28.6 Broadcast Module (BRC)

28.6.1 Overview

Since each write address for the sub-module channels of the GTM that are able to write to the ARU can only be read by a single module, it is impossible to provide a data stream to different modules in parallel (This statement holds not for sources, which do not invalidate their data after the data were read by any consumer, e.g. DPLL).

To overcome this issue for regular modules, the sub-module Broadcast (BRC) enables to duplicate data streams multiple times.

The BRC sub-module provides 12 input channels as well as 22 output channels.

In order to clone an incoming data stream, the corresponding input channel can be mapped to zero or more output channels.

When mapped to zero no channel is read.

To destroy an incoming data stream, the **EN_TRASHBIN** bit inside the **BRC_SRC_[x]_DEST** register has to be set.

The total number of output channels that are assigned to a single input channel is variable. However, the total number of assigned output channels must be less than or equal to 22.

28.6.2 BRC Configuration

As it is the case with all other sub-modules connected to the ARU, the input channels can read arbitrary ARU address locations and the output channels provide the broadcast data to fixed ARU write address locations.

The associated write addresses for the BRC sub-module are fixed and can be obtained from the product specific appendix.

The read address for each input channel is defined by the corresponding register **BRC_SRC_[x]_ADDR** (x: 0…11).

The mapping of an input channel to several output channels is defined by setting the appropriate bits in the register **BRC_SRC_[x]_DEST** (x: 0…11). Each output channel is represented by a single bit in the register **BRC_SRC_[x]_DEST**.

If no output channel bit is set within a register **BRC_SRC_[x]_DEST**, no data is provided to the corresponding ARU write address location from the defined read input specified by **BRC_SRC_[x]_ADDR**. This means that the channel does not broadcast any data and is disabled (reset state).

Besides the possibility of mapping an input channel to several output channels, the bit **EN_TRASHBIN** of register **BRC_SRC_[x]_DEST** may be set, which results in dropping an incoming data stream. In this case the data of an input channel defined by **BRC_SRC_[x]_ADDR** is consumed by the BRC module and not routed to any succeeding sub-module. In consequence, the output channels defined in the register **BRC_SRC_[x]_DEST** are ignored. Therefore, the bits 0 to 21 are set to zero (0) when trash bin functionality is enabled.

In general, the BRC sub-module can work in two independent operation modes. In the first operation mode the data consistency is guaranteed since a BRC channel requests only new data from a source when all destination channels for the BRC have consumed the old data value. This mode is called *Data Consistency Mode* (DCM).

In a second operation mode the BRC channel always requests data from a source and distributes this data to the destination regardless whether all destinations have already consumed the old data. This mode is called *Maximum Throughput Mode* (MTM).

MTM ensures that always the newest available data is routed through the system, while it is not guaranteed data consistency since some of the destination channels can be provided with the old data while some other destination channels are provided with the new data. If this is the case, the Data Inconsistency Detected Interrupt *BRC_DID_IRQ[x]* is raised but the channel continues to work.

Furthermore in MTM mode it is guaranteed that it is not possible to read a data twice by a read channel. This is blocked.

The channel mode can be configured inside the **BRC_SRC_[x]_ADDR** register.

To avoid invalid configurations of the registers **BRC_SRC_[x]_DEST**, the BRC also implements a plausibility check for these configurations. If the software assigns an already used output channel to a second input channel, BRC performs an auto correction of the lastly configured register **BRC_SRC_[x]_DEST** and it triggers the interrupt *BRC_DEST_ERR*.

Consider the following example for clarification of the auto correction mechanism. Assume that the following configuration of the 22 lower significant bits for the registers **BRC_SRC_[x]_DEST**:

BRC_SRC_0_DEST: 00 0000 0000 1000 1000 0000 (binary)

BRC_SRC_1_DEST: 00 0000 0000 0100 0000 0100 (binary)

BRC_SRC_2_DEST: 00 0000 0000 0001 0100 0010 (binary)

BRC_SRC_3_DEST: 00 0000 0000 0010 0001 1001 (binary)

If the software overwrites the value for register **BRC_SRC_2_DEST** with

BRC_SRC_2_DEST: 00 0000 0000 1001 0010 0010 (binary)

