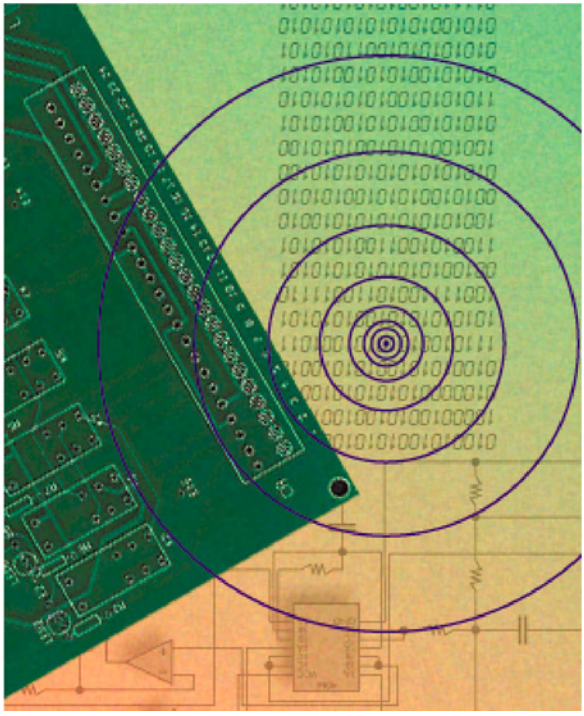
Reference Guide

Comprehensive Heuristic Unified Messaging Protocol



SAMSYS
THE POWER TO CHOOSE

SAMSys

Comprehensive Heuristic Unified Messaging Protocol Reference Guide

Third Edition (January 2004)

Disclaimer and Limitation of Liability

The content of this manual is for information use only and is subject to change without notice. SAMSys assumes no responsibility or liability for any errors or inaccuracies that may appear in this publication. No part of this manual may be reproduced in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of SAMSys.

SAMSys 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, property damage, or environmental damage.

About SAMSys

SAMSys is the world leader in the design and supply of Radio Frequency Identification (RFID) hardware solutions for high volume pallet and reusable container tracking applications in global logistics management, materials handling, and supply chain industries. SAMSys is a public company listed on the Canadian Venture Exchange under the symbol SMY.

SAMSys Technologies, Inc.
44 East Beaver Creek Rd., Unit 11
Richmond Hill, Ontario L4B 1G8 Canada
Phone: (905) 707-0404
Toll Free: (877) 463-6546
Fax: (905) 707-9944
E-mail: samsys@samsys.com
Web: www.samsys.com

Part Number: HI471-CH-RF-V3.2

Trademarks

SAMSys is a trademark of SAMSys Technologies, Inc.

HyperTerminal® is a registered trademark of Hilgraeve, Inc.

Tag-it™ is a trademark of Texas Instruments, Inc.

Intellitag® is a registered trademark of Intermec Technologies Corporation.

Other company, product or service names appearing in this manual may be trademarks or service marks of their respective owners.

Contents

Figures

Chapter 1

Introduction

This chapter provides a general overview of SAMSys reader programming and the Comprehensive Heuristic Unified Messaging Protocol (CHUMP). Topics discussed in this chapter include the following:

- RFID Reader Programming Overview
- What is an HDLC Protocol?
- CHUMP Protocol Overview

RFID Reader Programming Overview

SAMSys RFID readers are programmed and configured using a human-readable variation of the High-Level Data Link Control (HDLC) protocol. These commands are called the Comprehensive Heuristic Unified Messaging Protocol (CHUMP). CHUMP commands are issued to select reader operating modes as well as initiate specific actions.

Reader Operating Modes

SAMSys RFID readers are extremely flexible and easily programmed to function in several modes. These modes define the readers RF and serial communication characteristics. The RF modes are as follows:

- RF Continuous
- RF Polled

In RF Continuous mode, the reader periodically scans for tags over the air interface. In this mode, the reader does not require an external event or command to read a tag. A reader in RF Continuous mode is referred to as an active reader.

In RF Polled mode, the reader scans for tags only after the host sends a read command. In this mode, the reader remains idle until an external event initiates the read function. A reader in RF Polled mode is referred to as a passive reader.

