PM7364

FREEDM

INTERFACING THE FREEDM TO THE ST-BUS

PRELIMINARY

Issue 1: February, 1997

CONTENTS

OVERVIEW

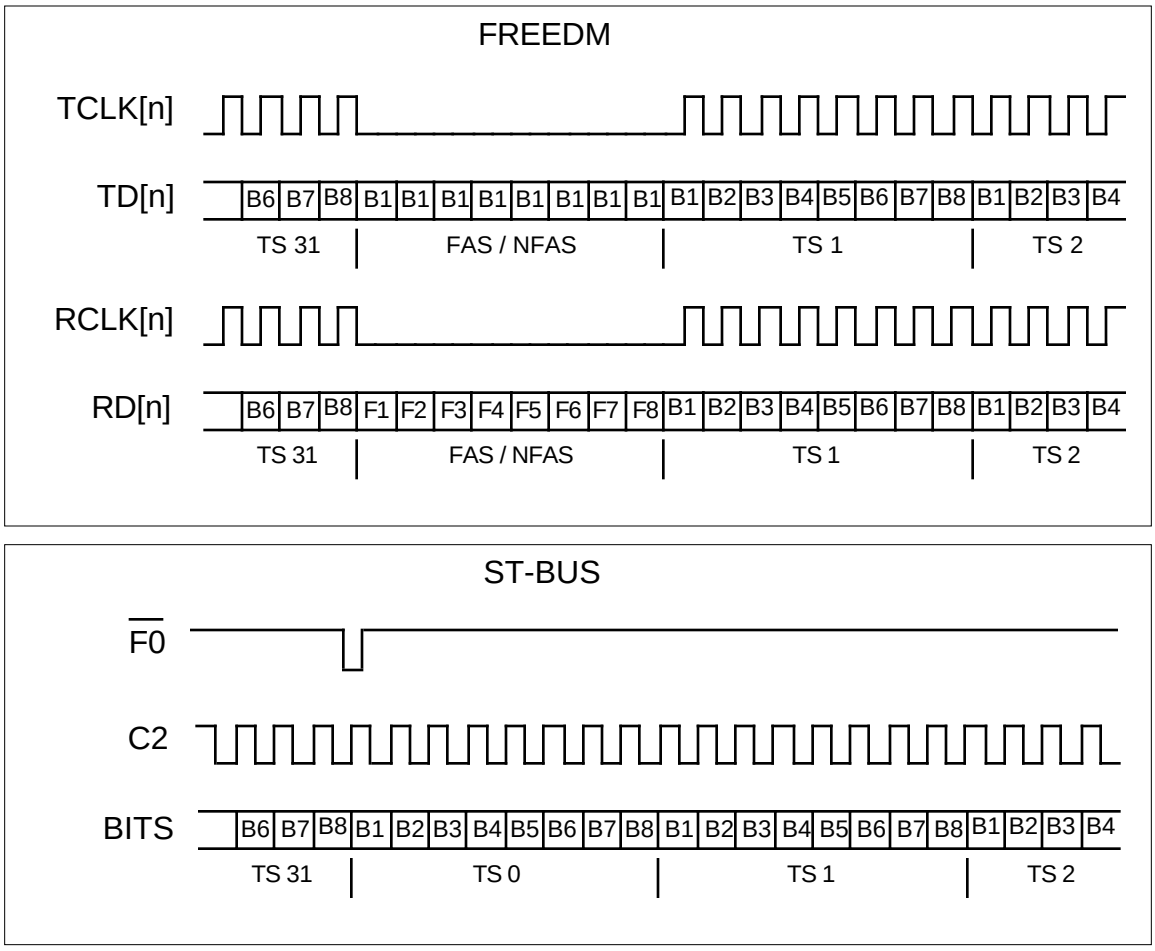
The ST-BUS (Serial Telecom Bus) is commonly used as a means of communication and information transfer between modules of a system. It is a high speed, synchronous serial bus and transports information in a digital format. This format can be interfaced to one or more of the 32 bi-directional serial ports of a FREEDM device. This application note examines the mapping of FREEDM channels to ST-BUS channels. It also describes the signals required to interface a FREEDM serial port to a ST-BUS and provides a drawing of the digital logic for the interface.

