BASIC\_CTRL IP SPEC

Table of Contents

[Introduction 1](#_Toc117684291)

[Feature 1](#_Toc117684292)

[Register Definition 3](#_Toc117684293)

[Register Map 3](#_Toc117684294)

[Functional Details 4](#_Toc117684295)

[Block Diagram 4](#_Toc117684296)

## Introduction

The BASIC\_CTRL module is used to convert input dual daisy chain signals or input SPI signal to byte data rx\_data[8:0], and convert byte data tx\_data[8:0] to dual daisy chain outputs or SPI output.

The DS\_BASIC module is used to convert input dual daisy chain signals to byte data rx\_data[8:0], and convert byte data tx\_data[8:0] to dual daisy chain outputs.

The SPI\_BASIC module is used only for bridge application. It converts input SPI signal to rx\_data[8:0], or converts tx\_data[8:0] to SPI output.

## Feature

Key features of the BASIC\_CTRL module are:

• Adjust for both AFE and bridge application

•Used as AFE: SPI\_EN low

•Used as bridge: SPI\_EN high

• Direction control

Key features of the DS\_BASIC module are:

• Analysis received daisy chain data

• Send daisy chain data

• FLT\_WAKE receiving and indicating in daisy chain communication

• Direction control

Key features of the SPI\_BASIC module are:

• Analysis received SPI data

• Send SPI data

## Register Definition

### Register Map

Table 1 BASIC\_CTRL Register Map

| **Name** | **Add** | **D7** | **D6** | **D5** | **D4** | **D3** | **D2** | **D1** | **D0** | **Default** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| COMM\_CONF2 | 0x0003 | COMN\_TX\_DIS | COMS\_TX\_DIS | STACK\_RESP\_CMD<5:0> | | | | | | 00 |
| CTRL2 | 0x2003 |  |  |  |  |  | CMP\_BIST\_GO | ADD\_W\_EN | SPI\_DIR |  |

## Functional Details

### Block Diagram

The following diagram shows the BASIC\_CTRL architecture and internal modules and connections.



