall nLo \in \mathbb{N}: 1 \le \frac{fMax - rf(ch) - conv}{2 * (rf(ch) + conv)} + 1 \right\}$$

$$fDelta_{Max} = f_{Max} \in FDelta_{Low} \cup FDelta_{High}: \forall\, f \in FDelta_{Low} \cup FDelta_{High}: f < f_{Max}$$

$$\text{spurWarning} = \begin{cases} \text{“Crystal frequency shift”}, & \text{if } \text{FDelta}_{\text{High}} \neq \emptyset \wedge \text{FDelta}_{\text{Low}} \neq \emptyset \\ \text{“No spur detected”}, & \text{if } \text{FDelta}_{\text{High}} = \emptyset \wedge \text{FDelta}_{\text{Low}} = \emptyset \\ \text{“Low side injection recommended”}, & \text{if } \text{FDelta}_{\text{High}} \neq \emptyset \wedge \text{FDelta}_{\text{Low}} = \emptyset \\ \text{“High side injection recommended”}, & \text{if } \text{FDelta}_{\text{High}} = \emptyset \wedge \text{FDelta}_{\text{Low}} \neq \emptyset \end{cases}$$

Note: If the variable “spurWarning” evaluates to “Crystal frequency shift”, then a crystal oscillator frequency shift by $fDelta_{Max}$ MHz is needed to avoid a spur.

2 Wizard Page 5 – Digital Receiving Unit

Input variables and constants:

- System frequency; $\text{fsys} = 21948717\ [\text{Hz}]$
- Datarate: datarate [bit/s]
- Chip per bit selection: cpb := 1 if one chip per bit is selected, 2 if two chip per bit is selected
- Attack (Up) Time: attTime [bit]
- Decay (Down) Time: decTime [bit]
- Expected max. FSK deviation: fdev [+/-kHz]
- Expected min. FSK deviation: $fdev_{min}$ [+/-kHz]
- Modulation type: modtype = (SFR_%Page%_CHCFG.MT) (taken from Wizard page 1 – Master Control Unit)
  modtype=0 → Pure ASK
  modtype=1 → Pure FSK
  modtype>1 → Mixed Mode
- Band Pass Filter Bandwidth (analog BW): bpf = (SFR_%Page%_IF1.BPFBWSEL)

- Predemodulation Bandwidth (digital BW): dbpf = (SFR_%Page%_ISUPFCSEL.FCSEL)
- Settling Time: stime [bit]

Intermediate variables:

- Band Pass Filter Bandwidth (analog BW): bpf_{BW} [kHz]
- Predemodulation Bandwidth (digital BW): $dbpf_{BW}$ [kHz]
- Maximum frequency offset: foffsetmax [kHz]
- Minimum frequency offset: foffsetmin [kHz]
- Maximum amplitude of FSK offset after predecimation before scaling: predec_max_bs_offset
- Maximum amplitude of FSK deviation after predecimation before scaling: predec_max_bs_fdev
- Maximum amplitude of ASK output after predecimation before scaling: predec_max_bs
- Minimal scaling for given coefficient: $scal_{min}$
- Selector for minimal scaling: sel_{min}

Output variables and registers allocation:

- Attack factor: attFactor = (SFR_%Page%_PMFUDSF.PMFUP)
- Decay factor: decFactor = (SFR_%Page%_PMFUDSF.PMFDN)
- Downsampling of signal after arctan demodulation: $\lfloor \mathrm{decimation} - 1 \rfloor$= (SFR_%Page%_PDECF.PREDECF)
- Matched filter: mf = (SFR_%Page%_MFC.MFL)
- Sample rate converter: src = (SFR_%Page%_SRC) | min=0 < max=255
- Predecimation Block Scaling Factor for ASK: $pdscale_{ASK}$ = (SFR_%Page%_PDECSCASK.PDSCALEA)
- Predecimation Block Scaling Factor for FSK: $pdscale_{FSK}$ = (SFR_%Page%_PDECSCFSK.PDSCALEF)
- FSK data interpolation enable: $interpol_{FSK}$ = (SFR_%Page%_PDECSCFSK.INTPOLENF)
- ASK data interpolation enable: $interpol_{ASK}$ = (SFR_%Page%_PDECSCASK.INTPOLENA)
- Saturation threshold of the Sigdet peak detector used for zero-tube threshold calculation: pd = (SFR_%Page%_SIGDETSAT) | 0 < pd < 254 (255=disabled)
- Raw data slicer BW selection: bwsel = (SFR_%Page%_EXTSLC.ESLCBW)
- Raw data slicer BW selection scaling: bwscal = (SFR_%Page%_EXTSLC.ESLCSCA)

