

DSI Bus Standard

Version 2.5

April 16, 2009

Implementation of the Bus Standard is governed by the terms of the Bus Standard Covenant. Changes to this specification must be agreed to by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC., and TRW AUTOMOTIVE INC.

DENSO

 **freescale™**
semiconductor
Launched by Motorola

TRW
Automotive

TABLE OF CONTENTS

1	INTRODUCTION.....	4
1.1	SCOPE	4
2	DSI OVERVIEW	4
2.1	GENERAL OVERVIEW.....	4
3	DSI NETWORK PHYSICAL LAYER.....	5
3.1	INTRODUCTION.....	5
3.1.1	<i>Network Configuration.....</i>	5
3.1.2	<i>Number of nodes</i>	5
3.2	DATA BIT ENCODING.....	6
3.2.1	<i>Voltage Mode Encoding.....</i>	6
3.2.2	<i>Current Mode Encoding.....</i>	6
3.3	BUS VOLTAGE LEVELS	7
3.4	BUS CURRENT LEVELS	7
3.5	ELECTRICAL CHARACTERISTICS	8
4	DSI NETWORK DATA LINK LAYER.....	10
4.1	MESSAGE FORMAT	10
4.2	WORD SIZES	10
4.3	BIT ORDER	11
4.4	CRC.....	11
4.5	COMMAND MESSAGE STRUCTURE.....	12
4.5.1	<i>Standard DSI Long Command Structure</i>	12
4.5.2	<i>Standard DSI Short Command Structure.....</i>	12
4.5.3	<i>Enhanced DSI Long Command Structure.....</i>	12
4.5.4	<i>Enhanced DSI Short Command Structure.....</i>	12
4.6	RESPONSE MESSAGE STRUCTURE.....	13
4.6.1	<i>Standard DSI Long Response Structure.....</i>	13
4.6.2	<i>Standard DSI Short Response Structure</i>	13
4.6.3	<i>Enhanced DSI Long Response Structure</i>	13
4.6.4	<i>Enhanced DSI Short Response Structure.....</i>	13
4.7	ERROR CHECKING	13
5	DSI ADDRESSING.....	14
5.1	INTRODUCTION.....	14

Implementation of the Bus Standard is governed by the terms of the Bus Standard Covenant. Changes to this specification must be agreed to by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC., and TRW AUTOMOTIVE INC.

5.2	SLAVE ADDRESSING	14
5.3	PROGRAMMABLE SLAVES	14
5.4	PRE-PROGRAMMED SLAVES.....	14
6	RESERVED DSI COMMANDS.....	15
6.1	MANDATORY DSI COMMANDS	15
6.1.1	<i>Initialization Command (0000)</i>	15
6.1.2	<i>Request Slave ID Information Command (0100)</i>	16
6.1.3	<i>Clear Command (0111)</i>	17
6.1.4	<i>Format Control (1010)</i>	18
6.2	OPTIONAL DSI COMMANDS	20
6.2.1	<i>Reverse Initialization Command (1111)</i>	20

1 INTRODUCTION

1.1 Scope

This document defines the Distributed System Interface (DSI). It documents the bus topology and electrical and physical characteristics. It also defines the message protocol and classes, formats, bit transmission order, and the method of programming devices with programmable addresses. Individual device messages are defined in the documentation for those devices.

2 DSI OVERVIEW

2.1 General Overview

The DSI is a niche area network (NAN) designed to interconnect multiple remote sensor and actuator devices to a central control module. The initial target application for the network is automotive airbag systems. Some of the characteristics for this application are the need for a low cost, highly robust, moderate speed interconnection limited to two wires. In addition it must failsafe, be deterministic, and have good EMC characteristics. Even though devices with all levels of intelligence and programmability may connect to the network, remote devices must be realizable with simple state machines. Since module size is very important a minimum of components in both the central module and remote units is critical.

Airbag systems have many types of components that may be connected to the network. Typically, these components are delivered from suppliers directly to the vehicle assembly plant. Some may be embedded in instrument panels and steering columns, others in seats, potentially others in the wiring harness. For these reasons it is highly desirable to allow network addressing to be self configuring at power-up. This minimizes the number of device types, and eliminates the need for special programming equipment at component suppliers and the vehicle assembly plant.

The above issues were paramount in the development of the DSI. To maintain determinism without sacrificing bus bandwidth and simplicity, a single master /multiple slave configuration is used. Robustness is maintained through the use of message cyclic redundancy check (CRC) codes and remote self diagnostics. High message density at moderate speeds and cost are facilitated by the simultaneous transmission of power, master commands, and slave responses. An optional daisy chain interconnection method is defined which allows the assignment of network addresses at power-up with a priori device information stored in the central module.

There are two variations on this Bus Standard, Standard and Enhanced as defined in Section 4. Standard and Enhanced devices can be mixed on one bus. The capability of the bus is defined as the minimum common capability of all devices on the bus.

3 DSI NETWORK PHYSICAL LAYER

3.1 Introduction

The DSI is a single master multiple slave data communications bus implemented on two wires. The bus utilizes voltage mode signaling for messages transmitted from the master to the slaves and current mode signaling from the slaves to the master. The master may transmit messages to one or a combination of slaves on the bus. Slaves only transmit in response to messages transmitted from the master. The number of nodes on the bus is variable but is known a priori for a particular configuration. One or more DSI busses may be used in a system.

3.1.1 Network Configuration

The network is configured as a two wire multi-drop bus. Slaves may attach to the bus in daisy chain or parallel connections. The optional daisy chain connection allows the central module to establish the node addresses at power-up. The parallel configuration may be used for devices that have preprogrammed or fixed addresses. The two may be combined on one bus. Figure 3-1 shows an example network configuration.