The ST-BUS supports information streams at aggregate rates of 2.048, 4.096 and 8.192 Mbit/s. It also supports two types of frame alignment. This application note addresses only the 2.048 Mbit/s information stream with type 0 frame alignment, and with a 2.048 MHz ST-BUS clock.

COMPARISON OF FREEDM AND ST-BUS DIGITAL FORMAT

A serial port on the FREEDM must be configured for E1 channelized operation to interface with the digital format of an ST-BUS. For this mode of operation the digital timing shown in figure 1 applies.

Fig. 1 Comparison of Digital Format



The ST-BUS configuration is for type 0 frame alignment and a 2.048 Mbit/s clock. The digital signal is synchronous and is composed of 125 usec frames, each with 32 time-slots. The frame alignment signal marks the end of one frame and the first time slot within the following frame. For the ST-BUS a seperate waveform provides the frame alignment signal, whereas the FREEDM achieves frame alignment by gapping of the RCLK and TCLK waveforms. The first bit within a time-slot is B1, and it is the most significant bit.

The synchronous clock frequency is the same for both formats except that the signals, RCLK and TCLK, of the FREEDM are inverted with respect to the ST-BUS clock. Also, RCLK and TCLK are gapped for frame alignment. Since the clock waveforms are inputs to the FREEDM they can be derived from the ST-BUS by inverting C2, and gapping it for the frame alignment signals (see figure 3).

Limitations of the Interface

- The ST-BUS was designed to allow nodes in a system to source and sink information on the same bus. For this reason tri-state buffers are used to prevent multiple nodes from driving the bus during the same time-slot. The FREEDM transmit ports are not tri-stated. Thus only one transmit serial port may be connected to an ST-BUS as it drives all 32 time-slots.

- Multiple FREEDM receive ports can be connected to the same ST-BUS provided the total load of the receive ports does not exceed the drive capability of the source of the data stream.

- As stated previously time-slot 0 of the ST-BUS in not available due to gapping of the RCLK and TCLK waveforms.

Mapping of FREEDM Channels to the Data Stream of the ST-BUS

A FREEDM device is capable of processing 128 transmit and 128 receive HDLC channels. Each time-slot (except time-slot 0) within the digital format can be mapped into an HDLC channel of the FREEDM, or multiple time-slots can be mapped into an HDLC channel of the FREEDM.

A receive channel is composed of time-slots within a serial data stream as per the following steps:

1) Configure the receive serial port for channelized E1 operation. Specify CEN = 1, E1 = 1, and LDLBEN = 0 within the **RCAS Link#n Configuration** register that belongs to one of the 32 receive serial ports being attached to the ST-BUS.

2) Configure the width of the frame alignment gap of RCLK. Specify FTHRES[6:0] within the **RCAS Framing Bit Threshold** register. This value is specified as a multiple of SYSCLK corresponding to the expected link bit rate. The programmed value applies to all receive links on a FREEDM.

3) For each time-slot (1 thru 31) of this link configure a FREEDM receive channel to either extract or ignore the data within the time-slot. A receive channel can extract data of just one time-slot or multiple time-slots. The following procedure should be applied for each of 31 time-slots:

To map the time-slot to a receive channel the **RCAS Indirect Channel Data** register followed by the **RCAS Indirect Link & Time-slot Select** register must be programmed. Specify the receive channel number (in the range 0 thru 127) within CHAN[6:0], CDLBEN=0, and PROV=1 for time-slot data that is extracted from the ST-BUS. If the data is not to be extracted then specify PROV=0 instead of PROV=1. Specify the time-slot (in the range 1 thru 31) within TSLOT[4:0], the link (in the range 0 thru 31) within LINK[4:0], and RWB=0 to write the values from these registers into the FREEDM configuration RAM.

A transmit channel is compsed of time-slots within a serial data stream as per the following steps:

1) Configure the transmit serial port for channelized E1 operation. Specify CEN = 1, and E1 = 1 within the **TCAS Link#n Configuration** register that belongs to one of the 32 transmit serial ports being attached to the ST-BUS.

2) Configure the width of the frame alignment gap of TCLK. Specify FTHRES[6:0] within the **TCAS Framing Bit Threshold** register. This value is specified as a multiple of SYSCLK corresponding to the expected link bit rate. The programmed value applies to all transmit links on a FREEDM.