$$bpf_{BW} = \begin{cases} 50, & if\ bpf = 0 \\ 80, & if\ bpf = 1 \\ 125, & if\ bpf = 2 \\ 200, & if\ bpf = 3 \\ 300, & else \end{cases}$$

Explorer version <= B12.6.33 used $dbpf_{BW\_old}$ instead of $dbpf_{BW}$

$$dbpf_{BW} = \begin{cases} 33, & if\ dbpf = 0 \\ 46, & if\ dbpf = 1 \\ 65, & if\ dbpf = 2 \\ 93, & if\ dbpf = 3 \\ 132, & if\ dbpf = 4 \\ 190, & if\ dbpf = 5 \\ 239, & if\ dbpf = 6 \\ 282, & else \end{cases} \qquad dbpf_{BW\_Old} = \begin{cases} 33, & if\ dbpf = 0 \\ 45, & if\ dbpf = 1 \\ 64, & if\ dbpf = 2 \\ 92, & if\ dbpf = 3 \\ 132, & if\ dbpf = 4 \\ 180, & if\ dbpf = 5 \\ 238, & if\ dbpf = 6 \\ 282, & else \end{cases}$$

$$foffsetmax = \max\left(0, \begin{cases} \dfrac{bpf - \dfrac{datarate * cpb}{1000}}{2}, & if\ modtype = 0 \\ \dfrac{\min(bpf, dbpf) - \dfrac{datarate * cpb}{1000} - 2 * fdev}{2}, & else \end{cases}\right)$$

$$foffsetmin = \max\left(0, \begin{cases} (bpf - \dfrac{datarate * cpb}{1000})/2, & if\ modtype = 0 \\ (\min(bpf, dbpf) - \dfrac{datarate * cpb}{1000} - 2 * fdev_{min})/2, & else \end{cases}\right)$$

$$coeff = \begin{cases} \dfrac{\dfrac{stime}{abs\left(\ln\frac{1}{8}\right)} * \dfrac{1}{datarate} + \dfrac{1}{datarate * cpb * \max(mf + 1, 1)}}{\dfrac{1}{datarate * cpb * \max(mf + 1, 1)}}, & if\ modtype = 0 \lor foffsetmin < fdev_{min} \\ \dfrac{\dfrac{stime}{abs\left(\ln\frac{fdev_{min}}{8 * foffsetmin}\right)} * \dfrac{1}{datarate} + \dfrac{1}{datarate * cpb * \max(mf + 1, 1)}}{\dfrac{1}{datarate * cpb * \max(mf + 1, 1)}}, & else \end{cases}$$

$$scal_{min} = \min\left(\max\left(\left\lceil \log_2 \frac{coeff}{48} \right\rceil - 1, 0\right), 3\right)$$

$$sel_{min} = \min\left(\left\lfloor \frac{coeff}{8 * 2^{\min\left(\max\left(\left\lceil \log_2 \frac{coeff}{48} \right\rceil - 1, 0\right), 3\right) + 1}} - 1 \right\rfloor, 5\right)$$

$$coeff_{high} = \begin{cases} 768, & if\ sel_{min} = 5 \land scal_{min} = 3 \\ \min(8 * 2^{scal_{min}+2}, 768), & if\ sel_{min} = 5 \land scal_{min} < 3 \\ \min\left((sel_{min} + 2) * 8 * 2^{scal_{min}+1}, 768\right), & else \end{cases}$$