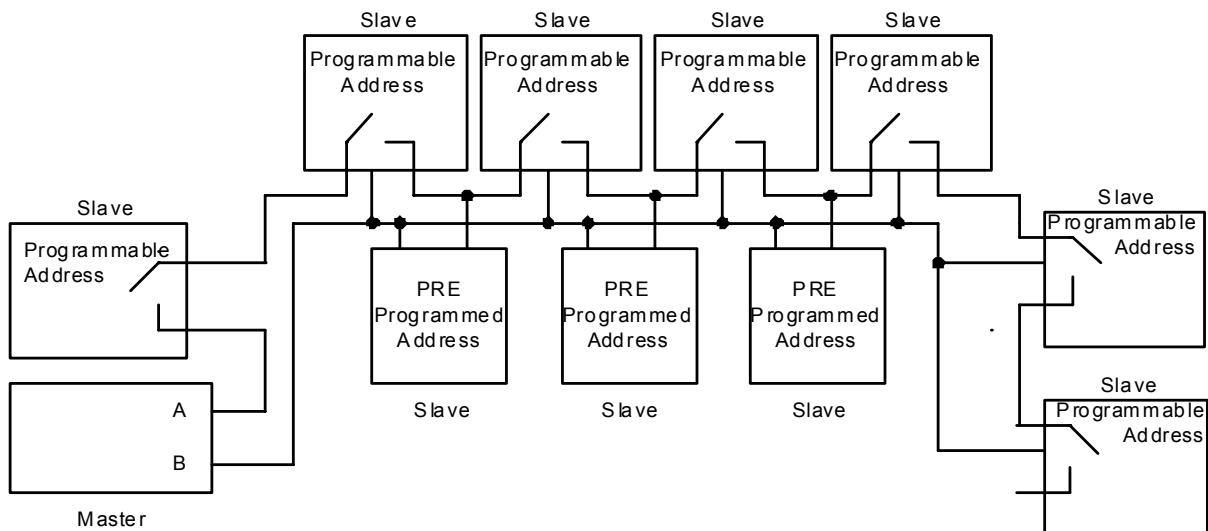


Figure 3-1 DSI Network Example

3.1.2 Number of nodes

The maximum number of nodes on a DSI bus is 16 (1 master and 15 slaves). The minimum number is 2 (1 master and 1 slave).

3.2 Data Bit Encoding

The DSI uses two methods of signaling: voltage mode for messages from the master to the slaves, and current mode for the responses from the slaves.

3.2.1 Voltage Mode Encoding

The voltage mode encoding uses a duty cycle modulated signal as shown in Figure 3-2.

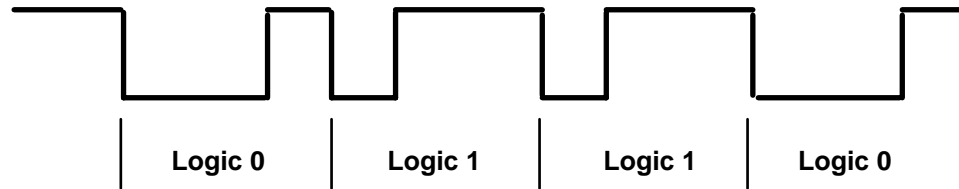


Figure 3-2 Voltage Mode Bit Encoding

Each bit time is broken up into thirds. For a logic zero the master produces a typical signal that is low for 2/3 of the bit time and high for the final 1/3. For a logic one the master produces a typical signal that is low for 1/3 of the bit time and high for the final 2/3.

3.2.2 Current Mode Encoding

Slave responses to commands are transmitted using a modulated current signal which is self synchronized to a falling edge voltage from the master. During the response time the master transmits a pulse train of any combination of ones or zeros (note: this signal can be the next command message if desired, or it can be a “null” message). The current mode bits are transmitted during the bit time. If a logic one is transmitted, the slave draws additional current from the source during the bit time. If a logic zero is transmitted the slave does not draw additional current during the bit time. Figure 3-3 shows a representation of the current waveform referenced to a voltage waveform.

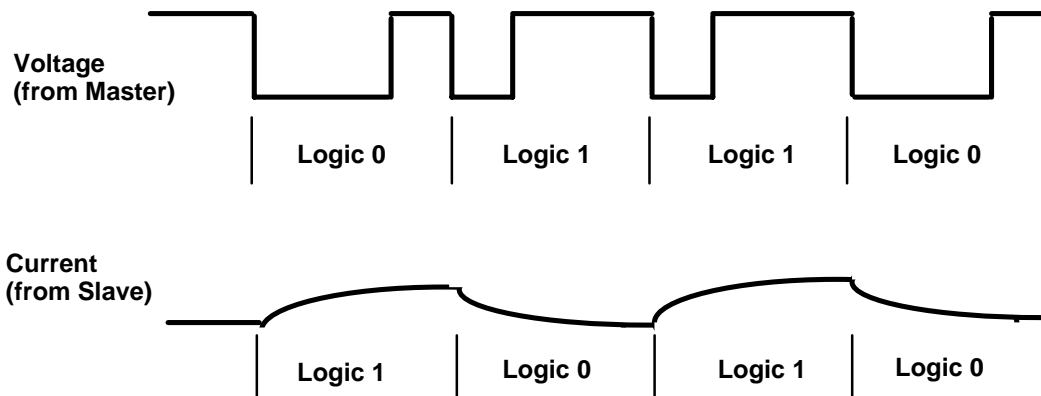


Figure 3-3 Current Waveform

3.3 Bus Voltage Levels

The voltage mode signaling uses a tri-level bus as shown in Figure 3-4. The master may supply a single ended or differential voltage. The slaves detect the absolute difference in the voltage between the two wires. Figure 3-4 shows the signal as seen at the slaves.