3) Specify the value driven onto the ST-BUS for time-slots that are configured as not being driven by a transmit channel. This is done via FDATA[7:0] within the **TCAS Idle Time-Slot Data** register.

4) For each time-slot (1 thru 31) of this link, either configure a transmit channel to fill the time-slot, or configure the time-slot to be filled with an idle byte value. A transmit channel can fill data into just one time-slot or multiple time-slots. The following procedure should be applied for each of 31 time-slots:

To map the time-slot to a transmit channel the **TCAS Indirect Channel Data** register followed by the **TCAS Indirect Link & Time-slot Select** register must be programmed. Specify the transmit channel number (in the range 0 thru 127) within CHAN[6:0], and PROV=1 for time-slot data that is driven by the transmit channel. If the data is not to be taken from the transmit channel then specify PROV=0 instead of PROV=1. Specify the time-slot (in the range 1 thru 31) within TSLOT[4:0], the link (in the range 0 thru 31) within LINK[4:0], and RWB=0 to write the values from these registers into the FREEDM configuration RAM.

Table 1 shows an example for mapping of a T1 stream within a ST-BUS. In this example the TOCTL provides the T1 stream to the FREEDM via the ST-BUS. To configure the first receive channel of the FREEDM to receive a B-channel from time-slot 5 of the T1 data stream, and the ST-BUS is attached to the third serial port, the following is programmed:

1) Specify CEN=1, E1=1 and LDLEN = 0 within **RCAS Link#2 Configuration** register.

2) For a 33Mhz SYSCLK, the default value of **RCAS Framing Bit Theshold** register is valid. There is no need to program this register.

3) Specify CHAN[6:0] = 0, CDLBEN=0, and PROV=0 within the **RCAS Indirect Channel Data** register. Then for each time-slot in the range 1 thru 31 specify TSLOT[4:0] , LINK[4:0]=2, and RWB=0 within the **RCAS indirect Link & Time-slot Select** register. This will overwrite provisioning information that may have existed on receive channel 0. This channel is now unprovisioned within the RCAS block.

4) Provision receive channel 0 to extract data from time-slot 5 of the data stream as follows. Specify CHAN[6:0] = 0, CDLBEN=0, and PROV=1 within the **RCAS Indirect Channel Data** register. Then specify TSLOT[4:0]=6 (see table 1), LINK[4:0]=2, and RWB=0 within the **RCAS indirect Link & Time-slot Select** register.

Table. 1 Mapping of T1 Time-slots From TOCTL to FREEDM via the ST-BUS

PROV	TSLOT[4:0]	T1 Time-slot	PROV	TSLOT[4:0]	T1 Time-slot
1	1	1	1	17	13
1	2	2	1	18	14
1	3	3	1	19	15
0	4	n/a	0	20	n/a
1	5	4	1	21	16
1	6	5	1	22	17
1	7	6	1	23	18
0	8	n/a	0	24	n/a
1	9	7	1	25	19
1	10	8	1	26	20
1	11	9	1	27	21
0	12	n/a	0	28	n/a
1	13	10	1	29	22
1	14	11	1	30	23
1	15	12	1	31	24
0	16	n/a			

TIMING CONSIDERATIONS

The timing requirements for the FREEDM must be better than the worst case timing provided by the bus to a ST-BUS component. A comparison of worst case timing is provided in the following sections. It describes the signals required to interface a ST-BUS component to the bus, a FREEDM serial port to the bus, and then provides the timing values for comparison.

Description of ST-BUS Component Signals

C2 The 2.048 Mbit/s clock signal is an input to a ST-BUS component.

FO The framing signal is an active low pulse which marks the end of the 31st time-slot, or the start of the 1st time-slot. This signal is an input to a ST-BUS component.

STo This is the data stream output of a ST-BUS component.

STi This is the data stream input to a ST-BUS component.

Description of FREEDM Serial Port Signals

RCLK[n] The receive data clock signal is an input to a FREEDM serial port, where n is the port number. This signal is gapped for frame alignment.

TCLK[n] The transmit data clock signal is an input to a FREEDM serial port, where n is the port number. This signal is gapped for frame alignment.

TD[n] This is the data stream output by a FREEDM serial port, where n is the port number.

RD[n] This is the data stream input to a FREEDM serial port, where n is the port number.