In RF Continuous mode, SAMSys readers can further be programmed to operate in different serial communication modes. The serial communication modes are as follows:

- Serial Continuous
- Serial Polled

In Serial Continuous mode, the reader automatically transmits tag data (on the serial link) to the host. In Serial Polled mode, the reader stores the tag data until prompted by the host.

Other reader modes and configurations include the following:

- FORTH or CHUMP programming mode
- Serial multidrop addressing
- Protocol configuration
- Protocol Selection

For detailed information on the various operating modes and configurations, refer to the *Reader Configuration* chapter.

Reader Programming

SAMSys readers are programmed by sending specific CHUMP commands to the reader along the RS-485 serial link.

In most cases, the reader is connected to a host system such as a PC or server. A terminal emulation program such as HyperTerminal® can be used to issue CHUMP commands to the reader.

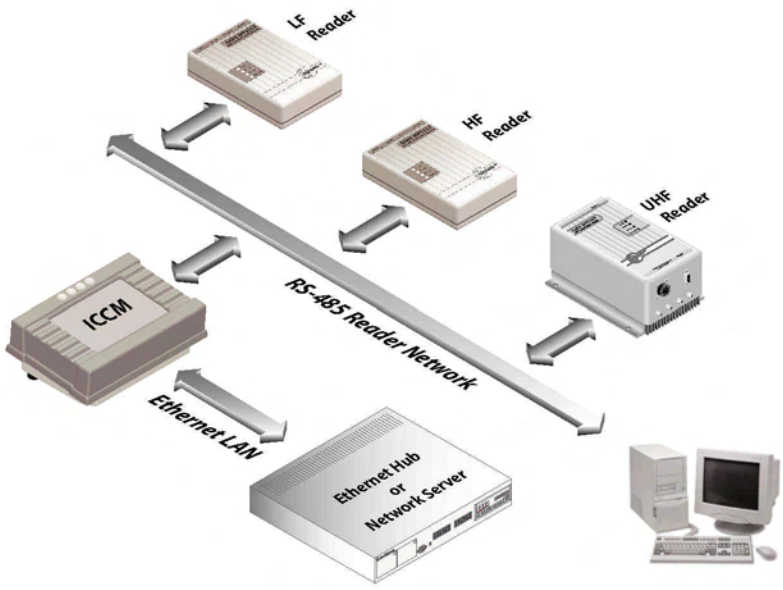


Figure 1 - *Sample Reader Network*

Typical CHUMP commands that are issued to the reader include the following:

- Read data from tag
- Write data to tag
- Halt/Quiet a tag
- Read the onboard I/O ports

For detailed information on the reader commands, refer to the *Command Descriptions* chapter.

What is an HDLC Protocol?

As explained in the *RFID Reader Programming Overview* section, CHUMP is a High Level Data Link Control (HDLC) protocol. HDLC is a class of general purpose protocols that fall within the Data Link Layer (Layer 2) of the Open Systems Interface (OSI) reference model. See Figure 2.

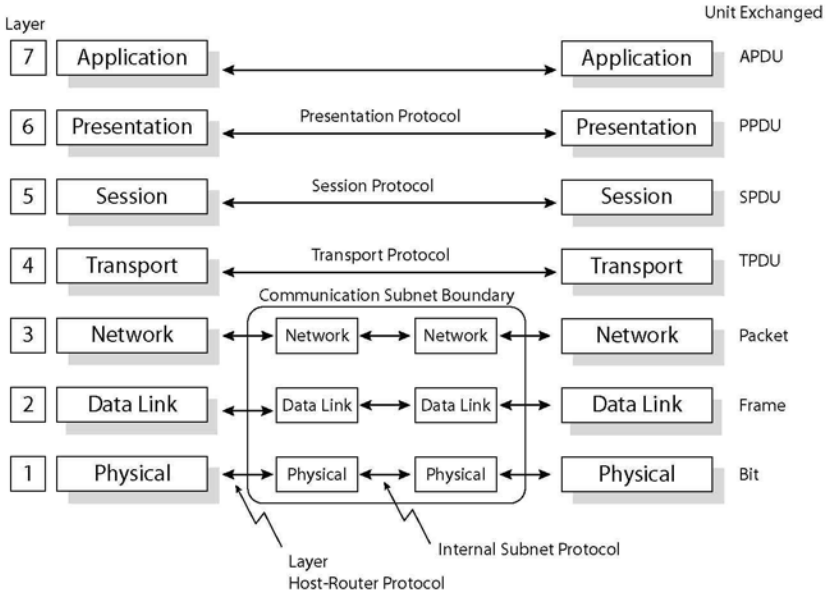


Figure 2 - *Open Systems Interface Reference Model*

Layer 2 provides a means to transfer data between network entities and correct transmission errors. The layer provides for activation, maintenance, and deactivation of data link connections, grouping of bits into characters and message frames, character and frame synchronization, error control, media access control, and flow control. Ethernet is a well known data link protocol.