$$coeff_{low} = \min\left((sel_{min} + 1) * 8 * 2^{scal_{min}+1}, 768\right)$$

$$bwsel = \begin{cases} \begin{cases} 5, & if\ sel_{min} = 5 \land scal_{min} = 3 \\ 0, & if\ sel_{min} = 5 \land scal_{min} < 3 \\ sel_{min} + 1, & else \end{cases}, & if\ coeff - coeff_{low} > coeff_{high} - coeff \\ sel_{min}, & else \end{cases}$$

$$bwscal = scal_{min}$$

$$attFactor = Round\left(\log_2\left(\frac{attTime}{datarate} * \frac{fsys}{80}\right)\right) - 1$$

$$decFactor = Round\left(\log_2\left(\frac{decTime}{datarate} * \frac{fsys}{80}\right)\right) - attFactor - 2$$

$$pd = \begin{cases} Round(predec\_max\_bs * (2^{pdscale_{ASK}-10}) * (mf+1) * 0.5 * 0.95/128), & if\ modtype = 0 \\ Round(predec\_max\_bs\_fdev * (2^{pdscale_{FSK}-10}) * (mf+1) * 0.5 * 0.95/128), & if\ modtype = 1 \\ \min\left(Round\left(predec\_max\_bs \ * (2^{pdscale_{ASK}-10}) * (mf+1) * 0.5 * 0.95/128\right),\right. & \\ \left. Round(predec\_max\_bs\_fdev * (2^{pdscale_{FSK}-10}) * (mf+1) * 0.5 * 0.95/128)\ \right), & else \end{cases}$$

2.1 Datarate Calculation for TDA5240/35

Note: Use the calculation declared in section 2.2 if (SFR_%Page%_CHCFG.EXTPROC) = 2 (taken from Wizard page 1 – Master Control Unit)

Input variables and constants:

- Samples per chip: samplesPerChip := 8

Pure FSK (modtype=1):

$$interpol_{FSK} = \begin{cases} 1, & if\ datarate * cpb * samplesPerChip > fsys/40 \\ 0, & else \end{cases}$$

$$decimation = \begin{cases} 0.5, & if\ datarate * cpb * samplesPerChip > fsys/40 \\ \min\left(\left\lfloor \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right), & else \end{cases}$$

$$\text{predec\_max\_bs\_offset} = \begin{cases} 8 * \text{foffsetmax} * \frac{64}{\text{fsys}/40000}, & if\ datarate * cpb * samplesPerChip > fsys/40 \\ \text{foffsetmax} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2, & \text{else} \end{cases}$$

$$\text{predec\_max\_bs\_fdev} = \begin{cases} 8 * \text{fdev} * \frac{64}{\text{fsys}/40000}, & if\ datarate * cpb * samplesPerChip > fsys/40 \\ \text{fdev} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2, & \text{else} \end{cases}$$

$$mf = Round\left(\frac{fsys/40}{\text{decimation} * cpb * datarate}\right) - 1$$

$$src = Round\left(\left(\frac{fsys/40}{decimation * datarate * cpb * samplesPerChip} - 1\right) * 256\right)$$

$$pdscale_{FSK} = \left\lfloor \log_2 \frac{4095}{\text{predec\_max\_bs\_offset} + 1.4 * \text{predec\_max\_bs\_fdev}} \right\rfloor + 10$$

Pure ASK (modtype=0):

$$interpol_{ASK} = 1$$

$$interpol_{FSK} = 1$$

$$decimation = \min\left(\left\lfloor\frac{fsys/40}{datarate * cpb * samplesPerChip}\right\rfloor, 69\right)$$

$$mf = Round\left(\frac{fsys/40}{\text{decimation} * cpb * datarate}\right) - 1$$

$$src = Round\left(\left(\frac{fsys/40}{decimation * datarate * cpb * samplesPerChip} - 1\right) * 256\right)$$

$$predec\_\max\ \_bs = \begin{cases} 31 * decimation^2, & if\ decimation > 1 \\ 8 * 31, & else \end{cases}$$

$$pdscale_{ASK} = \left\lfloor \log_2 \frac{4095}{predec\_\max\ \_bs} \right\rfloor + 10$$

