TEXAS INSTRUMENTS

Reader Series 4000

S4100 Multi-Function Reader Module RF-MGR-MNMN

ISO 15693 Library Reference Guide

11-06-21-707 Rev 1 July 2004

Second Edition - July 2004

This is the second edition of this manual. It describes the **TI Series 4000 Reader.**

It contains a description of the following reader module:

S4100 Multi-Function Reader Module P/N: **RF-MGR-MNMN-N3**

Texas Instruments (TI) reserves the right to make changes to its products or services or to discontinue any product or service at any time without notice. TI provides customer assistance in various technical areas, but does not have full access to data concerning the use and applications of customer's products

Therefore, TI assumes no liability and is not responsible for customer applications or product or software design or performance relating to systems or applications incorporating TI products. In addition, TI assumes no liability and is not responsible for infringement of patents and/or any other intellectual or industrial property rights of third parties, which may result from assistance provided by TI.

TI products are not designed, intended, authorized or warranted to be suitable for life support applications or any other life critical applications which could involve potential risk of death, personal injury or severe property or environmental damage.

Read This First

About This Manual

This reference guide for the Series 4000 Multi-Function (13.56 MHz & 134.2 KHz) Reader is designed for use by TI customers who are engineers experienced with RFID Systems and Radio Frequency Identification Devices (RFID).

Device Name
RF-MGR-MNMN-N3

The Regulatory, safety and warranty notices that must be followed are provided in Chapter 2.

Conventions

The following pictograms and designations are used in the operating instructions:

WARNING:

A WARNING IS USED WHERE CARE MUST BE TAKEN, OR A CERTAIN PROCEDURE MUST BE FOLLOWED, IN ORDER TO PREVENT INJURY OR HARM TO YOUR HEALTH.

CAUTION:

This indicates information on conditions, which must be met, or a procedure, which must be followed, which if not needed could cause permanent damage to the system.

Note:

Indicates conditions, which must be met, or procedures which must be followed, to ensure proper functioning.

Information:

Indicates conditions or procedures that should be followed to ensure proper functioning of the system.

If You Need Assistance

Application Centers are located in Europe, North and South America, the Far East and Australia to provide direct engineering support.

For more information, please contact your nearest TIRIS Sales and Application Center. The contact addresses can be found on our home page: http://www.tirfid.com.

Numerical Representations

Unless otherwise noted, numbers are represented as decimal.

Hexadecimal numbers are represented with the suffix $_{16}$, e.g. $A5F1_{16}$

Binary numbers are represented with the suffix $_2$, e.g. 1011_2

Byte representations: the least significant bit (lsb) is bit 0 and the most significant bit (msb) is bit 7.

Document Overview

Chapter 1

ISO 15693 Library

Topic Page

1.1 ISO 15693 Overview

The following sections define and detail the Protocol functionality in the 15693 Module of the MFR Base Application. This information includes 15693 Protocol Commands and the data/parameters associated with them.

Find Token Request (41_{16})

The host application can send the MFR Base Application a Request Packet to check if a token is present. This packet contains a loop count that sets the maximum number of times the MFR Module Base Application will search for the token. This function allows a great deal of flexibility. It is possible to search for a variety of transponders or a single type of token. If multiple transponders are selected, the function returns the first token that is read. The Application continues to loop as specified unless it detects a token. The Application doesn't return every type of token - just the first one it reads.

After a successful read the MFR Module Base Application responds with **ERROR_NONE** in <*Status*> followed by the token's RF Technology Type and any token data. The RF polls stop once a valid token is found. If a valid token is not found within the number of loops selected, the MFR Module Base Application responds with an **ERROR_TOKEN_NOT_PRESENT** <*Status*> field. The RF Poll stops once the loops are complete.

The ISO15693 token search is handled by the function Find_Token_15693(). Note that this function is not called when ISO15693 is not in the Priority table, unless the request is directed specifically to the ISO15693 library. It will not be called if a different token is discovered prior to reaching the ISO15693 format in the priority table.

Information:
The Find Token request is implemented internally as multiple "Inventory" and "Slot Marker" commands to arbitrate collisions.

The Application Entity limits the number of ISO15693 tokens reported to 1, but if multiple transponders are detected, the entity responds with ERROR_COLLISION_DETECT.

The ISO15693 Library Entity limits the number of ISO15693 tokens reported to 16 and concatenates their responses; if more transponders are detected, the entity responds with ERROR_COLLISION_DETECT.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	01 / 04	Entity ID
Cmd2	1	41	Find Token
LoopCount	1	00 – FF	Number of attempts to find token (0 ⇒ Loop until next request)

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	01 / 04	Entity ID
Cmd2	1	41	Find Token
Status	1	00 – FF	Standard Error Codes
EntityId	1	04	ISO15693 Library
ToknDat 1	0 / 10	00 – FF	Byte 1: Inventory Response Flags
		00 – FF	2: DSFID
		00 – FF	3-10: Unique ID
		≡	
ToknDat n[1]	0 / 10	00 – FF	Byte 1: Inventory Response Flags
		00 – FF	2: DSFID
		00 – FF	3-10: Unique ID

[1] $n \leq 16$

Example: The request packet specifies 10 loops.

Request Packet: (**01 09 00 03 04 41 0A 44 BB**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	41	Find Token Request
LoopCount	0A	10 loops maximum
BCC	44 BB	LRC and ~LRC

Response Packet: (**01 14 00 03 04 41 00 04 00 00 94 AE 81 06 00 00 07 E0 0D F2**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	14 00	Packet Length 20 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	41	Find Token Request
Status	00	ERROR_NONE
EntityID	04	ISO15693 Entity ID
ToknDat 1	00 00 94 AE 81 06 00 00 07 E0	Inventory Response Flags DSFID (Data Storage Format ID) UID (LSB first)
BCC	0D F2	LRC and ~LRC

Pass-Through Request (45_{16})

The Pass-Though Request Packet provides a way for a host to have direct communications to an Entity Module library. The Pass-Through Request packet specifies the Entity to direct the data to and then provides data that the library can send directly to its token. This function provides a way to access the low level RF protocol directly, bypassing any abstraction provided by the Library and Application Layer.