Unlike Ethernet, which is generally used on Local Area Networks (LANs), HDLC protocols are primarily used on slower serial networks.

HDLC protocols typically include the following three types of stations:

- Primary Station - The host or master station that manages the data link.
- Secondary Station - A slave station that replies to the Primary Station.
- Combined Station - A combined master/slave station that functions as both Primary and Secondary Stations.

HDLC protocols operate in one of the following three modes:

- Normal Response Mode (NRM) - The Primary Station controls the data link.
- Asynchronous Response Mode (ARM) - The Secondary Station can talk first.
- Asynchronous Balanced Mode (ABM) - Multiple combined stations share the data link (multi-master mode).

In HDLC protocols, each piece of data is encapsulated in a frame by a trailer and a header (see Figure 3). The header contains an HDLC address and an HDLC control field. The trailer is found at the end of the frame and contains a Frame Check Sum (FCS). The FCS detects any errors that may occur during transmission. The frames are separated by HDLC flag sequences which are transmitted between each frame and whenever there is no data to be transmitted.

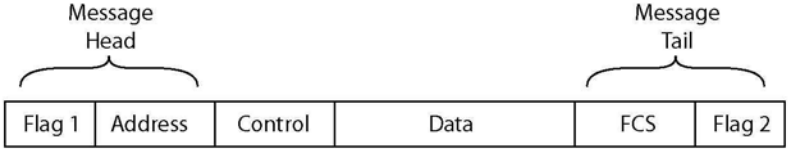


Figure 3 - *HDLC Frame Format*

As further explained in the following section, CHUMP is an HDLC protocol optimized for use in multidrop serial networks typically used for RFID readers. The SAMSys Interrogator Control and Concentrator Module (ICCM) is the Primary Station on a SAMSys reader network. The RFID readers are the Secondary Stations.

SAMSys readers can be configured to operate in Normal Response Mode (NRM) by selecting RF Polled Mode or RF Continuous/Serial Polled Mode. The readers can also be configured to operate in Asynchronous Response Mode (ARM) by selecting RF Continuous/Serial Continuous Mode.

CHUMP Protocol Overview

Unlike most HDLC protocols, CHUMP uses ASCII characters for the Flag, Address, Control, and FCS fields. Punctuation characters substitute for the more common (but unprintable) STX and ETX characters, providing a human readable message frame or packet.

Although not fully packetized, CHUMP can be described as operating between Layer 2 (Data Link Layer) and Layer 3 (Network Layer).

CHUMP Frame

As shown in Figure 4, each CHUMP frame consist of a Message Head, Command Block, and a Message Tail.

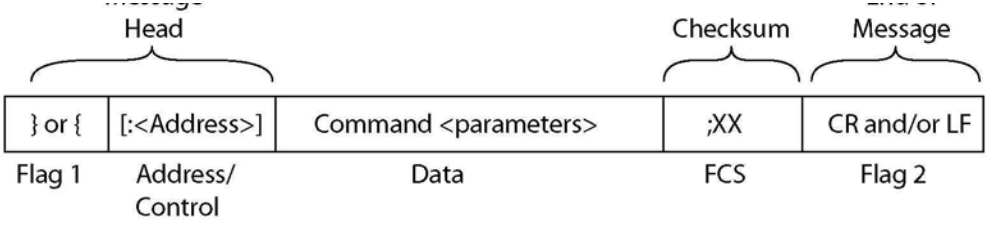


Figure 4 - *CHUMP Message Format*

Message Head

The message head contains a directional indicator and optional destination or source address information. The format is as follows:

`} [:dst_addr]` for commands (messages transmitted from the ICCM or host)

`{ [:src_addr]` for replies (messages received by the ICMM or host)

Message heads that include address information are referred to as multidrop message heads. Message heads without addresses are referred to as single-drop message heads

Command Block

The command block contains the CHUMP command and associated parameters. The command part of the CHUMP specification is a tagged, extensible, human readable protocol that directly implements OSI Layer 7 (Application Layer).