The bus idles at a voltage above the high threshold. At the start of frame, the bus transitions below the high threshold. This is followed by a transition below the low threshold to indicate the start of message. Data values are transmitted as specified in Section 3.2.1. The Low Threshold defines the difference between a Signal High and a Signal Low. The End of Message and the end of the last bit is signaled by the voltage rising above the high threshold. A “front porch” is added to the beginning of the word to allow the signal to transition from the idle voltage to the signal voltages without degrading the low time of the first bit. The Threshold and Signal values are specified in Section 3.5.

3.4 Bus Current Levels

There are two components to the bus current levels on the DSI related to signaling:

- The quiescent current draw of the slaves (I_{q_Total}). This is the sum of the quiescent currents of all of the slaves connected to the bus.
- The current drawn by the slaves during signaling (I_{SIG}).

The quiescent (I_{q_Total} , and I_q) and signal currents (I_{SIG}) are specified in Section 3.5.

3.5 Electrical Characteristics

#	Characteristic	Min	Typ	Max	Units
1	DSI Supply Current	I_{MIN}^1 125		I_{MAX}^2 400	mA
2	Total Bus Equivalent Load Capacitance (C_{MAX}) ³			20	nF
Master Transceiver					
3	Bus Idle Voltage ($I_{out} \leq I_{MIN}$) (V_{IDLE})	7		26.5	V
4	Signal High Voltage (V_{SIG_H})	4.175	4.5	4.825	V
5	Signal Low Voltage (V_{SIG_L})	1.325	1.5	1.675	V
6	Single Ended Drive	1.175	1.5	1.825	V
	Differential Drive				V
7	Signal Low Time for Logic Zero T_{0lo}	0.9(2/3 T_{bit})		1.1(2/3 T_{BIT})	μ s
8	Signal Low Time for Logic One T_{1lo}	0.9(1/3 T_{bit})		1.1(1/3 T_{BIT})	μ s
9	Voltage Slew Rate (V_{slew}) $C < 2000$ pF	0.10		8.00	Volts/ μ S
10	Received Logic Zero/One Trip Point (I_{TH})	5.00	6.00	7.00	mA
Slave Transceiver					
11	Logic One Signal Current (I_{SIG})	9.90	11	12.1	mA
12	Logic Zero Total Bus Quiescent Current (I_{q_Total})			1.6	mA
13	Logic Zero Bus Quiescent Current Per Slave (I_q)			0.1	mA
14	Current Slew Rate (I_{slew})	0.33		10.00	mA/ μ S
15	Bus High Threshold (V_{TH_H})	5.4	6.0	6.6	V
16	Bus Low Threshold (V_{TH_L})	2.7	3.0	3.3	V
Message Characteristics					
17	Signal Bit Time (T_{BIT}) ⁴	5.0		200	μ s
18	Master Front Porch ($V_{TH_H} \leq V_{BUS} \leq V_{IDLE}$) (t_{FP_Master}) Ref Figure 3-4			2	bit time
19	Slave Front Porch ($V_{TH_L} \leq V_{BUS} \leq V_{TH_H}$) (t_{FP_Slave}) Ref Figure 3-4	1/3		1	bit time
20	Master Inter-Frame Separation (t_{IFS_Master}) Ref Figure 3-4	4			T_{BIT}
21	Slave Inter-Frame Separation (t_{IFS_Slave}) Ref Figure 3-4	4			T_{BIT}
Daisy Chain Characteristics					
22	Daisy Chain Connect Resistance (Per Slave) ($R_{DC} \leq R_{DC_CLOSE}$)			16	Ω
23	Daisy Chain Disconnect Resistance ($R_{DC} \geq R_{DC_OPEN}$)	1			M Ω
24	Daisy Chain Bus Connect Time ($t_{BSCLOSE}$) ⁵			500	μ s
25	Daisy Chain Bus Disconnect Time (t_{BSOPEN}) ⁶			1000	μ s

Notes:

- I_{MIN} is the minimum current which the master device must be capable of continuously supplying
- The maximum DSI supply current is governed by the characteristics of the devices on the bus and other system bus conditions.
- Equivalent capacitance for single ended drive is total capacitance from bus to return or circuit ground. Equivalent capacitance for differential drive is total capacitance from either drive line to circuit ground plus twice the capacitance between the bus drive lines.
- The minimum bit time is bound by the voltage (#9) and current (#14) slew rate specifications, the characteristics of the devices on the bus and other system bus conditions.
- $t_{BSCLOSE}$ is the time from an Initialization Command or Reverse Initialization command received until the R_{DC} is within specification
- t_{BSOPEN} is the time from slave reset until R_{DC} is within specification

Implementation of the Bus Standard is governed by the terms of the Bus Standard Covenant. Changes to this specification must be agreed to by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC., and TRW AUTOMOTIVE INC.