The ISO15693 library module/entity allows direct access to the RF interface via the pass-through request. When the ISO15693 module receives a pass-through request, the module configures the HF ASIC to the ISO15693 specific RF scheme, according to the "Set Parameters" request. The data in the pass-through request is not interpreted and it has meaning only to the token(s) that may receive it. Therefore the process that initiated the request must know the structure and data content of the transponder protocol. These transponder protocol details are specified in the ISO15693 layer 3 standard. Application and security layer commands may also be sent to the token via the pass-through request as long as the calling process understands and follows the states of the token at hand.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	45	Pass Through
Data	0 – n(1)	00 – FF	Data to send via RF channel

(1) Length 'n' depends on RF message

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	45	Pass Through
Status	1	00 – FF	Standard Error Codes
Data	0 – n(1)	00 – FF	ISO15693 Reply data (if received)

(1) Length 'n' depends on RF message

Pass Through Stay Quiet Example:

Send a valid ISO15693 Pass-through "Stay Quiet" Request Packet to the Mfr module application. Have a valid ISO15693 compliant token in the field when the request is issued. This example shows a token with Unique ID code **E3 8F 47 01 00 00 07 E0**. Therefore, the UID (retrieved from the Find Token Request), CRC and BCC bytes should be changed to match the token currently being interrogated.

There is no RF response to this packet, so the <*Status*> field contains a value of **ERROR_TOKEN_NOT_PRESENT**. This response is in keeping with the purpose of the "Stay Quiet" request, which is to place the token in a state where it will not respond to "Inventory" commands (as used in the Find Token Request) or non-addressed commands until reset to Ready state, or until the Token loses power (i.e. transmitter deactivated or Token removed from read zone and returned). To test this request, the Transmitter On Request should be issued prior to "Stay Quiet".

Request Packet: (**01 14 00 03 04 45 23 02 E3 8F 47 01 00 00 07 E0 C6 92 EF 10**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	14 00	Packet Length 20 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	45	Pass-through Request
Data	23 02 E3 8F 47 01 00 00 07 E0 C692	ISO15693 Command Flags "Stay Quiet" Command UID CRC
BCC	EF 10	LRC and ~LRC

Response Packet: (**01 09 00 03 04 45 01 4B B4**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	45	Pass-through Request
Status	01	Token Not Present
BCC	4B B4	LRC and ~LRC

Pass Through Read Single Block Example:

Send a valid ISO15693 Pass-through "Read Single Block" Request Packet to the Mfr module application and validate the data received. Have a valid ISO15693 compliant token in the field when the request is issued. This example shows a token with Unique ID code **E3 8F 47 01 00 00 07 E0**. Therefore, the UID (retrieved from the Find Token Request), CRC and BCC bytes should be changed to match the token currently being interrogated.

Request Packet: (**01 15 00 03 04 45 23 20 E3 8F 47 01 00 00 07 E0 02 39 84 27 D8**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	15 00	Packet Length 21 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	45	Pass-through Request
Data	23 20 E3 8F 47 01 00 00 07 E0 02 3984	ISO15693 Command Flags "Read Single Block" Command UID Memory Block 2 CRC
BCC	27 D8	LRC and ~LRC

Response Packet: (**01 10 00 03 04 45 00 00 6C 77 54 7F 9D F2 0C F3**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	10 00	Packet Length 16 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	45	Pass-Through Reply
Status	00	ERROR_NONE
Data	00 6C 77 54 7F 9DF2	ISO15693 Response Flags Block Data CRC
BCC	0C F3	LRC and ~LRC

Pass Through Unaddressed Read Single Block Example:

Send a valid ISO15693 Pass-through "Read Single Block" Request Packet to the Mfr module application and validate the data received. Do not specify the UID of the token to read. Ensure a valid ISO15693 compliant token is in the field when the request is issued.

Request Packet: (**01 0D 00 03 04 45 03 20 01 12 1B 65 9A**)

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	0D 00	Packet Length 13 bytes
Device ID	03	Terminal is MFR Module
Command 1	04	Library Layer – ISO15693
Command 2	45	Pass-through Request
Data	03 20 01 121B	ISO15693 Command Flags "Read Single Block" Command Memory Block 1 CRC
BCC	65 9A	LRC and ~LRC

Response Packet: (**01 10 00 03 04 45 00 00 6C 77 54 7F 9D F2 0C F3**)

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	10 00	Packet Length 16 bytes
Device ID	03	Terminal is MFR Module
Cmd1	04	Library Layer – ISO15693
Cmd2	45	Pass-Through Reply
Status	00	ERROR_NONE
Data	00 6C 77 54 7F 9DF2	ISO15693 Response Flags Block Data CRC
BCC	0C F3	LRC and ~LRC

Transmitter On Request (48_{16})

Transmitter On Request is used to turn the transmitter ON, for a specific entity. The request packet specifies in the <*Cmd1*> field the intended entity. The application layer firmware redirects the request to the intended entity. The entity then sets up the appropriate hardware and software configurations needed to implement its specific details and then turns the transmitter on.

The ISO15693 Transmitter On request first configures the hardware ports to their proper data directions (input/output), sets up pull-up resistors for input ports and sets up the correct logic levels at the output ports. The transmitter itself is turned on and then the entity communicates with the HF ASIC to set up the ISO15693 protocol related register settings in the HF ASIC.

The request packet has no data bytes. The <*Cmd1*> field indicates the entity intended to turn on the transmitter. The response status byte will indicate the success/error status.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	48	Transmitter On

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	48	Transmitter On
Status	1	00 – FF	Standard Error Codes

Transmitter ON Example:

Request Packet: (**01 08 00 03 04 48 46 B9**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	08 00	Packet Length 8 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	48	Transmitter On Request
BCC	46 B9	LRC and ~LRC

Response Packet: (**01 09 00 03 04 48 00 47 B8**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd 2	48	Transmitter On Request
Status	00	ERROR_NONE
BCC	47 B8	LRC and ~LRC

Transmitter Off Request (49_{16})

Transmitter Off Request is used to turn the transmitter OFF for a specific entity. The request packet specifies in the <*Cmd1*> field the intended entity. The application layer firmware redirects the request to the intended entity. The entity then turns the transmitter off.