NOTE: *CHUMP command parameters and numeric data are expressed in Hexadecimal (hex) format. Refer to the individual command descriptions for the specific formats and syntax.*

Message Tail

The message tail contains a type indicator (**;** or **!**), an optional 8-bit Frame Checksum (FCS), and an end-of-packet indicator (**CR** and/or **LF**). The format is as follows:

;<FCS><CR and/or **LF>** for message tails with a checksum

!<CR and/or **LF>** for message tails without a checksum

The FCS is expressed as two ASCII bytes and is a sum of the bytes of the rest of the message. The FCS starts at the first character, "}" or "{", and ends at the semicolon FCS delimiter ";". Many of the shorter commands explained in the *Command Description* section include FCS values (refer to the Ro command).

NOTE: *The FCS can be eliminated for debug purposes by using the "!" message tail. However, installed systems should always use the FCS.*

Refer to Appendix B for sample checksum calculation code.

Example CHUMP Commands

The following example is a Read Version command sent to a reader. The `f:0` parameter returns the full version string with company, copyright, and date. The reply is shown in **bold**.

```
}Rv,f:0!
```

{Rv,d:"HFSARS_1.30[MB] Copyright (c) Samsys Technologies Inc., s:000000000000";3F

The next example is an ISO 15693 tag read for the value of the user writeable block 0x0f (=15):

```
}:07,Rt,d:E0070000015CFCCA,a:0f, t:IS15;d0
```

{:07,Rt,d:E0070000015CFCCA,b:F0123456, a:000F,t:IS15,f:0000;0D

The next example is a configuration read for the value of the General Configuration Word (GCW):

```
}:07,Cr,d:GCW;e5
```

{:07,Cr,d:GCW,b:00000007;32

NOTE: *CHUMP replies do not necessarily have all described parameters, fixed parameter order, or fixed field widths. For example: a:000F is the same as a:f*

Any software application that parses CHUMP replies should recognize punctuation and parameter names. The application should not rely on data position. Some readers may be configured to suppress tag type and flags to conserve serial bandwidth.

CHUMP Notation Conventions

The following notation conventions are used in CHUMP commands:

[...]	Message elements within square brackets denote optional information.
*****	Asterisk message elements can be repeated.
< ... >	Text within less than or greater than characters are short-hand descriptions of the actual value or character.
0x	Denotes a number in hexadecimal or base 16.

The following example is an ACK reply that may contain zero or more error codes:

{A[,e:xx]*;<FCS><CRLF>

NOTE: *Due to the extensible nature of the CHUMP command format, hosts can send parameters in any order. For example, the following commands are both valid and have the same meaning:*

}Ra,a:00,l:01!

}Ra,l:1,a:0!

Replies can also return parameters in any order. A reply parser should not assume that reply parameters are transmitted in any particular order.

Also, the CHUMP protocol is case insensitive. Upper or lower case letters can be used.

Chapter 2

CHUMP Commands

This chapter provides descriptions of the CHUMP commands and parameters. Topics discussed in this chapter include the following:

- Command Format
- Reader Management Commands
- Tag Commands
- Flow Control Commands
- Parameter Descriptions
- Command Descriptions

Command Format

CHUMP commands are typically one- or two-character commands with applicable parameters. As described in Chapter 1, these commands and replies are enclosed in framing characters. Refer to the following example for the format:

```
<Msg head> Command [<Parameter>] <Msg tail>
```

NOTE: *Command parameters and numeric data are typically expressed in hexadecimal (hex) format.*

Refer to the *Command Description* section for syntax, format, and parameter usage.

Reader Management Commands

Reader management commands control reader configuration, input/output ports, and display the software version. These commands are not tag or protocol dependent. The Reader Management commands are shown in the following table with the CHUMP software version where the command was first implemented.

Cmd	Description	Rev
Cr	Configuration read	1.30
Cw	Configuration write	1.30
Mr	Management reset	1.32
Ro	Read digital I/O port	1.30
Rv	Read version	1.20
Wo	Write digital I/O port	1.20

Tag Commands

Tag commands are protocol dependent and perform specific actions on a single tag or group of tags. The commands are shown in the following table with the applicable protocol and software version where the command was first implemented.