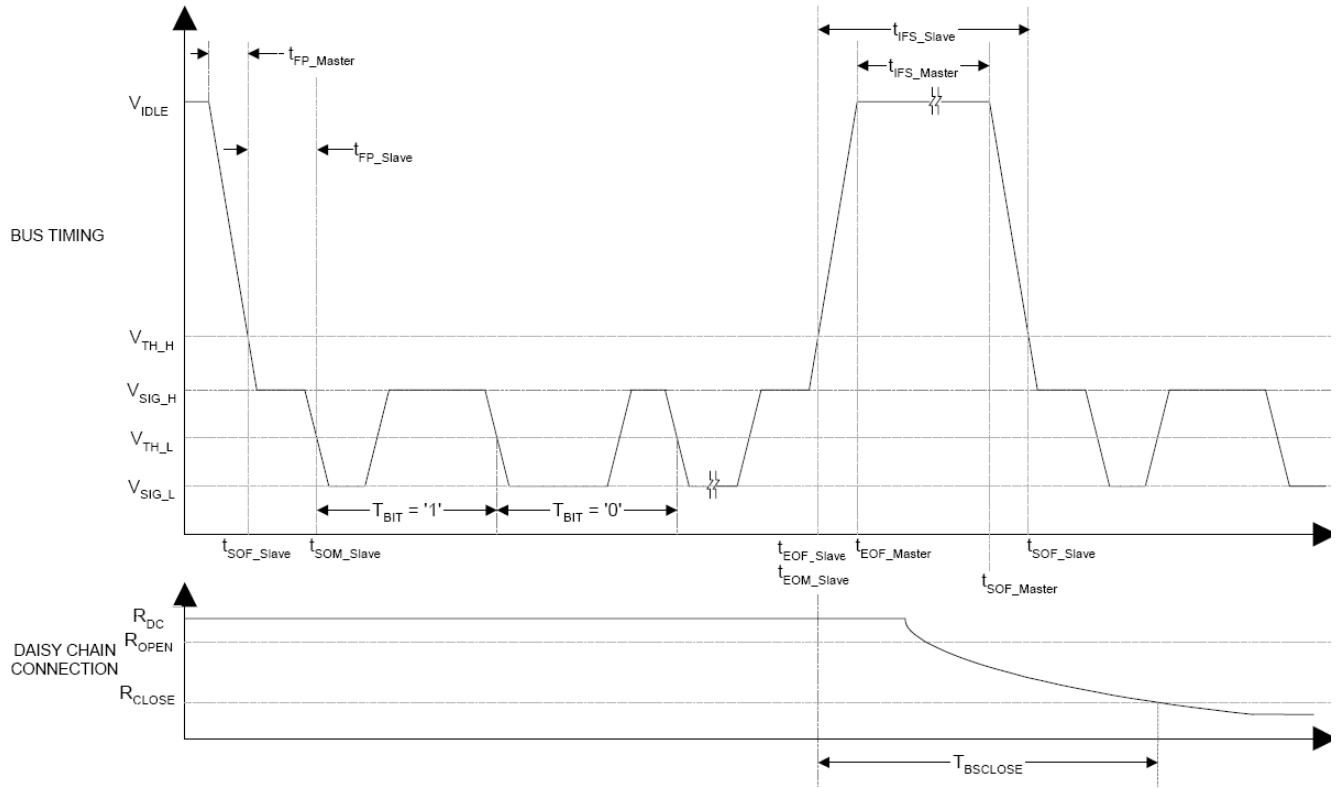


Figure 3-4: DSI Signal Timing Diagram

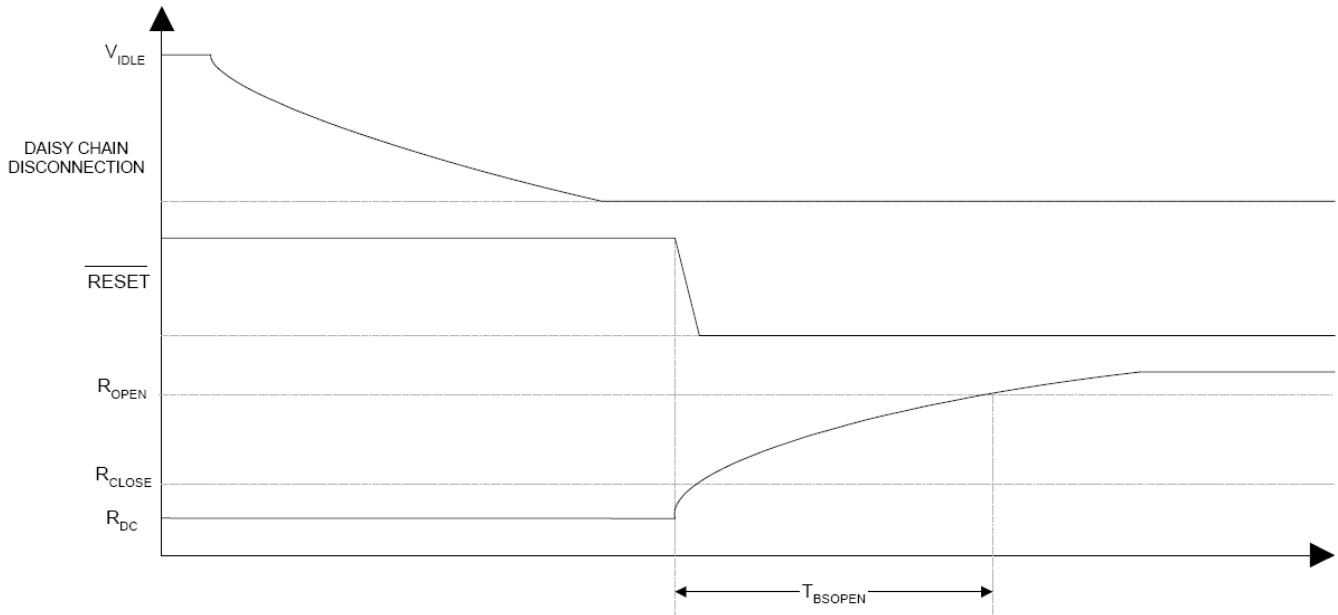


Figure 3-5: DSI Daisy Chain Disconnect Timing Diagram

Implementation of the Bus Standard is governed by the terms of the Bus Standard Covenant. Changes to this specification must be agreed to by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC., and TRW AUTOMOTIVE INC.