Fig. 2 FREEDM and ST-BUS Timing

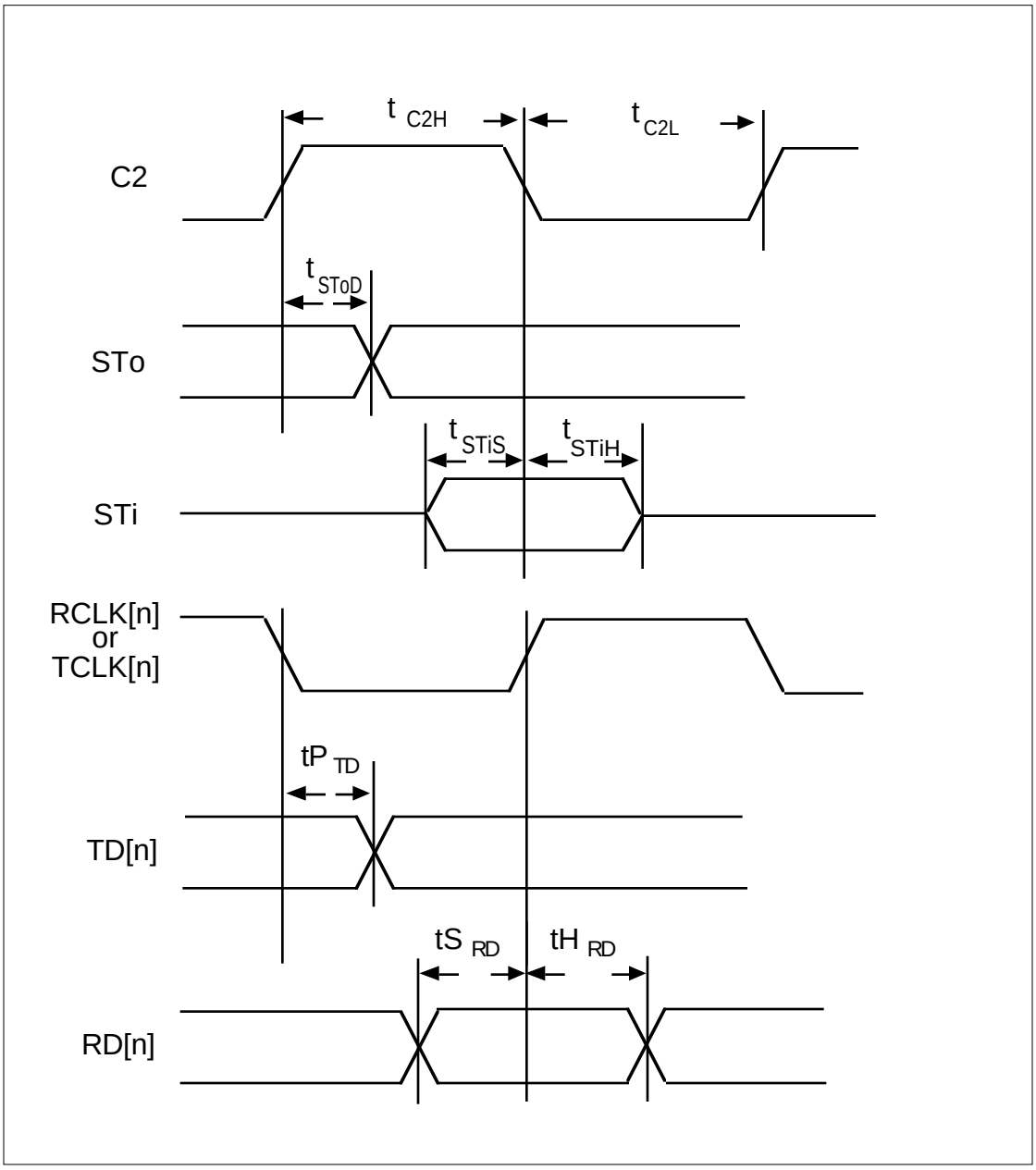


Table. 2 ST-BUS and FREEDM Component Timing Values

Characteristic	Min	Max	Units
Clock C2 High Width	220	268	ns
Clock C2 Low Width	220	268	ns
STo Delay	20	125	ns
STi Setup Time	30		ns
STi Hold Time	224		ns
TD Delay (link #0,1,2)	2	10	ns
TD Delay (link #3-31)	5	25	ns
RD Setup Time	5		ns
RD Hold Time	5		ns