Cmd	Description	Rev	EM 4102/Temic	HiTag 1	TIRIS	tAG iT	ISO 15693	Icode	ISO 18000-6A	ISO 18000-6B	EM 4022/4222	EPC Alien	EPC Matrics
Ht	Halt or quiet tag	1.30					✓						
Lb	Lock block	1.30							✓	✓			
Ld	Lock DSFID	1.30							✓				
Lf	Lock AFI	1.30							✓				
Ra	Read user data (any tag)	1.20	2	✓	✓	✓	✓	✓	1		2		
Rd	Read detect	1.20	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Rf	Read by AFI	1.30							✓				
Rn	Read detect w/ new round	1.30							✓				
Rt	Read user data (specific tag)	1.20		✓		✓	✓	✓	✓	✓			
Ru	Read Unselect (Protocol B)	1.30								✓			
Tm	Tag mute	1.30							✓				
Tr	Tag reset	1.30							✓	✓			
Ts	Tag select	1.30							✓				
Tw	Tag wakeup	1.30							✓				
Wa	Write user data (to any tag)	1.20		✓	✓	✓	✓	✓	1				
Wd	Write DSFID	1.30							✓				
Wf	Write AFI	1.30							✓				
Wt	Write user data (specific tag)	1.20		✓	✓	✓	✓	✓	✓	✓			
Yl	Get lock block status	1.30							✓	✓			
Ys	Get system information	1.30							✓				

1 - Command works only after tag has been previously selected (Ts).
2 - If tag has no user data, the command becomes a Read Detect (Rd).

Flow Control Commands

Flow control commands are data link and transport layer commands that acknowledge data flow or request data retransmission. The Flow Control commands are shown in the following table with the software version where the command was first implemented.

Cmd	Description	Rev
A	Acknowledgement (Ack)	1.20
N	Negative Ack	1.20
S	Request resend of last serial msg	1.30

Parameter Descriptions

The parameters that may be sent with the CHUMP commands are as follows:

a: <address, offset, or location>

– Read or Write fields - start address

b: <HEX block data value or string>

– R/W fields - data (nn HEX data, "abc" ASCII data)
– Feedback field - text (b:"abc" ASCII data)

f:

– Operation specific low level flags

d: <tag id>

– Tag identification number

l:

– Length, weight, amount, time

t:

– Tag type (manufacturer, protocol). For example, ICP is ICode by Philips. See Chapter 3 - *Reader Configuration*, for more information on Protocl Select Words (PSW).

Additional parameters that may be received with CHUMP replies include the following:

c:<temp>

– First character is the sign (+ or -)
– Second and third are temperature in decimal

e: <error number>

– Error. Can be included in any message reply to indicate problems with the current or a previous message.

Command Descriptions

A

Name

Acknowledge (Ack)

Description

Transport layer acknowledgement within the message protocol. Message can also include error sequences.

Type

Flow control reply

Format

`{A;F7<CRLF>`

Return Message

`N/A`

Cr

Name

Configuration Read

Description

Reads configuration variable with a reply. If no data is available, an "`e:42`" sequence is returned. One or more "`e:42`" sequences can be returned with 2-digit HEX error codes.

If no "`d:`" sequence is specified, all configuration variables will be sent in individual Cr replies.

Type

Reader management command with reply

Format

```
}Cr[,d:<name>];<FCS><CRLF>
```

```
}Cr[,d:<name>]!<CRLF>
```

Return Message

```
{Cr,d:<name>,b:<value>;<FCS><CRLF>
```

```
{Cr[,e:xx]*;<FCS><CRLF>
```

Cw

Name

Configuration Write

Description

Writes configuration variable with reply.

Type

Reader management command with reply

Format

`}Cw[,d:<name>,b:<value>][,f:xx];`
`<FCS><CRLF>`

`}Cw[,d:<name>,b:<value>][,f:xx]! <CRLF>`

where:

`d:<name>` is an identifier with the 3-letter name of the configuration variable (CV).

`b:` is the block data (up to 4 bytes).

`f:` is the flag byte.

Return Message

`{A[,e:xx];<FCS><CRLF>`

Special Note

If `f:02`, all CVs are erased. At the next reset, the reader returns to factory defaults.

If `f:01`, data is written to Non-Volatile Memory (NVM).