The ISO15693 entity turns the transmitter off and returns a response. The request packet consists of no data bytes. The <*Cmd1*> field indicates the entity intended to turn off the transmitter. The response status byte will indicate the success/error status.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	49	Transmitter Off

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	49	Transmitter Off
Status	1	00 – FF	Standard Error Codes

Transmitter OFF Example:
Request Packet: (**01 08 00 03 04 49 47 B8**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	08 00	Packet Length 8 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	49	Transmitter Off Request
BCC	47 B8	LRC and ~LRC

Response Packet: (**01 09 00 03 04 49 00 46 B9**)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR Module
Cmd1	04	ISO15693 Entity ID
Cmd2	49	Transmitter Off Request
Status	00	ERROR_NONE
BCC	46 B9	LRC and ~LRC

Set Parameters Request (61_{16})

The Set Parameters Request is used to specify the desired RF communication options permitted in the ISO15693 standard. Although all compliant transponders can support all communication modes, some configurations may provide optimal performance over others. If this request is not used, the S4100 will use the following default configuration:

High Baud Rate, FM uplink, 10% downlink modulation with 1-of-4 coding

This command does not send an RF message, but could affect all subsequent RF communication with the ISO15693 Library, until another 'Set Parameters' Request is sent.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	61	Set Parameters
Baud	1	00 – FF	0 : Low Baud Rate 1 : High Baud Rate (other) : Default Baud Rate
UplnkMod	1	00 – FF	0 : Amplitude Modulation (AM) 1 : Frequency Modulation (FM) (other) : Default Modulation
DnlnkDepth	1	00 – FF	0 : 10% Modulation Depth 1 : 100% Modulation Depth (other) : Default Modulation Depth
DnlnkCode	1	00 – FF	0 : 1 of 4 1 : 1 of 256 (other) : Default Coding
NonVolatile	0 / 1	00 – FF	00 : Do Not Preserve Settings 01 – FF : Preserve Settings over Reset *(Note: If field not present, effects are identical to value of 00)*

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	61	Set Parameters
Status	1	00 – FF	Standard Error Codes

Set Parameters Example:
Request Packet: (**01 0D 00 03 04 61 00 FF 01 FF 01 6A 95)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0D 00	Packet Length 13 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	61	Set Parameter Request
Baud	00	Low Baud Rate
UplnkMod	FF	Default (FM) Uplink Modulation
DnlnkDepth	01	100% Downlink Depth
DnlnkCode	FF	Default (1 of 4) Data Coding
NotVolatile	01	Store Settings in EEPROM
BCC	6A 95	LRC and ~LRC

Response Packet: **(01 09 00 03 04 61 00 6E 91)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	61	Set Parameter
Status	00	ERROR_NONE
BCC	6E 91	LRC and ~LRC

Inventory Request (62_{16})

The Inventory Request is used to acquire the Unique ID's of ISO15693 transponders in the read zone. Two inventory methods are supported: slotted, and non-slotted. A non-slotted request allows all transponders in the read zone to reply to a single command. In cases where more than one token is present, such a request would cause a data collision and yield no discernable response. A slotted inventory sequence decreases the likelihood of a data collision by forcing compliant transponders to respond in 1 of 16 slots based on a portion of their Unique ID's. To perform a slotted sequence, the 'Slot Marker / End-Of-Frame' Request is used in conjunction with this request. Any collision that does occur in a slotted sequence can be further arbitrated by using the Anti-collision mask in an algorithm similar to that outlined in the ISO15693 standard.

Information:
When requesting the 16-slot method, the ISO15693 transmitter remains **ON** in order to preserve the token states changed by the request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	62	Inventory
IsOneSlot	1	00 – FF	0 : Initiate 16-slot sequence (other) : Perform 1-slot request
AfiFlg	1	00 – FF	0 : AfiVal field is not present (other) : Transmit AfiVal in DnInk
AfiVal	0 / 1	00 – FF	Application Family ID, sent in command if previous field is Non-Zero
MskLen	1	00 – 08	Length of next field
MskVal	0 - 8	00 – FF	Anti-collision mask per ISO15693-3

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	62	Inventory
Status	1	00 – FF	Standard Error Codes
InvReply	0 / 10	00 – FF	Byte 1: Inventory Response Flags
		00 – FF	2: DSFID
		00 – FF	3-10: Unique ID

Inventory Example:
Request Packet: **(01 0B 00 03 04 62 01 00 00 6E 91)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	62	Inventory Command Request
IsOneSlot	01	1 Slot
AfiFlg	00	No AFI byte included
MskLen	00	Mask Length = 00
BCC	6E 91	LRC and ~LRC

Response Packet: **(01 13 00 03 04 62 00 00 00 E5 B0 81 06 00 00 07 E0 42 BD)**

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	13 00	Packet Length 19 bytes
Device ID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	62	Inventory
Status	00	ERROR_NONE
Inv. Resp. Flags	00	No Error
DSFID	00	Data Storage Format ID
UID	E5 B0 81 06 00 00 07 E0	ID of the token (LSB) first
BCC	42 BD	LRC and ~LRC

Slot Marker / End-Of-Frame Request (63_{16})

The Slot Marker / End-Of-Frame Request is used in conjunction with other requests as part of an exchange sequence. Its purpose is to send the ISO15693 End-Of-Frame marker to transponders within read range and await a response. In the Inventory process, the EOF is interpreted as the "Slot Marker" for anti-collision. The EOF marker is also used in conjunction with pass-thru write and lock operations in order to poll the token for a response if that form of response is requested in the request flags.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	63	Slot Marker / End-Of-Frame