(changed bits are underlined), then the BRC releases a *BRC_DEST_ERR* interrupt since bit 11 is already assigned in register **BRC_SRC_0_DEST**. The auto correction forces bit 11 to be cleared. The modifications of the bits 5 and 6 are accepted, since there is no violation with previous configurations. So the result of the write access mentioned above results in the following modified register configuration:

BRC_SRC_2_DEST: 00 0000 0000 0001 0010 0010 (binary)

For debug purposes, the interrupt *BRC_DEST_ERR* can also be released by writing to register **BRC_IRQ_FORCINT**. Nevertheless, the interrupt has to be enabled to be visible outside of the GTM.

28.6.3 BRC Interrupt Signals

Table 15 BRC Interrupt Signals

Signal	Description
BRC_DEST_ERR_IRQ	Indicating configuration errors for BRC module
BRC_DID_IRQ[x]	Data inconsistency occurred in MTM mode (x:0…11)

28.6.4 BRC Configuration Register Overview

Table 16 BRC Configuration Register Overview

Register Name	Description	see Page
BRC_SRC_[z]_ADDR	BRC read address for input channel z	74
BRC_SRC_[z]_DEST	BRC destination channels for input channel z	75
BRC_IRQ_NOTIFY	BRC interrupt notification register	76
BRC_IRQ_EN	BRC interrupt enable register	77
BRC_EIRQ_EN	BRC error interrupt enable register	79
BRC_IRQ_FORCINT	BRC force interrupt register	78

Table 16 BRC Configuration Register Overview (cont'd)

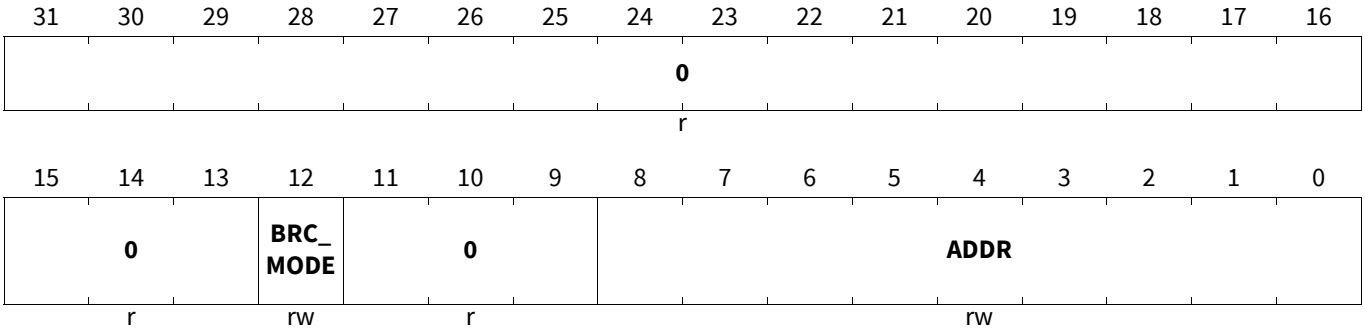
Register Name	Description	see Page
BRC_RST	BRC software reset register	80
BRC_IRQ_MODE	BRC interrupt mode configuration register	78

28.6.5 BRC Configuration Register Description

28.6.5.1 Register BRC_SRC_[z]_ADDR

BRC Read Address for Input Channel z

BRC_SRC_z_ADDR (z=0-11)
BRC Read Address for Input Channel z **(000400_H+z*8)** **Application Reset Value: 0000 $01FE_H$**



Field	Bits	Type	Description
ADDR	8:0	rw	**Source ARU address. Defines an ARU read address used as data source for input channel z** This bit field is only writable if channel is disabled.
BRC_MODE	12	rw	**BRC_MODE: BRC Operation mode select** This bit field is only writable if channel is disabled. 0_B Consistency Mode (DCM) selected 1_B Maximum Throughput Mode (MTM) selected
0	11:9, 31:13	r	**Reserved** Read as zero, shall be written as zero.