Mixed Mode (modtype>1):

$$interpol_{ASK} = 1$$

$$interpol_{FSK} = 0$$

$$decimation = \min\left(\left\lfloor\frac{fsys/40}{datarate * cpb * samplesPerChip}\right\rfloor, 69\right)$$

$$mf = Round\left(\frac{fsys/40}{\text{decimation} * cpb * datarate}\right) - 1$$

$$predec\_\max\ \_bs = \begin{cases} 31 * decimation^2, & if\ decimation > 1 \\ 8 * 31, & else \end{cases}$$

$$pdscale_{ASK} = \left\lfloor \log_2 \frac{4095}{predec\_\max\ \_bs} \right\rfloor + 10$$

$$\text{predec\_max\_bs\_offset} = \text{foffsetmax} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

$$\text{predec\_max\_bs\_fdev} = \text{fdev} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

$$pdscale_{FSK} = \left\lfloor \log_2 \frac{4095}{\text{predec\_max\_bs\_offset} + 1.4 * \text{predec\_max\_bs\_fdev}} \right\rfloor + 10$$

$$src = Round\left(\left(\frac{fsys/40}{decimation * datarate * cpb * samplesPerChip} - 1\right) * 256\right)$$

2.2 Datarate Calculation for TDA5225

Note: Use the calculation declared here if (SFR_%Page%_CHCFG.EXTPROC) = 2 (taken from Wizard page 1 – Master Control Unit)

Input variables and constants:

- Samples per chip: samplesPerChip = 16

Pure FSK (modtype=1) and datarate*cpb < 1000:

$$interpol_{FSK} = 0$$

$$decimation = \begin{cases} \min\left(\left\lceil \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rceil, 69\right), if \ \frac{fsys/40}{\min\left(\left\lfloor \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right) * datarate * cpb} > 16.5 \\ \min\left(\left\lfloor \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right), \quad else \end{cases}$$

$$\text{predec\_max\_bs\_offset} = \text{foffsetmax} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

$$\text{predec\_max\_bs\_fdev} = \text{fdev} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

Pure FSK (modtype=1) and datarate*cpb >= 1000:

$$interpol_{FSK} = 1$$

$$decimation = \begin{cases} \min\left(\left\lceil \frac{fsys/20}{datarate * cpb * samplesPerChip} \right\rceil, 69\right), if \ \frac{fsys/20}{\min\left(\left\lfloor \frac{fsys/20}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right) * datarate * cpb} > 16.5 \\ \min\left(\left\lfloor \frac{fsys/20}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right), \quad else \end{cases}$$

$$predec\_max\_bs\_offset = \begin{cases} \text{foffsetmax} * \dfrac{64}{\text{fsys}/40000} * \text{decimation}^2, & \text{if decimation} > 1 \\ 8 * \text{foffsetmax} * \dfrac{64}{\text{fsys}/40000}, & \text{else} \end{cases}$$

$$predec\_max\_bs\_fdev = \begin{cases} \text{fdev} * \dfrac{64}{\text{fsys}/40000} * \text{decimation}^2, & \text{if decimation} > 1 \\ 8 * \text{fdev} * \dfrac{64}{\text{fsys}/40000}, & \text{else} \end{cases}$$

Pure FSK (modtype=1, use intermediate variable from datarate-depended formulas above):

$$pdscale_{FSK} = \left\lfloor \log_2 \frac{4095}{\text{predec\_max\_bs\_offset} + 1.4 * \text{predec\_max\_bs\_fdev}} \right\rfloor + 10$$

$$mf = Round\left(\frac{fsys/20}{\text{decimation} * cpb * datarate}\right) - 1$$

$$src = 0$$

Pure ASK (modtype=0):

$$decimation = \begin{cases} \min\left(\left\lceil \dfrac{fsys/40}{datarate * cpb * samplesPerChip} \right\rceil, 69\right), if \dfrac{fsys/40}{\min\left(\left\lceil \dfrac{fsys/40}{datarate * cpb * samplesPerChip} \right\rceil, 69\right) * datarate * cpb} > 16.5 \\ \min\left(\left\lfloor \dfrac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right), \quad else \end{cases}$$