If `d:` and `b:`, then the specified CV is written to NVM.

If `f:01` appears alone, all CVs are written to NVM.

Ht

Name

Halt

Description

Halts or quiets a specific tag.

Type

Tag command with reply

Format

`}Ht,d:<tag ID>[,t:<tag type>];  <FCS><CRLF>`

`}Ht,d:<tag ID>[,t:<tag type>]!<CRLF>`

where:

`d:<tag ID>` is the serial ID of requested tag.

`t:<tag type>` is the optional tag type.

Return Message

`{A[,e:xx]*;<FCS><CRLF>`

Lb

Name

Lock block

Description

For Protocol A tags, runs a Lock Block command. If the "`d:`" sequence is specified, the SUID flag is set and SUID bytes are sent, regardless of the configuration variable settings (see Chapter 3 - *Reader Configuration*). If the "`d:`" sequence is not specified, the tag must first be selected (see Ts command).

For Protocol B tags, triggers a LOCK command (opcode 0x0f). The `<tag ID>` is required for B tags.

Type

Tag command with reply

Format

`}Lb,a:xx[,d:<tag ID>];<FCS><CRLF>`

`}Lb,a:xx[,d:<tag ID>]!<CRLF>`

where:

`a:xx` is the block address.

`d:<tag ID>` is the serial ID of requested tag.

Return Message

`{A[,e:xx]*;<FCS><CRLF>`

Ld

Name

Lock DSFID

Description

If the “d:” sequence is specified, the SUID flag is set and SUID bytes are sent, regardless of the configuration variable settings (see Chapter 3 - *Reader Configuration*). If the “d:” sequence is not specified, the tag must first be selected (see Ts command).

Type

Tag command with reply

Format

`}Ld[,d:<tag ID>];<FCS><CRLF>`

`}Ld[,d:<tag ID>]!<CRLF>`

where:

`d:<tag ID>` is the serial ID of requested tag.

Return Message

`{A[,e:xx]*;<FCS><CRLF>`

Lf

Name

Lock AFI

Description

If the “d:” sequence is specified, the SUID flag is set and SUID bytes are sent, regardless of S register settings. If the “d:” sequence is not specified, the tag must first be selected (see Ts command).

Type

Tag command with reply

Format

```
}Lf[,d:<tag ID>];<FCS><CRLF>
```

```
}Lf[,d:<tag ID>]!<CRLF>
```

where:

`d:<tag ID>` is the serial ID of requested tag.

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Mr

Name

Management Reset

Description

Reboots the reader.

Type

Reader management command

Format

`}Mr!<CRLF>`

Return Message

`N/A`

N

Name

Negative Acknowledge (NAck)

Description

Low-level error message within the message protocol that indicates a serial message was received with errors. Typically, the message contains error codes.

Type

Flow control reply

Format

`{N;<FCS><CRLF>`

`{N[,e:xx]*;<FCS><CRLF>`

where:

`e:xx` is the error code.

Return Message

`N/A`

Ra

Name

Read user data from any tag.

Description

Read user data from any tag in field. If no "`a:`" sequence is included, the first block (block 00) is read. If no "`l:`" sequence is included, only one block is read.

Type

Tag command with reply

Format

```
}Ra[,a:<blk addr>][,f:xx][,l:<# blks>];
<FCS><CRLF>
```

```
}Ra[,a:<blk addr>][,f:xx][,l:<# blks>]!
<CRLF>
```

where:

`a:<blk addr>` is the tag-specific block adderess.

`f:xx` is an optional opcode override for Protocol B.

`l:<# blks>` is the tag-specific block size (1 or 4).

Return Message

```
{Ra,l:xx,d:<data>;<FCS><CRLF>
```

```
{Ra[,e:xx;]*;<FCS><CRLF>
```

NOTE: *ISO 18000-6A provides for a Fast Dialog Mode. In this mode, an inventory round is suspended when a tag is found to allow dialog with the tag (the tag is in a selected state during the dialog). A typical sequence is as follows:*

```
}Rd,t:IS186A,f:04!<CRLF>
```

```
{Rd,d:<tag id>,f:04;<FCS><CRLF> <tag dialog>
```

```
{Tn,f:04!<CRLF>  <resume inventory>
```

Rd

Name

Read detect

Description

Read detect command with replies.

If errors are waiting to be reported, one or more "`e:`" sequences can be returned.