28.6.5.2 Register BRC_SRC_[z]_DEST

BRC Destination Channels for Input Channel z

BRC_SRC_z_DEST (z=0-11)

BRC Destination Channels for Input Channel z(000404_H+z*8) **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0									EN_TRASHBIN	EN_DEST21	EN_DEST20	EN_DEST19	EN_DEST18	EN_DEST17	EN_DEST16
r									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN_DEST15	EN_DEST14	EN_DEST13	EN_DEST12	EN_DEST11	EN_DEST10	EN_DEST9	EN_DEST8	EN_DEST7	EN_DEST6	EN_DEST5	EN_DEST4	EN_DEST3	EN_DEST2	EN_DEST1	EN_DEST0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
EN_DESTq (q=0-21)	q	rw	**Enable BRC destination address q** The bits 0 to 21 are cleared by auto correction mechanism if a destination channel is assigned to multiple source channels. When a BRC input channel is disabled (all EN_DESTq (q: 0...21) bits are reset to zero), the internal states are reset to their reset value. 0_B Destination address q not mapped to source BRC_SRC_[x]_ADDR 1_B Destination address q mapped to source BRC_SRC_[x]_ADDR
EN_TRASHBIN	22	rw	**EN_TRASHBIN: Control trash bin functionality** When bit **EN_TRASHBIN** is enabled bits 0 to 21 are ignored for this input channel. Therefore, the bits 0 to 21 are set to zero (0) when trash bin functionality is enabled. 0_B Trash bin functionality disabled 1_B Trash bin functionality enabled
0	31:23	r	**Reserved** Read as zero, shall be written as zero.

28.6.5.3 Register BRC_IRQ_NOTIFY

BRC Interrupt Notification Register

BRC_IRQ_NOTIFY
BRC Interrupt Notification Register **(000460_H)** **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0			DID11	DID10	DID9	DID8	DID7	DID6	DID5	DID4	DID3	DID2	DID1	DID0	DEST_ERR
r			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
DEST_ERR	0	rw	**Configuration error interrupt for BRC sub-module** This bit will be cleared on a CPU write access of value '1'. (As the bit is rw, otherwise no clear.) A read access leaves the bit unchanged. 0_B No BRC configuration error occurred 1_B BRC configuration error occurred
DIDx (x=0-11)	x+1	rw	**Data inconsistency occurred for channel x in MTM mode** This bit will be cleared on a CPU write access of value '1'. A read access leaves the bit unchanged.
0	31:13	r	**Reserved** Read as zero, shall be written as zero

28.6.5.4 Register BRC_IRQ_EN

BRC Interrupt Enable Register

BRC_IRQ_EN
BRC Interrupt Enable Register **(000464_H)** **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0			DID_IRQ_EN11	DID_IRQ_EN10	DID_IRQ_EN9	DID_IRQ_EN8	DID_IRQ_EN7	DID_IRQ_EN6	DID_IRQ_EN5	DID_IRQ_EN4	DID_IRQ_EN3	DID_IRQ_EN2	DID_IRQ_EN1	DID_IRQ_EN0	DEST_ERR_IRQ_EN
r			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
DEST_ERR_IRQ_EN	0	rw	**BRC_DEST_ERR_IRQ interrupt enable** 0_B Disable interrupt, interrupt is not visible outside GTM 1_B Enable interrupt, interrupt is visible outside GTM
DID_IRQ_ENx (x=0-11)	x+1	rw	**Enable DID interrupt for channel x** 0_B Disable interrupt, interrupt is not visible outside GTM 1_B Enable interrupt, interrupt is visible outside GTM
0	31:13	r	**Reserved** Read as zero, shall be written as zero

28.6.5.5 Register BRC_IRQ_FORCINT

BRC Force Interrupt Register

BRC_IRQ_FORCINT
BRC Force Interrupt Register **(000468$_H$)** **Application Reset Value: 0000 0000$_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0			TRG_DID11	TRG_DID10	TRG_DID9	TRG_DID8	TRG_DID7	TRG_DID6	TRG_DID5	TRG_DID4	TRG_DID3	TRG_DID2	TRG_DID1	TRG_DID0	TRG_DEST_ERR
r			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
TRG_DEST_ERR	0	rw	**Trigger destination error interrupt** This bit is cleared automatically after write. This bit is write protected by bit RF_PROT of register GTM_CTRL. 0_B Corresponding bit in status register will not be forced 1_B Assert corresponding field in BRC_IRQ_NOTIFY register
TRG_DIDx (x=0-11)	x+1	rw	**Trigger DID interrupt for channel x** This bit is cleared automatically after write. This bit is write protected by bit RF_PROT of register GTM_CTRL.
0	31:13	r	**Reserved** Read as zero, shall be written as zero