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	63	Slot Marker / End-Of-Frame
Status	1	00 – FF	Standard Error Codes
RplyData: InvReply[1]	10	00 – FF	Byte 1: Inventory Response Flags
		00 – FF	2: DSFID
		00 – FF	3-10: Unique ID
— OR —			
RplyData: AckResp[2]	1	00 – FF	Byte 1: ISO15693 Response Flags
— OR —			
RplyData: ErrorResp[3]	2	00 – FF	Byte 1: ISO15693 Response Flags
		00 – FF	2: Error Code
— OR —			
RplyData: NoData[4]	0	—	(See <*Status*> field)

(1) Contents with valid Inventory Sequence Reply
(2) Contents with Pass-Through Write/Lock Sequence token Reply “Success”
(3) Contents with Pass-Through Write/Lock Sequence token Reply “Error”
(4) No Data returned due to condition described in <*Status*> field

Slot Marker/ EOF Example:

Request Packet: **(01 0B 00 03 04 62 00 00 00 6F 90)** (Inventory: Slot #00)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	62	Inventory
IsOneSlot	00	16-Slot Sequence
AfiFlg	00	No AFI byte included
MskLen	00	Mask Length = 00
BCC	6F 90	LRC and ~LRC

Response Packet: **(01 09 00 03 04 62 01 6C 93)**

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	09 00	Packet Length 9 bytes
Device ID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	62	Inventory Command
Status	01	ERROR_TOKEN_NOT_PRESENT
BCC	6C 93	LRC and ~LRC

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #01)

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	08 00	Packet Length 8 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	63	Slot Marker / EOF
BCC	6D 92	LRC and ~LRC

Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	09 00	Packet Length 9 bytes
Device ID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	63	Inventory Command
Status	01	ERROR_TOKEN_NOT_PRESENT
BCC	6D 92	LRC and ~LRC

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #02)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #03)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #04)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #05)
Response Packet: **(01 13 00 03 04 63 00 00 00 E5 B0 81 06 00 00 07 E0 43 BC)**

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	13 00	Packet Length 19 bytes
Device ID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	63	Slot Marker / EOF
Status	00	ERROR_NONE
Inv. Resp. Flags	00	No Error
DSFID	00	Data Storage Format ID
UID	E5 B0 81 06 00 00 07 E0	ID of the token (LSB) first
BCC	43 BC	LRC and ~LRC

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #06)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #07)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #0C)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #0D)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #0E)
Response Packet: **(01 13 00 03 04 63 00 00 00 CE B1 81 06 00 00 07 E0 69 96)**

Field	Contents	Summary
SOF	01	Start of Frame
Packet Length	13 00	Packet Length 19 bytes
Device ID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	63	Slot Marker / EOF
Status	00	ERROR_NONE
Inv. Resp. Flags	00	No Error
DSFID	00	Data Storage Format ID
UID	CE B1 81 06 00 00 07 E0	ID of the token (LSB) first
BCC	69 96	LRC and ~LRC

Request Packet: **(01 08 00 03 04 63 6D 92)** (Slot Marker: Slot #0F)
Response Packet: **(01 09 00 03 04 63 01 6D 92)**

— END 16-SLOT INVENTORY SEQUENCE —

Stay Quiet Request (64_{16})

The Stay Quiet Request is used to silence a token, preventing it from responding to any non-addressed or inventory related requests. The token will, however, respond to requests with matching UID. As there is no response to this request from the receiving token, only request status and errors are reported.

Information:
The ISO15693 transmitter remains **ON** in order to preserve the token states changed by this request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	64	Stay Quiet
UID	8	00 – FF (ea)	Unique ID of token to silence

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	64	Stay Quiet
Status	1	00 – FF	Standard Error Codes

Stay Quiet Example:

Request Packet: **(01 10 00 03 04 64 B8 9A 92 06 00 00 07 E0 23 DC)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	10 00	Packet Length 16 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	64	Stay Quiet
UID	B8 9A 92 06 00 00 07 E0	Unique ID of token
BCC	23 DC	LRC and ~LRC

Response Packet: **(01 09 00 03 04 64 00 6B 94)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	64	Stay Quiet
Status	00	ERROR_NONE
BCC	6B 94	LRC and ~LRC

Read Single Block Request (65_{16})

The Read Single Block Request gets the data from one memory block of the responding token. In addition to this data, a Block Security Status byte can be requested. This byte shows the write-protection of the block specified (i.e. Unlocked, (User/Factory) Locked, etc.).

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	65	Read Single Block
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
ReqSecurity	1	00 – FF	0 : No Block Security Status (other) : Request Blk. Security Status
BlkNum	1	00 – FF	Specifies block that is to be read
UID[1]	0 / 8	00 – FF	Unique ID of token

[1] If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
Reply	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	65	Read Single Block
	Status	1	00 – FF	Standard Error Codes
	StdResp[1]	n[2]	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2-n[2]: Read Data
	— OR —			
	SecureResp[3]	n[2]	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Block Security Status
			00 – FF	3-n[2]: Read Data
	— OR —			
	ErrorResp[4]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[5]	0	-	(See <Status> field)

[1] Response when token responds with data read from its memory
[2] The value of 'n' varies with the block size for the specific token read
[3] Response when token responds with Block Security Status and memory data
[4] Response when error flag is set in token reply
[5] No Data returned due to condition described in <*Status*> field

Read Single Block Example:
Request Packet: **(01 13 00 03 04 65 00 01 05 B8 9A 92 06 00 00 07 E0 25 DA)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	13 00	Packet Length 19 bytes
DeviceID	03	Device is MFR
Cmd 1	04	ISO15693 Entity ID
Cmd 2	65	Read Single Block
IsSelectMsg	00	Not a select message
ReqSecurity	01	Return Security Status
BlkNum	05	Block Number 5
UID	B8 9A 92 06 00 00 07 E0	Unique ID of token
BCC	25 DA	LRC and ~LRC

Response Packet: **(01 0F 00 03 04 65 00 00 00 12 34 56 78 64 9B)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0F 00	Packet Length 15 bytes
DeviceID	03	Device is MFR
Cmd 1	04	ISO15693 Entity ID
Cmd 2	65	Read Single Block
Status	00	ERROR_NONE
Blk. Security Stat.	00	Block unlocked
Read Data	12 34 56 78	Data read from block 5
BCC characters	64 9B	LRC and ~LRC