4 DSI NETWORK DATA LINK LAYER

4.1 Message Format

DSI messages are composed of individual words separated by the Inter-Frame Separation (t_{IFS_Master}). Transfers are full duplex. Command messages from the master occur at the same time as responses from the slaves. Slave responses to commands occur during the next command message. This allows slaves time to decode the command, retrieve the information and prepare to transmit it to the master. A bus traffic example is shown in Figure 4-1.

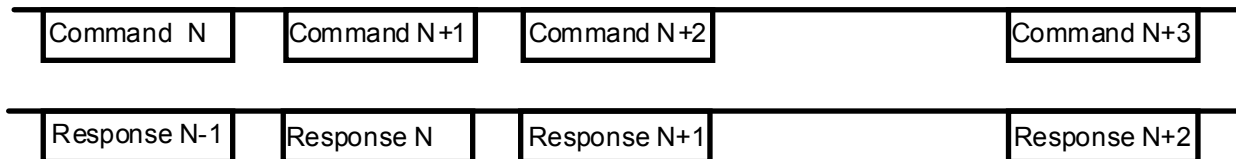


Figure 4-1 Bus Traffic Example

The example shows three commands separated by the Inter-Frame Separation (t_{IFS_Master}), followed by a command after a longer delay. The minimum t_{IFS_Master} , as specified in Section 3.5, is required to allow recharge of the energy storage devices in the slaves. This is necessary because the slave receives its power from the signal line. The minimum IFS required is dependent upon several factors including the bus speed, the current consumption of the slaves and the amount of energy storage in the network.

4.2 Word Sizes

There are two word sizes, long and short. A standard DSI long word consists of 16-bits of information followed by a 4-bit cyclic redundancy check (CRC). A standard DSI short word is composed of 8-bits of information followed by the 4-bit CRC. An enhanced DSI long word consists of 16 bits of information followed by a 0 to 8-bit CRC. An enhanced DSI short word consists of 8 to 15-bits of information followed by a 0 to 8-bit CRC.

For the target applications it is expected that most master/slave communications can be completed within one of these word sizes. However, longer messages can be composed of multiple words with an appropriately defined bit encoding.

At startup and after a “Clear” command has been transmitted, as defined in Section 6.1.3, all slaves must support standard DSI long words. Once standard slaves are initialized, they may additionally support standard DSI short and long words. Once Enhanced mode slaves are initialized, they may additionally support standard DSI short and long words as well as Enhanced DSI short and long words. Support of enhanced mode commands is controlled by using the Format Control Command as specified in Section 6.1.4.

4.3 Bit Order

All commands and responses are transmitted with the message information bits first and the CRC last. The byte and bit transmission for the Standard DSI order is shown in Figures 4-2 and 4-3. The byte and bit transmission for the Enhanced DSI order is shown in Figure 4-4 and Figure 4-5.

Enhanced DSI devices can have the length of the CRC changed in a long word. Enhanced DSI devices can have the length of the Data and the length of the CRC changed in a short word. This is accomplished using the Format Command as specified in Section 6.1.4.

First																Last			
BYTE1								BYTE2								CRC			
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	X3	X2	X1	X0

Figure 4-2 Standard DSI Long Command Bit Order

First												Last			
BYTE								CRC							
D7	D6	D5	D4	D3	D2	D1	D0	X3	X2	X1	X0				

Figure 4-3 Standard DSI Short Command Bit Order

First																Last			
BYTE1								BYTE2								CRC			
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0 to 8 Bits of CRC			

Figure 4-4 Enhanced DSI Long Command Bit Order

First												Last			
BYTE								CRC							
8 to 15 Bits of Data								0 to 8 Bits of CRC							

Figure 4-5 Enhanced DSI Short Command Bit Order

4.4 CRC

At startup and after a “Clear” command has been transmitted, as defined in Section 6.1.3, all slaves must support a 4-bit CRC using a polynomial of X^4+1 , and a seed value of ‘1010’.

The length, polynomial and seed can be changed in Enhanced DSI devices using the Format Command as specified in Section 6.1.4.

4.5 Command Message Structure

Command messages are transmitted from the master to the slave(s). Messages from the master use either the Standard or Enhanced, long or short word DSI structure as described below.

4.5.1 Standard DSI Long Command Structure

The Standard DSI long command structure is shown in Figure 4-6.

DATA								ADDRESS				COMMAND				CRC			
D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0	X3	X2	X1	X0

Figure 4-6 Standard DSI Long Command

The Standard DSI long commands consists of 8 bits of data, the encoded 4-bit address of the intended slave, a 4-bit encoded command, and the calculated 4-bit CRC.

4.5.2 Standard DSI Short Command Structure

The Standard DSI short word command structure is shown in Figure 4-7.

ADDRESS				COMMAND				CRC			
A3	A2	A1	A0	C3	C2	C1	C0	X3	X2	X1	X0

Figure 4-7 Standard DSI Short Command

The Standard DSI short command consists of the encoded 4-bit address of the intended slave, a 4-bit encoded command, and the calculated 4-bit CRC.

4.5.3 Enhanced DSI Long Command Structure

The Enhanced DSI Long command contains the same number of non-CRC bits as the Standard DSI Long command. They are in the same order as the Standard DSI Long command. The length of the CRC can range from 0 to 8 bits and is at the end of the message. Figure 4-8 shows the Enhanced DSI Long Command structure.