28.6.5.6 Register BRC_IRQ_MODE

BRC Interrupt Mode Configuration Register

BRC_IRQ_MODE
BRC Interrupt Mode Configuration Register **(00046C$_H$)** **Application Reset Value: 0000 0000$_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0														IRQ_MODE	
r														rw	

Field	Bits	Type	Description
IRQ_MODE	1:0	rw	**IRQ mode selection** Note: The interrupt modes are described in **Section 28.4.5**. 00_B Level mode 01_B Pulse mode 10_B Pulse-Notify mode 11_B Single-Pulse mode
0	31:2	r	**Reserved** Read as zero, shall be written as zero

28.6.5.7 Register BRC_EIRQ_EN

BRC Error Interrupt Enable Register

BRC_EIRQ_EN
BRC Error Interrupt Enable Register **(000474_H)** **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0			**DID_EIRQ_EN11**	**DID_EIRQ_EN10**	**DID_EIRQ_EN9**	**DID_EIRQ_EN8**	**DID_EIRQ_EN7**	**DID_EIRQ_EN6**	**DID_EIRQ_EN5**	**DID_EIRQ_EN4**	**DID_EIRQ_EN3**	**DID_EIRQ_EN2**	**DID_EIRQ_EN1**	**DID_EIRQ_EN0**	**DEST_ERR_EIRQ_EN**
r			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

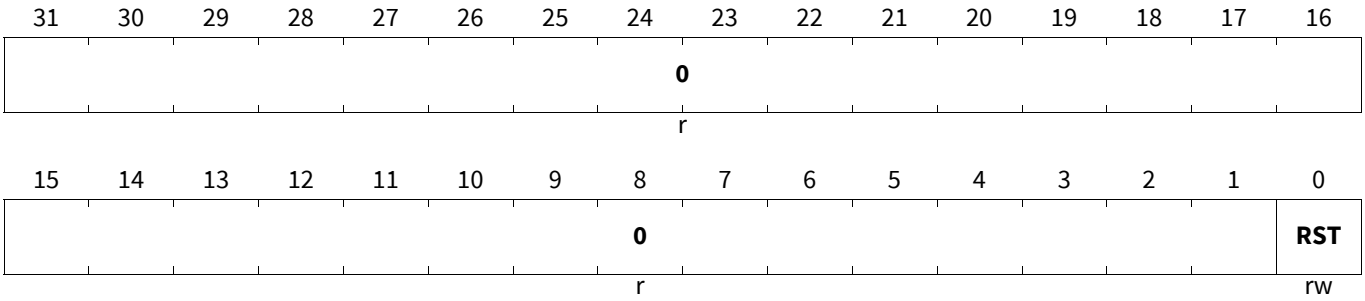
Field	Bits	Type	Description
DEST_ERR_EIRQ_EN	0	rw	**BRC_DEST_ERR_EIRQ error interrupt enable** 0_B Disable error interrupt, error interrupt is not visible outside GTM 1_B Enable error interrupt, error interrupt is visible outside GTM
DID_EIRQ_ENx (x=0-11)	x+1	rw	**Enable DID interrupt for channel x** 0_B Disable error interrupt, error interrupt is not visible outside GTM 1_B Enable error interrupt, error interrupt is visible outside GTM
0	31:13	r	**Reserved** Read as zero, shall be written as zero.

28.6.5.8 Register BRC_RST

BRC Software Reset Register

BRC_RST

BRC Software Reset Register **(000470_H)** **Application Reset Value: $0000\ 0000_H$**



Field	Bits	Type	Description
RST	0	rw	**Software reset** This bit is cleared automatically after write by CPU. The channel registers are set to their reset values and channel operation is stopped immediately. 0_B No action 1_B Reset BRC
0	31:1	r	**Reserved** Read as zero, shall be written as zero