$$mf = Round\left(\frac{fsys/40}{\text{decimation} * cpb * datarate}\right) - 1$$

$$predec\_max\ \_bs = \begin{cases} 31 * decimation^2, & if\ decimation > 1 \\ 8 * 31, & else \end{cases}$$

$$pdscale_{ASK} = \left\lfloor \log_2 \frac{4095}{predec\_max\ \_bs} \right\rfloor + 10$$

$$src = 0$$

Mixed Mode (modtype>1):

$$decimation = \begin{cases} \min\left(\left\lceil \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rceil, 69\right), & if \frac{fsys/40}{\min\left(\left\lfloor \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right) * datarate * cpb} > 16.5 \\ \min\left(\left\lfloor \frac{fsys/40}{datarate * cpb * samplesPerChip} \right\rfloor, 69\right), & else \end{cases}$$

$$\text{predec\_max\_bs\_offset} = \text{foffsetmax} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

$$\text{predec\_max\_bs\_fdev} = \text{fdev} * \frac{64}{\text{fsys}/40000} * \text{decimation}^2$$

$$pdscale_{FSK} = \left\lfloor \log_2 \frac{4095}{\text{predec\_max\_bs\_offset} + 1.4 * \text{predec\_max\_bs\_fdev}} \right\rfloor + 10$$

$$predec\_max\_bs = \begin{cases} 31 * decimation^2, & if\ decimation > 1 \\ 8 * 31, & else \end{cases}$$

$$pdscale_{ASK} = \left\lfloor \log_2 \frac{4095}{predec\_max\_bs} \right\rfloor + 10$$

$$mf = Round\left(\frac{fsys/40}{\text{decimation} * cpb * datarate}\right) - 1$$

$$src = 0$$

2.3 Modifications for Jitter optimized Datarate Calculation

This feature is available for Explorer version equal or greater than B12.6.45.
These values are affected by activation (option at Wizard Page 1) of the jitter optimized datarate calculation:

Input variables and constants:

- Datarate oversampling factor: drOversampling
- Datarate: datarate [bit/s] := datarate * drOversampling (the datarate input is multiplied by the oversampling factor
- Samples per chip: samplesPerChip := 8 (also for TDA5225)

$$src = \begin{cases} 0, & for\ TDA5225 \\ 2, & else \end{cases}$$

3 Wizard Page 6 – Clock Data Recovery

Input variables and constants:

- Equal bits of pattern A: eqbitsA := number of equal chips within the TSI pattern A, beginning from the least significant bit of the register (SFR_%Page%_TSIPTA0), divided by two
- Maximum length of code violation within datapacket: cv [bit] | min=1 < max=11
- TSI mode: tsimode = (SFR_%Page%_TSIMODE.TSIDETMOD)

Output variables and registers allocation:

- Timing Violation Window Length: tv = SFR_%Page%_TVWIN.TVWIN

$$tv = \begin{cases} Round\left(\max\left((16 + cv * 16) * 1.25, (32 + 16 * eqbitsA) * 1.25\right)\right), & if\ tsimode = 2 \\ Round\left((16 + cv * 16) * 1.25\right), & else \end{cases}$$

4 Wizard Page 7 – Frame Synchronization Unit

Input variables and constants:

- Equal bits of pattern A: eqbitsA := number of equal chips within the TSI pattern A, beginning from the least significant bit of the register (SFR_%Page%_TSIPTA0), divided by two
- Gap time between TSIA+TSIB: gaptime [bit] | min=0 < max=15.5
- TSI Gap resync mode: tsiresync = (SFR_%Page%_TSIMODE.TSIGRSYN)

Output variables and registers allocation:

- Gap time T2: tsigapt2 = (SFR_%Page%_TSIGAP.TSIGAP)
- Gap time T16: tsigap16 = (SFR_%Page%_TSIGAP.GAPVAL)

$$tsigap2 = \lfloor 2 * \max(0, gaptime - \mathrm{eqbitsA}) \rfloor$$

$$tsigap16 = \begin{cases} \lfloor 8 * (2 * \max(0, gaptime - eqbitsA) - \lfloor 2 * \max(0, gaptime - eqbitsA) \rfloor) \rfloor, & if\ tsiresync = 0 \\ 0, else & \end{cases}$$

5 Wizard Page 9 – Polling Timer Unit

Input variables and constants:

- System frequency; $\mathrm{fsys} = 21948717\ [\mathrm{Hz}]$
- Time base: trt [ms]
- Wake-Up level observation time: wulot_time [ms]
- Timeout SYNC: timeout_sync [ms]
- Timeout TSI: timeout_tsi [ms]
- Timeout EOM: timeout _eom [ms]
- On time (Config. %Page%): ontime_x [ms]
- Off time: offtime [ms]
- Runin length: runlen = (SFR_%Page%_CDRRI.RUNLEN)

- Datarate: datarate [bit/s] (taken from Wizard page 5 – Digital Receiving Unit)

Output variables and registers allocation:

- Startup time; $st = Round\left(\frac{64000}{fsys} * 156, 4\right) [ms]$
- Channel hop time: $cht = 0.111\ [ms]$
- Self polling mode reference timer: spmrt = (SFR_SPMRT) | min=1 < 255 < max=0
- Wake-Up Level Observation Time: wulot = (SFR_%Page%_WULOT.WULOT) | min=1 < 31 < max=32(=regval=0)
- Wake-Up Level Observation Time PreScaler: wulotps = (SFR_%Page%_WULOT.WULOTPS)
- Timeout SYNC: totim_sync = (SFR_%Page%_TOTIM_SYNC) | min=1 < max=255
- Timeout TSI: totim_sync = (SFR_%Page%_TOTIM_TSI) | min=1 < max=255
- Timeout EOM: totim_sync = (SFR_%Page%_TOTIM_EOM) | min=1 < max=255
- On time (Config. %Page%): $spmont_x$ = (SFR_SPMONT%Page%1,SFR_SPMONT%Page%0) | min=1 < 16383 < max=16384(=regval=0)
- Off time: spmofft = (SFR_SPMOFFT1,SFR_SPMOFFT0) | min=1 < 16383 < max=16384(=regval=0)
- Sync Search Timeout: sysrcto = (SFR_%Page%_SYSRCTO) | min=0 < max=255

$$spmrt = \begin{cases} \left\lfloor (fsys * \frac{trt}{1000})/64 \right\rfloor + 1, if \left| \left\lfloor (fsys * Round\left(64 * \frac{\left\lfloor (fsys * \frac{trt}{1000})/64 \right\rfloor}{fsys}, 4\right))/64 \right\rfloor - \left\lfloor (fsys * \frac{trt}{1000})/64 \right\rfloor \right| > 0.00005 \\ \left\lfloor (fsys * \frac{trt}{1000})/64 \right\rfloor, \quad else \end{cases}$$

$$wulotps = \max\left(\lfloor \log_2(wulot\_time * fsys/1024000) - 2 \rfloor, 0\right)$$

$$wulot = \left\lceil \frac{wulot\_time}{1000} / (\frac{64}{fsys} * 2^{2+wulotps}) \right\rceil$$

$$totim\_sync = \lceil Round(timeout\_sync * fsys/32768000, 4) \rceil$$

$$totim\_tsi = \lceil Round(timeout\_tsi * fsys/32768000, 4) \rceil$$

$$totim\_eom = \lceil Round(timeout\_eom * fsys/65536000, 4) \rceil$$

$$spmont_x = \begin{cases} \left\lfloor \frac{ontime_x}{trt} \right\rfloor + 1, if\ Round\left(\left\lfloor \frac{ontime_x}{trt} \right\rfloor * trt, 4\right) - ontime < -0.00005 \\ \lfloor ontime_x/trt \rfloor, else \end{cases}$$

$$spmofft = \begin{cases} \left\lceil \frac{offtime}{trt} \right\rceil - 1, if\ Round\left(\left\lceil \frac{offtime}{trt} \right\rceil * trt, 4\right) - ontime > 0.00005 \\ \lceil offtime/trt \rceil, else \end{cases}$$

$$sysrcto = \lceil ((0.0000125/(1/datarate)) + 4 + (runin/2)) * 16 * 1.1 \rceil$$