DATA								ADDRESS				COMMAND				CRC
D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0	0 to 8 bits (MSB first)

Figure 4-8 Enhanced DSI Long Command

4.5.4 Enhanced DSI Short Command Structure

Enhanced DSI devices can change to a different number of data and CRC bits in the short command. The total length including CRC (if any) cannot be less than 8 bits. The non-CRC length must be at least 8 bits. If the non-CRC length is more than 8 bits, the bits in excess of 8 are not defined by this specification. The number and use of these bits is defined in individual device specifications. Figure 4-9 shows the Enhanced DSI Short Command structure.

Unassigned	ADDRESS				COMMAND				CRC
0 to 7 bits	A3	A2	A1	A0	D3	D2	D1	D0	0 to 8 bits (MSB first)

Figure 4-9 Enhanced DSI Short Command

4.6 Response Message Structure

Response messages are transmitted from the slaves to the master. Responses are always transmitted during the subsequent command and with the same length as the previously received command. When the word format changes between successive commands the first response transmitted during the new format will be invalid since it will not have the proper number of bits.

4.6.1 Standard DSI Long Response Structure

A Standard DSI long response is transmitted from the slave to the master in response to a Standard DSI long command. The standard DSI long response consists of two 8-bit data bytes and the calculated 4-bit CRC. The response structure is shown in Figure 4-10.

DATA BYTE1								DATA BYTE2								CRC			
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	X3	X2	X1	X0

Figure 4-10 Standard DSI Long Response Structure

4.6.2 Standard DSI Short Response Structure

A Standard DSI short response is transmitted from the slave to the master in response to a Standard DSI short command. The Standard DSI short response consists of one 8-bit data byte and the calculated 4-bit CRC. The response structure is shown in Figure 4-11.

DATA BYTE2								CRC			
D7	D6	D5	D4	D3	D2	D1	D0	X3	X2	X1	X0

Figure 4-11 Standard DSI Short Response Structure

4.6.3 Enhanced DSI Long Response Structure

An Enhanced DSI long response is transmitted from the slave to the master in response to an Enhanced DSI long command. The Enhanced DSI long response must contain the same number of CRC and non-CRC bits as the Enhanced DSI long command received. These values can be changed using the Format Command as specified in Section 6.1.4. The response structure is shown in Figure 4-12.

DATA																CRC			
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0 to 8 bits (Same as command CRC bits)			

Figure 4-12 Enhanced DSI Long Response Structure

4.6.4 Enhanced DSI Short Response Structure

An Enhanced DSI short response is transmitted from the slave to the master in response to an Enhanced DSI short command. The Enhanced DSI short response must contain the same number of CRC and non-CRC bits as the Enhanced DSI short command received. The number of non-CRC bits must be at least 8. These values can be changed using the Format Command as specified in Section 6.1.4. The response structure is shown in Figure 4-13.

Data	CRC
8 to 15 bits (Same as non-CRC command bits)	0 to 8 bits (Same as command CRC bits)

Figure 4-13 Enhanced DSI Short Response Structure

4.7 Error Checking

The master and slaves calculate a CRC on the information portion of their received messages. The message is valid only if the calculated CRC matches the CRC transmitted as part of the message. An error bit is set in the master when it receives an invalid message. The slaves discard and ignore all invalid received messages and in addition do not respond to them.

5 DSI Addressing

5.1 Introduction

This section establishes a method for device addressing and programming slave addresses in the system.

5.2 Slave Addressing

Each slave on the bus must be given a unique 4-bit address. The address may be pre-programmed into the device or it may be programmed using the technique described in section 5.3. When address 0 is used, it is called a global command and all devices are addressed at once. Generally, devices do not respond to global commands. The device address encoding is shown in Figure 5-1.

A3	A2	A1	A0	Slave Number
0	0	0	0	All Slaves
0	0	0	1	Slave 1
0	0	1	0	Slave 2
0	0	1	1	Slave 3
0	1	0	0	Slave 4
0	1	0	1	Slave 5
0	1	1	0	Slave 6
0	1	1	1	Slave 7
1	0	0	0	Slave 8
1	0	0	1	Slave 9
1	0	1	0	Slave 10
1	0	1	1	Slave 11
1	1	0	0	Slave 12
1	1	0	1	Slave 13
1	1	1	0	Slave 14
1	1	1	1	Slave 15

Figure 5-1 Slave Address Encoding

5.3 Programmable Slaves

In the optional daisy chain configuration, after system power up, the master must set the address of programmable slaves before network communications can commence. Programmable slaves have a method to connect or disconnect additional slaves to the master. At power up, the programmable slaves must disconnect the master from the next slave on the bus. The programmable slave must only connect the next slave to the master when an initialization or reverse-initialization message is received.

On power up, the first slave is the only slave connected to the master. When the master transmits a slave initialization command the first slave stores its address information and connects the master to the next slave on the bus. Now the second daisy chain slave is connected to the network and can be initialized. The first slave responds with an initialization response message during the subsequent command, which may be the initialization command for the second slave. This sequence continues until all slaves are initialized.

5.4 Pre-programmed Slaves

Slaves that are pre-programmed do not require a method to connect or disconnect additional slaves to the master. If included, it must be disconnected following power-up. On power-up the stored pre-programmed address becomes the slave address. However, pre-programmed devices still must receive an Initialization or reverse-initialization command and reply with an initialization or reverse-initialization response before responding to any other bus commands. The initialization or reverse-initialization command will control the connection to additional slaves on the bus, similar to a programmable slave.

6 Reserved DSI Commands

6.1 Mandatory DSI Commands

All DSI devices are required to decode and, if applicable, respond to the commands listed in this section.