Conclusions Concerning Timing

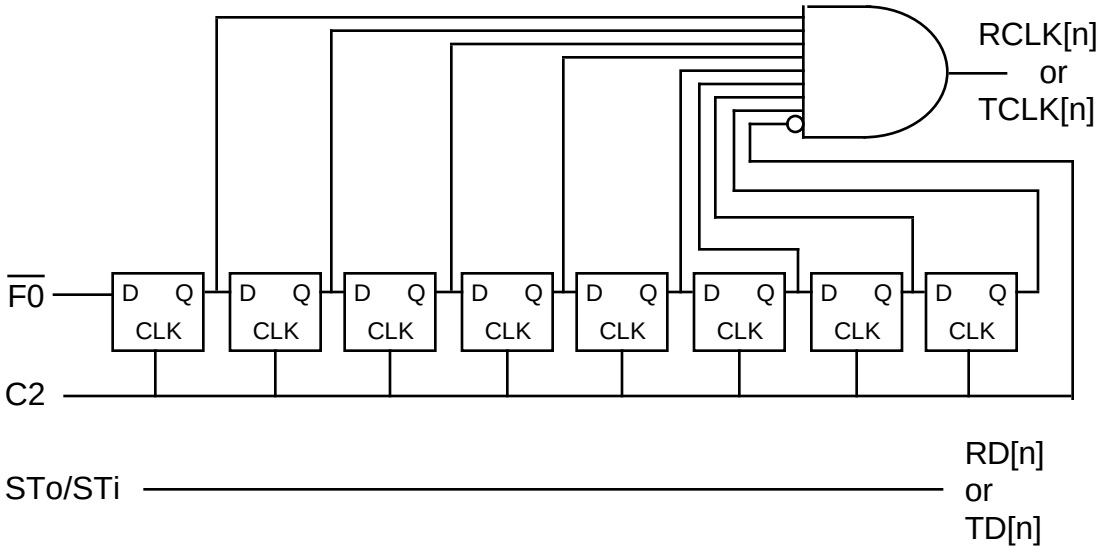
- The FREEDM receive data setup and hold times, and the transmit data propagation delay, meet the requirements for a ST-BUS component.
- The FREEDM is a TTL device, and an interface to ST-BUS components which are CMOS devices must be investigated. The timing may be affected by duty cycle distortion, and the ability of the CMOS device to drive a number of receive serial ports across the bus.
- The FREEDM is a 3.3V device, capable of interfacing to 5V devices. The interface timing may need to be investigated for the component which is interfaced to the FREEDM.
- The digital logic required to generate the gap in RCLK or TCLK will introduce timing delay. The delay must be investigated and compared with the worst case timing values, and it should be negligable.
- Each ST-BUS component will have different timing values, likely better than the worst case timing values shown here. The interface timing needs to be examined for the component which is interfaced to the FREEDM.

Recommended Logic at the Interface

The digital logic shown in figure 3 indicates how to interface the data stream of the ST-BUS to either a receive port or a transmit port of the FREEDM. A FREEDM device can be connected to as many as 64 ST-BUS's since there are 32 receive ports and 32 transmit ports. This drawing can be extended for the number of ports that are connected to a ST-BUS.

For systems in which C2 of all ST-BUS's are synchronized only one of the circuits of figure 3 is required. In this case the drawing does not need to be extended for all ports connected to a ST-BUS, and all FREEDM ports are clocked from the same signal.

Fig. 3 Digital Logic of FREEDM / ST-BUS Interface



REFERENCES

[1] PMC-931127, PMC-Sierra, "Frame Relay Protocol Engine and Datalink Manager" Standard Product Datasheet, July, 1996, Issue 5

[2] MSAN-126, Mitel Semiconductor, "ST-BUS Generic Device Specification (Rev. B)" Application Note

CONTACTING PMC-SIERRA

PMC-Sierra, Inc.
105 - 8555 Baxter Place
Burnaby, B.C.
Canada V5A 4V7

Telephone: 604-415-6000
Facsimile: 604-415-6200

Product Information: info@pmc-sierra.bc.ca
Applications information: apps@pmc-sierra.bc.ca

World Wide Web Site: http://www.pmc-sierra.com

NOTES

PMC-961218 Issue date: February, 1997

Printed in Canada