Write Single Block Request (66_{16})

The Write Single Block Request writes data to one memory block of the addressed token(s). In order to successfully write data, the Host must know the size of the memory block of the token. This information is available through the 'Get System Information' Request, if supported by the token. A corrupted response or lack of response does not necessarily indicate a failure to perform the write operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	66	Write Single Block
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other) : Polled Reply (Prog. Burst)
BlkNum	1	00 – FF	Specifies block that is to be written
BlkBytes	1	00 – 1F	Length of next field (token dependent)
BlkData	0 – 32[2]	00 – FF	Data to be written to specified block
UID[3]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders
(2) Length specified by previous field
(3) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
Rply	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	66	Write Single Block
	Status	1	00 – FF	Standard Error Codes
	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
	— OR —			
	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Write Single Block Example (non-addressed write operation):

Request Packet: **(01 10 00 03 04 66 00 01 05 04 12 35 36 38 59 A6)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	10 00	Packet Length 16 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	66	Write Single Block
IsSelectMsg	00	Not a select message
RespType	01	Poll for a reply (prog. burst)
BlkNum	05	Block Number 5
BlkBytes	04	4 bytes per block
BlkData	12 35 36 38	Data to write to block
BCC	59 A6	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 66 00 00 6A 95)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	66	Write Single Block
Status	00	ERROR_NONE
AckResp	00	Successful write
BCC	6A 95	LRC and ~LRC

Lock Block Request (67_{16})

The Lock Block Request command write-protects one memory block of the addressed token(s). A corrupted response or lack of response does not necessarily indicate a failure to perform the lock operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	67	Lock Block
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other) : Polled Reply (Prog. Burst)
BlkNum	1	00 – FF	Specifies block that is to be locked
UID[2]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders
(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
RplyDat	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	67	Lock Block
	Status	1	00 – FF	Standard Error Codes
	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
	— OR —			
	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Lock Block Example:

Request Packet: **(01 13 00 03 04 67 00 01 0B 90 AE 81 06 00 00 07 E0 26 D9)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	13 00	Packet Length 19 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	67	Lock Block
IsSelectMsg	00	Not a select message
RespType	01	Polled reply (prog. burst)
BlkNum	0B	Lock block 11
UID	90 AE 81 06 00 00 07 E0	Unique ID of token
BCC	26 D9	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 67 00 00 6B 94)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	67	Lock Block
Status	00	ERROR_NONE
AckResp	00	Successful lock
BCC	6B 94	LRC and ~LRC

Read Multiple Blocks Request (68_{16})

The Read Multiple Blocks Request gets the data from multiple memory blocks of the responding token. In addition to this data, a Block Security Status byte can be requested for each block. This byte shows the write-protection of the block specified (i.e. Unlocked, (User/Factory) Locked, etc.).

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	68	Read Multiple Blocks
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
ReqSecurity	1	00 – FF	0 : No Block Security Status (other) : Request Blk. Security Status
StartBlk	1	00 – FF	Specifies first block that is to be read
NumBlks	1	00 – FF	Number of blocks to read after 1st block
UID(1)	0 / 8	00 – FF	Unique ID of token

(1) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
Reply	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	68	Read Multiple Blocks
	Status	1	00 – FF	Standard Error Codes
	StdResp(1)	n(2)	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2-n(2): Concatenated Block Read Data
	— OR —			
	SecureResp(3)	n(4)	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Block Security Status
			00 – FF	3-s(5): Block Read Data
			(Both fields repeated <NumBlks> times)	
	— OR —			
	ErrorResp(6)	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData(7)	0	-	(See <Status> field)

(1) Response when token responds with data read from its memory
(2) n = [(<*NumBlks*> + 1) × (Token Block Size)] + 1
(3) Response when token responds with memory data and Block Security Status
(4) n = [(<*NumBlks*> + 1) × (Token Block Size + 1)] + 1
(5) (<*Block Read Data*> length = Token Block Size) ∴ s = Token Block Size + 2
(6) Response when error flag is set in token reply
(7) No Data returned due to condition described in <*Status*> field

Read Multiple Blocks Example:

Request Packet: **(01 0C 00 03 04 68 00 01 00 07 75 8A)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0C 00	Packet Length 12 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	68	Read Multiple Blocks
IsSelectMsg	00	Not a select message
ReqSecurity	01	Return Security Status
StartBlk	00	Start reading at block 0
NumBlks	07	Read next 7 blocks also
BCC characters	75 8A	LRC and ~LRC

Response Packet: (**01 32 00 03 04 68 00 00 00 44 33 22 11 00 00 00 00 00 00 04 03 02 01 00 00 00 00 00 00 00 00 00 00 01 00 00 30 86 00 00 00 00 00 00 00 00 00 00 AB 54**)

	Field	Contents	Summary
	SOF	01	Start of Frame
	PacketLen	32 00	Packet Length 50 bytes
	DeviceID	03	Device is MFR
	Cmd1	04	ISO15693 Entity ID
	Cmd2	68	Read Multiple Blocks
	Status	00	ERROR_NONE
SecurResp	RespFlags	00	ISO15693 Resp. Flags
SecurResp	Block Security Status + Block Read Data	00 44 33 22 11 – Block 0 00 00 00 00 00 – Block 1 00 04 03 02 01 – Block 2 00 00 00 00 00 – Block 3 00 00 00 00 00 – Block 4 01 00 00 30 86 – Block 5 00 00 00 00 00 – Block 6 00 00 00 00 00 – Block 7	Block Security Status bytes followed by Block Read Data, repeated for each block requested *Note: Block 5 is locked.*
	BCC	AB 54	LRC and ~LRC

Write Multiple Blocks Request (69_{16})