6.1.1 Initialization Command (0000)

Initialization commands are used to activate slaves on the bus. The Initialization command must be transmitted as a Standard DSI long command. Figure 6-1 shows the structure of an Initialization command.

D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0
DD	BS	DD	DD	PA3	PA2	PA1	PA0	A3	A2	A1	A0	0	0	0	0

Figure 6-1 Initialization Command Structure

Where:

A[3:0] = The address of the slave to be Initialized. The Initialization command can use the global address for programmable slaves and specific addresses for pre-programmed slaves. For un-programmed slaves, the Initialization command sets the address of the slave to PA[3:0] for future commands.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

BS = "1" is a request to connect the bus to additional slaves.

"0" is a request to disconnect the bus from additional slaves.

PA[3:0] = Address to set this slave to if un-programmed. Address of programmed device if pre-programmed.

In the case where a ring topology is used, the Initialization command must be applied to the input terminal of the slave. Initialization commands applied to the output terminal of a slave in ring topology must be ignored.

Figure 6-2 shows the structure of the response to an Initialization command.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
A3	A2	A1	A0	DD	DD	DD	DD	DD	DD	DD	DD	PA3	PA2	PA1	PA0

Figure 6-2 Initialization Response Structure

Where:

PA[3:0] = An echo of the address to set this slave to if un-programmed. The address of the programmed device if pre-programmed.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

A[3:0] = The address of the slave.

6.1.2 Request Slave ID Information Command (0100)

Request Slave ID Information commands are used to identify the slave type and revision information. The master can use this information to determine the internal memory map of the slave. The Request Slave ID Information command can be transmitted in any command structure defined in Section 4.5. Figure 6-3 shows the structure of a Request Slave ID Information command.

D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0
DD	DD	DD	DD	DD	DD	DD	DD	A3	A2	A1	A0	0	1	0	0

Figure 6-3 Request Slave ID Command Structure

Where:

A[3:0] = The address of the slave.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

Figure 6-4 shows the structure of the response to a Request Slave ID Information command.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
A3	A2	A1	A0	DD	DD	DD	DD	V2	V1	V0	ID4	ID3	ID2	ID1	ID0

Figure 6-4 Request Slave ID Information Response Structure

Where:

ID[4:0] = The slave Identification code. Reference Figure 6-5.

V[2:0] = The revision # of the slave.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

A[3:0] = The address of the slave.

Slave Identification codes can be used by the master to determine the correct slave type, and to infer internal memory map information of the slave. Figure 6-5 identifies the pre-defined slave identification codes. Allocation of new Device ID definitions must be approved by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC. and TRW AUTOMOTIVE INC.

ID4	ID3	ID2	ID1	ID0	Slave Type
0	0	0	0	0	DSI2.02 Analog Slave Interface Device
0	0	0	0	1	DSI2.02 Analog Slave Interface Device
0	0	0	1	0	Available for New Devices
0	0	0	1	1	DSI2.02 Analog Slave Interface Device
0	0	1	0	0	Integrated DSI2.02 Inertial Sensor Slave
0	0	1	0	1	Available for New Devices
0	0	1	1	0	Integrated DSI2.02 Pressure Sensor Slave
0	0	1	1	1	Available for New Devices
0	1	0	0	0	Available for New Devices
0	1	0	0	1	Available for New Devices
0	1	0	1	0	Available for New Devices
0	1	0	1	1	Available for New Devices
0	1	1	0	0	Integrated DSI2.02 Inertial Sensor Slave
0	1	1	0	1	Available for New Devices
0	1	1	1	0	Available for New Devices
0	1	1	1	1	Available for New Devices
1	0	0	0	0	DSI2.02 Analog Slave Interface Device
1	0	0	0	1	Reserved
1	0	0	1	0	Reserved
1	0	0	1	1	Reserved
1	0	1	0	0	Reserved
1	0	1	0	1	Reserved
1	0	1	1	0	Reserved
1	0	1	1	1	Reserved
1	1	0	0	0	Reserved
1	1	0	0	1	Reserved
1	1	0	1	0	Reserved
1	1	0	1	1	Reserved
1	1	1	0	0	Reserved
1	1	1	0	1	Reserved
1	1	1	1	0	Reserved
1	1	1	1	1	Reserved

Figure 6-5 Pre-Defined Slave Identification Codes

6.1.3 Clear Command (0111)

The clear command is used to reset the slaves and place them in the pre-Initialization state. After receiving a Clear command, the slave must disable the connection to additional slaves on the bus within t_{BSOPEN} , specified in Section 3.5. The Clear command can be transmitted in any command structure defined in Section 4.5. Figure 6-6 shows the structure of a Clear command. There is no response to a Clear command.

D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0
DD	DD	DD	DD	DD	DD	DD	DD	A3	A2	A1	A0	0	1	1	1

Figure 6-6 Clear Command Structure

Where:

A[3:0] = The address of the slave to be reset. The Clear command can use a specific slave address to reset one slave, or the global address to clear all of the slaves on the bus with a single command.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

Implementation of the Bus Standard is governed by the terms of the Bus Standard Covenant. Changes to this specification must be agreed to by DENSO CORPORATION, FREESCALE SEMICONDUCTOR, INC., and TRW AUTOMOTIVE INC.

6.1.4 Format Control (1010)

Format Control commands are used to change the properties of the DSI commands and responses for future transmissions. The Format Control command can be transmitted as either a Standard long command, or an Enhanced long command as defined in Section 4.5. The Format Control command is used to read or modify the contents of the Format Control registers in Figure 6-9. Figure 6-7 shows the structure of a Format Control command.