Figure 1 BASIC\_CTRL diagram

### Module input/output list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Dir | Width | Discirption | duration |
| rev\_dsy\_data | O | 9 | received daisy chain data | Level(32M domain) |
| neg\_rx\_en\_dsy | O | 1 | negedge of rx\_en\_dsy | 4 CLK\_32M |
| neg\_rx\_en\_dsy\_8M | O | 1 | negedge of rx\_en\_s\_dsy or rx\_en\_n\_dsy | 1 CLK\_REG |
| neg\_rx\_en\_s\_dsy | O | 1 | negedge of rx\_en\_s\_dsy | 4 CLK\_32M |
| neg\_rx\_en\_n\_dsy | O | 1 | negedge of rx\_en\_n\_dsy | 4 CLK\_32M |
| MISS | O | 1 |  |  |
| ORDER | O | 1 |  |  |
| SYNCT | O | 1 |  |  |
| SYNCD | O | 1 |  |  |
| BIT | O | 1 |  |  |
| WR\_COM\_PLUS | O | 1 | daisy chain output data in positive phase | 8 CLK\_32M |
| WR\_COM\_MINUS | O | 1 | daisy chain output data in negtive phase | 8 CLK\_32M |
| TX\_EN\_S | O | 1 | enable daisy chain transmitting on S port |  |
| TX\_EN\_N | O | 1 | enable daisy chain transmitting on N port |  |
| D2A\_RX\_EN\_S | O | 1 | enable daisy chain receiving on S port |  |
| D2A\_RX\_EN\_N | O | 1 | enable daisy chain receiving on N port |  |
| send\_char\_end\_pos | O | 1 | mark byte transmitting end time | 4 CLK\_32M |
| tx\_crc | O | 16 | crc16 result of tx\_one |  |
| rx\_en\_n | O | 1 | daisy chai signal is being received on N port |  |
| rx\_en\_s | O | 1 | daisy chai signal is being received on S port |  |
| rx\_en | O | 1 | daisy chai signal is being received |  |
| tx\_en\_32M | O | 1 | sync send\_start with CLK\_32M | 2 CLK\_32M |
| neg\_TX\_EN\_S | O | 1 | negedge of TX\_EN\_S | 1 CLK\_32M |
| neg\_TX\_EN\_N | O | 1 | negedge of TX\_EN\_N | 1 CLK\_32M |
| clr\_crc\_dsy | O | 1 | Daisy chain crc clear | 3~4 CLK\_32M |
| clr\_crc\_spi | O | 1 | Spi crc clear, In RX state when the falling edge of A2D\_SPI\_CSB | 1 CLK\_REG |
| rx\_data | O | 9 | Received data from daisy chain or SPI | Level(32M domain) |
| TX\_timeout | O | 1 | no data to tranmit for a timeout time when TX\_EN\_X high |  |
| pos\_TBYTE\_FAST | O | 1 | fault flag: receiving data is too fast | 4 CLK\_32M |
| pos\_TBYTE\_TO | O | 1 | fault flag: receiving data is too slow | 4 CLK\_32M |
| D2A\_SPI\_SPI\_RDY | O | 1 | Indicate slave can be read or write by master |  |
| D2A\_SPI\_MISO | O | 1 | Master input slave output, MSB first |  |
| SPI\_RX\_EN | O | 1 | Indicate RX\_DATA\_SPI is update | 1 CLK\_REG |
| SPI\_CLR\_DET | O | 1 | CLR\_DET module detect COMM\_CLEAR command after detect falling edge and then receive 8'h00 | 1 CLK\_REG |
| RX\_FIFO\_OF | O | 1 |  | 1 CLK\_REG |
| TX\_FIFO\_OF | O | 1 |  | 1 CLK\_REG |
| TX\_FIFO\_UF | O | 1 |  | 1 CLK\_REG |
| TX\_DONE | O | 1 | All TX FIFOis empty and timeout | 1 CLK\_REG |
| rst\_spi | O | 1 | Reset tx logics when SPI\_EN high | 1 CLK\_REG |
| CLK\_32M\_SC | I | 1 | CLK\_32M after scan mux |  |
| resetb\_CLK | I | 1 | Asynchronous reset signal(synchronously released) |  |
| rstb\_32M\_ok\_and\_sr | I | 1 | CLK\_32M\_OK low or soft reset |  |
| CLK\_REG\_SC | I | 1 | Scan-mux result of 8MHz clock from CLK\_32M | 8MHz |
| CLK\_REG | I | 1 | 8MHz clock divided from CLK\_32M | 8MHz |
| SOFT\_RSTB\_32M | I | 1 | Soft reset |  |
| SLEEP\_MODE | I | 1 | Synchronous result of A2D\_SLEEP\_1P8 by CLK\_256K\_SC | Level |
| rx\_en\_256K | I | 1 | Daisy chain or spi rx\_en | Level(CLK\_256K domain) |
| A2D\_COM\_PLUS\_S | I | 1 | Positive input comparator in S port | async |
| A2D\_COM\_MINUS\_S | I | 1 | Negative input comparator in S port | async |
| A2D\_COM\_PLUS\_N | I | 1 | Positive input comparator in N port | async |
| A2D\_COM\_MINUS\_N | I | 1 | Negative input comparator in N port | async |
| TONE\_TRANS\_EN\_N | I | 1 | N port tone transmission enable | Level(CLK\_OUT domain) |
| TONE\_TRANS\_EN\_S | I | 1 | S port tone transmission enable | Level(CLK\_OUT domain) |
| state\_tx\_init | I | 1 | tx\_state is STATE\_INIT | 1 CLK\_REG |
| state\_tx\_pec | I | 1 | tx\_state is STATE\_PEC | 1 CLK\_REG |
| state\_rx\_init | I | 1 | state is STATE\_INIT | 1 CLK\_REG |
| state\_rx\_bps | I | 1 | state is STATE\_BYPASS | 1 CLK\_REG |
| response | I | 1 | Current device response | Level(8M domain) |
| pos\_response | I | 1 | Positive edge of response | 1 CLK\_REG |
| neg\_response | I | 1 | Negative edge of response | 1 CLK\_REG |
| pos\_next\_rps | I | 1 | Current device is the next to response | 1 CLK\_32M |
| bypass\_end | I | 1 | Mark the ending time of a bypass byte | 1 CLK\_REG |
| rx\_dev\_addr | I | 1 | Receive 9’h1C0 when state is STATE\_INT or STATE\_BYPASS | 4 CLK\_32M |
| cnt\_rx\_byte\_num | I | 8 | Rx byte numer | Level(8M domain) |
| rd | I | 1 | Current device in read station | Level(8M domain) |
| D2A\_TOP\_DEV | I | 1 | Current device is fastest from bridge | Level(8M domain) |
| stack | I | 1 | Stack operation | Level(8M domain) |
| COMN\_TX\_DIS | I | 1 | N port transmit disable | Level(8M domain) |
| COMS\_TX\_DIS | I | 1 | S port transmit disable | Level(8M domain) |
| wait\_re\_clocking | I | 14 | Wait time before transmitting | CLK\_REG domain |
| adr\_idty\_done | I | 1 | Address identify done | Level(8M domain) |
| tail\_blanking | I | 1 | Tail blanking time | Level(8M domain) |
| neg\_rx\_en | I | 1 | Negedge of rx\_en | 1 CLK\_REG |
| next\_rps | I | 1 | Current device is the next to response | Level(8M domain) |
| neg\_tx\_init | I | 1 | Pulse after tx\_state jump to STATE\_INIT from STATE\_PEC | 1 CLK\_REG |
| STACK\_RESPONSE | I | 6 | Internal time between response bytes | Level(8M domain) |
| FRAME\_DONE | I | 9 | Received frame done | 1 CLK\_REG |
| FR\_CRC\_FLT | I | 1 | Frame CRC fault | 1 CLK\_REG |
| A2D\_SPI\_SCLK | I | 1 | Spi clock input | N/A |
| A2D\_SPI\_MOSI | I | 1 | Master output slave input, MSB first | N/A |
| A2D\_SPI\_CSB | I | 1 | Chip selection input | N/A |
| tx\_data | I | 9 | Data to be transmit | CLK\_REG domain |
| tx\_start | I | 1 | Transmitting start | 1 CLK\_REG |
| tx\_capture | I | 1 | Tx\_data enable | 1 CLK\_REG |
| SPI\_EN | I | 1 | Enable SPI | Async |
| SPI\_DIR | I | 1 | "1" for north interface, "0" for south interface | 1 CLK\_REG |
| dev\_addr\_dlv | I | 1 | Device address identify delivery | Level(8M domain) |
| dev\_addr\_dlv\_spi | I | 1 | Device address identify delivery when SPI\_EN high | Level(8M domain) |
| RESP | I | 1 | RESP =1 , indicate is maser read state RESP =0 , indicate is master write state | Level(8M domain) |
| RD\_DET | I | 1 | Device address identify delivery when SPI\_EN high | Level(8M domain) |
| COPY\_NXT | I | 1 | SPI to send next data to COMM\_CTRL | Level(8M domain) |