The Write Multiple Blocks Request writes data to multiple memory blocks of the addressed token(s). In order to successfully write data, the Host must know the size of the memory block of the token. This information is available through the 'Get System Information' Request, if supported by the token. A corrupted response or lack of response does not necessarily indicate a failure to perform the write operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	69	Write Multiple Blocks
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
RespType	1	00 – FF	0 : Asynchronous Reply (other) : Polled Reply (Prog. Burst)
StartBlk	1	00 – FF	Specifies first block to write
NumBlks	1	00 – FF	No. of blocks to write after 1st block
BlkBytes	1	00 – 1F	Length of token memory block
BlkData	n(1)	00 – FF	Data to be written to specified blocks
UID(2)	0 / 8	00 – FF	Unique ID of token

(1) n = (<*NumBlks*> + 1) × <*BlkBytes*>
(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	69	Write Multiple Blocks
	Status	1	00 – FF	Standard Error Codes
RplyDat	AckResp(1)	1	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat	— OR —			
RplyDat	ErrorResp(2)	2	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat			00 – FF	2: Error Code
RplyDat	— OR —			
RplyDat	NoData(3)	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Write Multiple Blocks Example (non-addressed write operation):

Request Packet: **(01 15 00 03 04 69 00 01 02 01 04 12 35 36 38 21 53 63 83 C7 38)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	15 00	Packet Length 16 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	69	Write Multiple Blocks
IsSelectMsg	00	Not a select message
RespType	01	Poll for a reply (prog. burst)
StartBlk	02	Start writing at block 2
NumBlks	01	Write next block also
BlkBytes	04	4 bytes per block
BlkData	12 35 36 38 21 53 63 83	Data to write to block 2 Data to write to block 3
BCC	C7 38	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 66 00 00 65 9A)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	69	Write Multiple Blocks
Status	00	ERROR_NONE
AckResp	00	Successful write
BCC	65 9A	LRC and ~LRC

Select Request ($6A_{16}$)

The Select Request places the addressed token in the "Select" state. In this state, it responds to requests with the ISO15693 "Select Flag" set. This flag is directly controlled by the *<IsSelectMsg>* field present in many ISO15693 library request messages. Any receiving token currently in the "Select" state with UID not matching the value sent in the request, exits that state and enters the "Ready" state but does not send a reply.

Information:
The ISO15693 transmitter remains **ON** in order to preserve the token states changed by this request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6A	Select
UID	8	00 – FF	Unique ID of token

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6A	Select
	Status	1	00 – FF	Standard Error Codes
RplyDat	AckResp(1)	1	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat	— OR —			
RplyDat	ErrorResp(2)	2	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat			00 – FF	2: Error Code
RplyDat	— OR —			
RplyDat	NoData(3)	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Select Example:
Request Packet: **(01 10 00 03 04 6A E5 B0 81 06 00 00 07 E0 49 B6)**

Field	**Contents**	**Summary**
SOF	01	Start of Frame
PacketLen	10 00	Packet Length 16 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6A	Select
UID	E5 B0 81 06 00 00 07 E0	Unique ID of token
BCC	49 B6	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 6A 00 00 66 99)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6A	Select Command
Status	00	ERROR_NONE
AckResp	00	Successful select
BCC	66 99	LRC and ~LRC

Reset To Ready Request ($6B_{16}$)

The Reset To Ready Request places the addressed token in the "Ready" state. In this state, it does not respond to requests with the ISO15693 "Select Flag" set, but to any non-addressed request or request matching its UID.

Information:
The ISO15693 transmitter state is not altered by this message, as the state requested is identical to the power-up state of the token.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6B	Reset to ready
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other): Set 'Select' flag in RF msg.
UID(1)	0 / 8	00 – FF	Unique ID of token

(1) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6B	Reset to ready
	Status	1	00 – FF	Standard Error Codes
RplyDat	AckResp(1)	1	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat	— OR —			
RplyDat	ErrorResp(2)	2	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat			00 – FF	2: Error Code
RplyDat	— OR —			
RplyDat	NoData(3)	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Reset to Ready Example:
Request Packet: **(01 09 00 03 04 6B 01 48 B7)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6B	Reset to Ready
IsSelectMsg	01	Send to 'Selected' token
BCC characters	48 B7	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 6B 00 00 67 98)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6B	Reset to Ready
Status	00	ERROR_NONE
AckResp	00	Successful execution
BCC characters	67 98	LRC and ~LRC

Write AFI Request ($6C_{16}$)

The Write AFI Request records a new value to the AFI register of the addressed token(s). A corrupted response or lack of response does not necessarily indicate a failure to perform the write operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6C	Write AFI
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other): Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other): Polled Reply (Prog. Burst)
AfiVal	1	00 – FF	Application Family ID
UID[2]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders
(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
RplyDat	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6C	Write AFI
	Status	1	00 – FF	Standard Error Codes
	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
	— OR —			
	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Write AFI Example: (AFI location already locked)

Request Packet: **(01 0B 00 03 04 6C 01 01 AA CB 34)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6C	Write AFI
IsSelectMsg	01	Not a select message
RespType	01	Polled reply (prog. burst)
AfiVal	AA	Application Family ID
BCC	CB 34	LRC and ~LRC

Response Packet: **(01 0B 00 03 04 6C 00 01 12 72 8D)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6C	Write AFI
Status	00	ERROR_NONE
ErrorResp	01	ISO15693 Error Flag
	12	Error: Locked Address
BCC characters	72 8D	LRC and ~LRC

Lock AFI Request ($6D_{16}$)

The Lock AFI Request write-protects the AFI register of the addressed token(s). A corrupted response or lack of response does not necessarily indicate a failure to perform the lock operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6D	Lock AFI
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other): Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other): Polled Reply (Prog. Burst)
UID[2]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders
(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6D	Lock AFI
	Status	1	00 – FF	Standard Error Codes
RplyDat	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat	— OR —			
RplyDat	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat			00 – FF	2: Error Code
RplyDat	— OR —			
RplyDat	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Lock AFI Example:

Request Packet: **(01 12 00 03 04 6D 00 01 A0 A7 81 06 00 00 07 E0 1F E0)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	12 00	Packet Length 18 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6D	Lock AFI
IsSelectMsg	00	Not a select message
RespType	01	Polled reply (prog. burst)
UID	A0 A7 81 06 00 00 07 E0	Unique ID of token
BCC	1F E0	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 6D 00 00 61 9E)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6D	Lock AFI
Status	00	ERROR_NONE
AckResp	00	Successful lock
BCC	61 9E	LRC and ~LRC

Write DSFID Request ($6E_{16}$)

The Write DSFID Request writes a new value in the DSFID register of the addressed token(s). A corrupted response or lack of response does not necessarily indicate a failure to perform the write operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6E	Write DSFID
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other) : Polled Reply (Prog. Burst)
DsfidVal	1	00 – FF	Data Storage Format IDentifier
UID[2]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders

(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
RplyDat	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6E	Write DSFID
	Status	1	00 – FF	Standard Error Codes
	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
	— OR —			
	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"

(2) Contents with token reply "Error"

(3) No Data returned due to condition described in <*Status*> field

Write DSFID Example:

Request Packet: **(01 0B 00 03 04 6E 00 01 AA C8 37)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6E	Write DSFID
IsSelectMsg	00	Not a select message
RespType	01	Polled reply (prog. burst)
DSFID	AA	Data Storage Format ID
BCC characters	C8 37	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 6E 00 00 62 9D)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6E	Write DSFID
Status	00	ERROR_NONE
AckResp	00	Successful write
BCC	62 9D	LRC and ~LRC

Lock DSFID Request ($6F_{16}$)

The Lock DSFID Request write-protects the DSFID register of the addressed token(s). A corrupted response or lack of response does not necessarily indicate a failure to perform the lock operation. Additionally, multiple transponders may process a non-addressed request.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	6F	Lock DSFID
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other): Set 'Select' flag in RF msg.
RespType[1]	1	00 – FF	0 : Asynchronous Reply (other): Polled Reply (Prog. Burst)
UID[2]	0 / 8	00 – FF	Unique ID of token

(1) A Non-zero value is required for Tag-it™ HF-I transponders
(2) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	6F	Lock DSFID
	Status	1	00 – FF	Standard Error Codes
RplyDat	AckResp[1]	1	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat	— OR —			
RplyDat	ErrorResp[2]	2	00 – FF	Byte 1: ISO15693 Response Flags
RplyDat			00 – FF	2: Error Code
RplyDat	— OR —			
RplyDat	NoData[3]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) Contents with token reply "Error"
(3) No Data returned due to condition described in <*Status*> field

Lock DSFID Example:

Request Packet: **(01 12 00 03 04 6F 00 01 B8 9A 92 06 00 00 07 E0 2B D4)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	12 00	Packet Length 18 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6F	Lock DSFID
IsSelectMsg	00	Not a select message
RespType	01	Polled reply (prog. burst)
UID	B8 9A 92 06 00 00 07 E0	Unique ID of token
BCC	2B D4	LRC and ~LRC

Response Packet: **(01 0A 00 03 04 6F 00 00 63 9C)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0A 00	Packet Length 10 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	6F	Lock DSFID
Status	00	ERROR_NONE
AckResp	00	Successful lock
BCC	63 9C	LRC and ~LRC

Get System Information Request (70_{16})

The Get System Information Request retrieves identification, application family, and data formatting and sizes as specified in the ISO15693 standard.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	70	Get System Information
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
UID[1]	0 / 8	00 – FF	Unique ID of token

(1) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	70	Get System Information
	Status	1	00 – FF	Standard Error Codes
RplyDat	InfoResp[1]	10 – 15	00 – FF	Byte 1: ISO15693 Response Flags
			00 – 0F	2: Info Flags[2]
			00 – FF	3-10: UID (Unique ID of token)
			00 – FF	… : Information fields[2]
	— OR —			
	ErrorResp[3]	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData[4]	0	—	(See <Status> field)

(1) Contents with token reply "Success"
(2) See ISO15693 Standard for details on this field
(3) Contents with token reply "Error"
(4) No Data returned due to condition described in <*Status*> field

Get System Information Example:

Request Packet: **(01 09 00 03 04 70 00 7F 80)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	70	Get System Information
IsSelectMsg	00	Not a select message
BCC	7F 80	LRC and ~LRC

Response Packet: **(01 18 00 03 04 70 00 00 0F E5 B0 81 06 00 00 07 E0 AC AF 3F 03 88 E3 1C)**

	Field	Contents	Summary
	SOF	01	Start of Frame
	PacketLen	18 00	Packet Length 24 bytes
	DeviceID	03	Device is MFR
	Cmd1	04	ISO15693 Entity ID
	Cmd2	70	Get System Information
	Status	00	ERROR_NONE
	RespFlags	00	Flags (no errors)
InfoResp	InfoFlags	0F	DSFID, AFI, Memory Size and IC ref supported
InfoResp	UID	E5 B0 81 06 00 00 07 E0	Unique ID of token
InfoResp	DSFID	AC	Data Storage Format ID
InfoResp	AFI	AF	Application Family ID
InfoResp	VICCMemorySize	3F 03	64 blocks @ 4 bytes each
InfoResp	IcReference	88	Manufacturer Info.
InfoResp	BCC	E3 1C	LRC and ~LRC

Get Multiple Block Security Status Request (71_{16})

The Get Multiple Block Security Status Information Request gets a Block Security Status byte for each block requested. This byte encodes the write-protection of the block specified (i.e. Unlocked, (User/Factory) Locked, etc.).