D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0
R/W	FA2	FA1	FA0	FD3	FD2	FD1	FD0	A3	A2	A1	A0	1	0	1	0

Figure 6-7 Format Control Command Structure

Where:

- A[3:0] = The address of the slave.
- FD[3:0] = The data to be written to the Format Register addressed by FA[2:0] if R/W is '1'. Unused if R/W is '0'.
- FA[2:0] = The address of the Format Register to be read or written.
- R/W = '1' is a request to write to the data contained in FD[3:0] to the Format Register addressed by FA[2:0].
 '0' is a request to read the data contained the Format Register addressed by FA[2:0] and provide the data in the DSI response.

Figure 6-8 shows the structure of the response to a Format Control command.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
A3	A2	A1	A0	DD	DD	DD	DD	R/W	FA2	FA1	FA0	FD3	FD2	FD1	FD0

Figure 6-8 Format Control Response Structure

Where:

- FD[3:0] = The data which was written to, or read from the Format Register addressed by FA[2:0].
- FA[2:0] = The address of the Format Register to be read or written.
- R/W = Echo of the R/W bit from the Format Control Command.
- DD = Device Dependent. Value and function are not restricted or controlled by this standard.
- A[3:0] = The address of the slave.

Figure 6-9 shows the register map and default values for the Format Control registers. The registers associated with Format Control will default to values which correspond to Standard DSI operation upon power-up or after receiving a “Clear” command.

Format Register Address				Description	Default / Standard DSI Values				
FA2	FA1	FA0	Decimal Address		FD3	FD2	FD1	FD0	Definition
0	0	0	0	CRC Polynomial – Low Nibble	0	0	0	1	CRC Polynomial = $X^4 + 1$
0	0	1	1	CRC Polynomial – High Nibble	0	0	0	1	
0	1	0	2	Seed – Low Nibble	1	0	1	0	Seed = ‘1010’
0	1	1	3	Seed – High Nibble	0	0	0	0	
1	0	0	4	CRC Length (0 to 8)	0	1	0	0	CRC Length = 4
1	0	1	5	Short Word Data Length	1	0	0	0	Short Word Data Length = 8
1	1	0	6	Reserved	0	0	0	0	N/A
1	1	1	7	Format Selection	0	0	0	0	N/A

Figure 6-9 Format Control Registers

6.1.4.1 CRC Polynomial

The CRC Taps control the feedback for the CRC Polynomial. The Most Significant Bit represents the X^7 coefficient. The Least Significant Bit represents X^0 coefficient. The standard DSI CRC of X^4+1 is obtained by loading ‘0001’ into Format register 0 and ‘0001’ into Format Control register 1. On a reset or clear, the standard DSI CRC taps are loaded into these registers.

6.1.4.2 Seed

The Seed is the starting value loaded into the CRC checking registers before each transaction starts. The default DSI seed of ‘1010’ is selected by loading ‘1010’ into control register 2 and ‘0000’ into control register 3. On reset or clear, the standard DSI seed is loaded into these registers.

6.1.4.3 CRC Length

The CRC length value can range from 0 bits (no CRC) to 8 bits. On a reset or clear, the value in this register defaults to 4. If a value greater than 8 is written to the register, it is ignored and the contents of the register are not changed. The standard DSI CRC length is set by loading ‘0100’ into this register.

6.1.4.4 Short Word Data Length

The Short Word Data Length can range from 8 bits to 15 bits. On a reset or clear, the value in this register defaults to 8. If a value less than 8 is written to the register, it is ignored and the contents of the register are not changed.

6.1.4.5 Format Selection

The contents of the Format Selection register determine whether the standard DSI values or the values in the format control registers are used. If the Format Selection register contains ‘1111’, the Format Control register values are active. Any write to the Format Control registers will become active upon completion of the write. A write of ‘0000’ to the Format Selection register activates the standard DSI values. A write to the Format Selection register of any other value is ignored.

6.2 Optional DSI Commands

The commands listed in this section are pre-defined and must only be used for the function described.

6.2.1 Reverse Initialization Command (1111)

Reverse Initialization commands are used to activate slaves on the bus in the reverse direction of a ring topology. The Reverse Initialization command must be transmitted as a Standard DSI long command. Figure 6-10 shows the structure of an Initialization command.

D7	D6	D5	D4	D3	D2	D1	D0	A3	A2	A1	A0	C3	C2	C1	C0
DD	BS	DD	DD	PA3	PA2	PA1	PA0	A3	A2	A1	A0	1	1	1	1

Figure 6-10 Reverse Initialization Command Structure

Where:

A[3:0] = The address of the slave to be Initialized. The Reverse Initialization command can use the global address for programmable slaves and specific addresses for pre-programmed slaves. For un-programmed slaves, the Reverse Initialization command sets the address of the slave to PA[3:0] for future commands.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

BS = "1" is a request to connect the bus to additional slaves.

"0" is a request to disconnect the bus from additional slaves.

PA[3:0] = Address to set this device to if un-programmed. Address of programmed device if pre-programmed.

The Reverse Initialization command is intended only for slaves in a ring topology and must always be applied to the output terminal of the slave. Reverse Initialization commands applied to the input terminal of the slave must be ignored.

Figure 6-11 shows the structure of the response to a Reverse Initialization command.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
A3	A2	A1	A0	DD	DD	DD	DD	DD	DD	DD	DD	DD	PA3	PA2	PA1	PA0

Figure 6-11 Reverse Initialization Response Structure

Where:

PA[3:0] = An echo of the address to set this slave to if un-programmed. The address of the programmed device if pre-programmed.

DD = Device Dependent. Value and function are not restricted or controlled by this standard.

A[3:0] = The address of the slave.