Explorer version <= B12.6.33 used $sysrcto_{old}$ instead of $sysrcto$

$$sysrcto_{old} = \lceil ((0.0000125/(1/2000)) + 4 + (runin/2)) * 16 * 1.1 \rceil$$

6 Wizard Page 10 – AFC AGC

Input variables and constants:

- System frequency; $fsys = 21948717$ [Hz]
- Datarate: datarate [bit/s] (taken from Wizard page 5 – Digital Receiving Unit)
- Chip per bit selection: cpb := 1 if one chip per bit is selected, 2 if two chip per bit is selected (taken from Wizard page 5 – Digital Receiving Unit)
- Settling time; st ∊ {ultra_fast, fast, normal, slow, very_slow}
- RF channel frequency for channel ch: rf(ch) [MHz]; ch ∊ Channels; Channels = {1, 2, 3}
- RXRF Receive Side Band Select: sbsel = (SFR_%Page%_IF1.SSBSEL)
- AFC Limit: afclim = (SFR_%Page%_AFCLIMIT.AFCLIMIT)

Intermediate variables:

- Single/Double conversion frequency: conv [MHz] (taken from section 1 - Wizard Page 2 – RF PLL)

Output variables and registers allocation:

- Integrator 1 Gain Coefficient; coeff1 = (SFR_%Page%_AFCK1CFG1,SFR_%Page%_AFCK1CFG0)
- Integrator 2 Gain Coefficient: coeff2 = (SFR_%Page%_AFCK2CFG1,SFR_%Page%_AFCK2CFG0)
- Warning message "For a proper function of AFC it is recommended to change the LO injection side or to use a smaller AFC limit value.": warning := $warning = warning(ch => 1) \vee warning(ch => 2) \vee warning(ch => 3)$

$$coeff1 = \begin{cases} \dfrac{datarate * cpb}{10 * 2}, if\ st = ultra\_fast \\ \dfrac{datarate * cpb}{25 * 2}, if\ st = fast \\ \dfrac{datarate * cpb}{50 * 2}, if\ st = normal \\ \dfrac{datarate * cpb}{75 * 2}, if\ st = slow \\ \dfrac{datarate * cpb}{100 * 2}, if\ st = very\_slow \end{cases}$$

Explorer version <= B12.6.33 used $coeff1_{old}$ instead of $coeff1$

$$coeff1_{old} = \begin{cases} \dfrac{datarate * (3 - cpb)}{10}, if\ st = ultra\_fast \\ \dfrac{datarate * (3 - cpb)}{25}, if\ st = fast \\ \dfrac{datarate * (3 - cpb)}{50}, if\ st = normal \\ \dfrac{datarate * (3 - cpb)}{75}, if\ st = slow \\ \dfrac{datarate * (3 - cpb)}{100}, if\ st = very\_slow \end{cases}$$

$$coeff2 = coeff1$$

$$warning(ch) = \begin{cases} shown, & if \left\lfloor \frac{rf(ch) - \left(\left((sbsel * 2) - 1\right) * -conv\right) + \frac{afclim * fsys}{1024000000}}{fsys/1000000} \right\rfloor \neq \left\lfloor \frac{rf(ch) - \left(\left((sbsel * 2) - 1\right) * -conv\right)}{fsys/1000000} \right\rfloor \\ & \vee \left\lfloor \frac{rf(ch) - \left(\left((sbsel * 2) - 1\right) * -conv\right) - \frac{afclim * fsys}{1024000000}}{fsys/1000000} \right\rfloor \neq \left\lfloor \frac{rf(ch) - \left(\left((sbsel * 2) - 1\right) * -conv\right)}{fsys/1000000} \right\rfloor \\ hidden, & else \end{cases}$$

w w w . i n f i n e o n . c o m

Published by Infineon Technologies AG