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	71	Get Multiple Block Security Status
IsSelectMsg	1	00 – FF	0 : Do not set 'Select' flag (other) : Set 'Select' flag in RF msg.
StartBlk	1	00 – FF	Specifies first block that is to be read
NumBlks	1	00 – FF	Number of blocks to read after 1st block
UID(1)	0 / 8	00 – FF	Unique ID of token

(1) If UID field is not present, all transponders in read zone are addressed

Response Packet:

	Field	Length (Bytes)	Value Range (Hex)	Description
RplyDat	Cmd1	1	04	ISO15693 Entity ID
	Cmd2	1	71	Get Multiple Block Security Status
	Status	1	00 – FF	Standard Error Codes
	StdResp(1)	n(2)	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2-n(2): Block Security Status
	— OR —			
	ErrorResp(3)	2	00 – FF	Byte 1: ISO15693 Response Flags
			00 – FF	2: Error Code
	— OR —			
	NoData(4)	0	—	(See <Status> field)

(1) Response when token replies with Block Security Status for blocks requested
(2) n = <*NumBlks*> + 1
(3) Response when error flag is set in token reply
(4) No Data returned due to condition described in <*Status*> field

Get Security Status Example:

Request Packet: **(01 0B 00 03 04 71 00 00 3F 43 BC)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	71	Get Security Status
IsSelectMsg	00	Not a select message
StartBlk	00	Start reading at block 0
NumBlks	3F	Read next 63 blocks also
BCC	43 BC	LRC and ~LRC

Response Packet: (**01 4A 00 03 04 71 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3C C3**)

	Field	Contents	Summary
	SOF	01	Start of Frame
	PacketLen	4A 00	Packet Length 74 bytes
	DeviceID	03	Device is MFR Lite
	Cmd1	04	ISO15693 Entity ID
	Cmd2	71	Get Security Status
	Status	00	ERROR_NONE
StdResp	RespFlags	00	ISO15693 Resp. Flags
StdResp	BlockSecurityStat	00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Concatinated Block Security Status bytes for 64 blocks (block 2 is locked, all others are unlocked)
	BCC	3C C3	LRC and ~LRC

Set HF Timing (72_{16})

The Set HF Timing Request adjusts timing parameters of S6700 HF Transceiver used on the MFR module. When the MFR module is connected to a new antenna, the properties of the antenna can introduce variations in phase of the received signal. To account for this, the MFR module's read timing can be adjusted to increase or reduce the time delay before scanning for an ISO15693 response.

When adjusting the MFR for optimal performance with new antenna designs, the designer may be required to traverse the range of the <BscanDelay> (or Boundary Scan Delay) field, attempting to read multiple tokens after each trial. The <LtcDelay> field adjusts the timing of AM uplink detection only, and is usually not adjusted. Once an optimal setting is found, the timing values can be stored on non-volatile memory so that performance will be maintained over power-up or other reset conditions. The same settings can be used in subsequently produced devices as long as the antenna properties do not vary widely from unit to unit in production.

Request Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	Entity ID
Cmd2	1	72	Set HF Timing
LtcDelay	1	00 – FF	00 – 1F : Delay in increments of 2/13.56MHz = 147.5 nsec (other) : (Do Not Change Setting)
BscanDelay	1	00 – FF	Specifies first block that is to be read 00 – 7F : Delay in increments of 4/13.56MHz = 295 nsec (other) : (Do Not Change Setting)
NonVolatile	1	00 – FF	00 : Do Not Preserve Settings (other) : Preserve Settings over Reset

Response Packet:

Field	Length (Bytes)	Value Range (Hex)	Description
Cmd1	1	04	ISO15693 Entity ID
Cmd2	1	72	Set HF Timing
Status	1	00 – FF	Standard Error Codes

Set HF Timing Example:

Request Packet: **(01 0B 00 03 04 72 18 2C 01 4A B5)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	0B 00	Packet Length 11 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	72	Set HF Timing
LtcDelay	18	24 X 147.5 nsec = 3.54 µsec
BscanDelay	2C	44 X 295 nsec = 12.98 µsec
NonVolatile	01	Preserve Settings over Reset
BCC	4A B5	LRC and ~LRC

Response Packet: **(01 09 00 03 04 72 00 7D 82)**

Field	Contents	Summary
SOF	01	Start of Frame
PacketLen	09 00	Packet Length 9 bytes
DeviceID	03	Device is MFR
Cmd1	04	ISO15693 Entity ID
Cmd2	72	Set Parameter
Status	00	ERROR_NONE
BCC	7D 82	LRC and ~LRC

Chapter 2

Regulatory and Warranty Notices

Topic **Page**

2.1 FCC Conformity

The Series 4000 Multi-Function Reader is an intentional radiator. The transmitter portion operates at 13.56 MHz and is subject to FCC Part 15, Subpart C, “Intentional Radiator,” paragraph 15.225 (13.553-13.567MHz). Radiated emissions from the device are subject to the limits in Section 15.209 of the Rules outside of the 13.56 +/- 0.007 MHz band.

Note:

Any device or system incorporating the Series 4000 reader, in full or in part, needs to obtain FCC certification as part of the system within which this reader unit resides. A system containing this product may be operated only under an experimental license or final approval issued by the relevant approval authority. Before any such device or system can be marketed, an equipment authorization must be obtained form the relevant approval authority.

2.2 ETSI Conformity

Any device or system incorporating the Series 4000 reader, in full or in part, may need to comply with European Standard EN300330. It is the responsibility of each system integrator to have their complete system tested and to obtain approvals as required from the local authorities before operating or selling this system.

2.3 CE Conformity

Any device or system incorporating the Series 4000 reader, in full or in part, may need to have a CE Declaration of Conformity stating that it meets European EMC directive 99/5/EC. This must be issued by the system integrator or user of such a system prior to marketing or operating it in the European community.

2.4 Warranty and Liability

The "General Conditions of Sale and Delivery" of Texas Instruments Incorporated or a TI subsidiary apply. Warranty and liability claims for defect products, injuries to persons and property damages are void if they are the result of one or more of the following causes:

- Improper use of the reader module.
- Unauthorized assembly, operation and maintenance of the reader module.
- Operation of the reader modules with defective and/or non-functioning safety and protective equipment.
- Failure to observe the instructions during transport, storage, assembly, operation, maintenance and setting up of the reader modules.
- Unauthorized changes to the reader modules.
- Insufficient monitoring of the reader modules' operation or environmental conditions.
- Improperly conducted repairs.
- Catastrophes caused by foreign bodies and acts of God.