#### Clock Domain

The clock for BASIC\_CTRL is CLK\_32M\_SC and A2D\_SPI\_SCLK.

For DS\_BASIC, CLK\_32M\_SC is used.

For SPI\_BASIC, both CLK\_32M\_SC and A2D\_SPI\_SCLK are used.

### BASIC\_CTRL function description

BASIC\_CTRL module is the top module that instanced DS\_BASIC and SPI\_BASIC.

Only 2 signals are generated in BASIC\_CTRL:

Rx\_data(HWR001\_BASIC\_CTRL, HWR003\_BASIC\_CTRL):

When receiving data from SPI interface(SPI\_EN high, and the corresponding D2A\_RX\_EN\_x depended on SPI\_DIR is high), choose rx\_data\_spi[8:0] as input data; else choose rev\_dsy\_data[8:0] as input data.

Rst\_spi:

logic and result of SPI\_EN and SPI\_CLR\_DET.

Outputs TX\_EM\_S, TX\_EN\_N, WR\_COM\_PLUS and WR\_COM\_MINUS are from DS\_BASIC directly. These 4 signals can transmit data at right directions.(HWR001\_BASIC\_CTRL)

(Unrealized)

(HWR002\_BASIC\_CTRL) only sub-module DS\_BASIC can be synchronously reset when CLK\_32M\_OK is low.

Sun-module SPI\_BASIC cannot.