Type

Tag command with reply

Format

`}Rd[,t:<tag type>];<FCS><CRLF>`

`}Rd[,t:<tag type>]!<CRLF>`

where:

`t:<tag type>` is the tag type indicator (typically used to increase search speed).

Return Message

`{Rd,d:<tag id>[,t:<tag type>]; <FCS><CRLF>`

`{Rd[,e:xx]*;<FCS><CRLF`

Rf

Name

Read by AFI

Description

ISO 18000-6A specific command that performs on inventory round by AFI. The value is configured into P6A.

Type

Tag command with reply

Format

```
}Rf[,t:<tag type>];<FCS><CRLF>
```

```
}Rf[,t:<tag type>]!<CRLF>
```

where:

`t:<tag type>` is the tag type indicator (typically used to increase search speed).

Return Message

```
{Rf,d:<tag id>[,t:<tag type>]; <FCS><CRLF>
```

```
{Rf[,e:xx]*;<FCS><CRLF>
```

Rn

Name

Read detect command with New Round

Description

This command is an ISO 18000-6A specific read detect command.

Type

Tag command with reply

Format

```
}Rn!<CRLF>
```

Return Message

```
{Rn,d:<tag id>[,t:<tag type>]; <FCS><CRLF>
```

```
{Rn[,e:xx]*;<FCS><CRLF
```

Ro

Name

Read digital input and output

Description

SAMSys readers have four optically isolated input lines and four transistor buffered output lines. The lines are unidirectional.

Refer to the reader owner's manual for deatiled information on the digital input/output lines.

This command reads the digital input/output port on the reader and returns a 2-digit HEX value (`d:xx`). The lower digit (second x) represents the last value written to the four digital output lines. The higher digit represents the current value of the four digital input lines.

Current flow through an input produces a logical 0 at the microprocessor which produces a logical 0 in the Ro reply. If the inputs are all open, the Ro reply is "`ff`". On powerup, the output lines are all set to 1. As a result, the Ro command returns all 1s prior to the first Wo command.

Type

Reader management command with reply

Format

```
}Ro;79<CRLF>
```

```
}Ro!<CRLF>
```

Return Message

```
{Ro,d:<data>;<FCS><CRLF>
```

Rt

Name

Read user data from specific tag

Description

Reads user data from a specific tag. If no "a:" sequence is included, the first block (block 00) is read. If no "l:" sequence is included, only one block is read. The reader does not translate block addresses.

Type

Tag command with reply

Format

```
}Rt,d:<tagID>[,a:<blk addr>][,l:<# blks>];
<FCS><CRLF>
```

```
}Rt,d:<tagID>[,l:<# blks>]!<CRLF>
```

where:

`a:<blk addr>` is the block address.

`l:<# blks>` is the number of blocks.

Return Message

```
{Rt,d:<data>;<FCS><CRLF>
```

```
{Rt[,e:xx]*;<FCS><CRLF>
```

Ru

Name

Read unselect for Protocol B

Description

Read unselect command with replies.

For Protocol B tags, the command initiates an anticollision sequence based on the configured Unselect Command, Unselect Mask, and Unselect Data.

If errors are waiting to be reported, one or more "`e:`" sequences can be returned.

Type

Command with reply

Format

`}Ru;7F<CRLF>`

Return Message

`{Ru,d:<tag id>;<FCS><CRLF>`

`{Ru[,e:xx]*;<FCS><CRLF>`

Rv

Name

Read version

Description

Returns reader software version as ASCII character string.

Type

Reader management command with reply

Format

```
}Rv;[f:<detail level>;<FCS><CRLF>
```

```
}Rv;[f:<detail level>!<CRLF>
```

where:

`f:0` returns all (version, copyright, serial no., date).

`f:1` drops the copyright.

`f:2` drops the serial number.

`f:3` drops the text in [...].

`f:4` drops the version and date.

Return Message

```
{Rv,d:<version><FCS><CRLF>
```

```
If f:0{Rv,d:”HFSARS_1.30[MB] Copyright (c)
Samsys Technologies Inc., s:000000000000”;
```

```
If f:1{Rv,d:”HFSARS_1.30[MB],
s:000000000000”;
```

```
If f:2{Rv,d:”HFSARS_1.30[MB]”;
```

```
If f:3{Rv,d:”HI469-13S 1.30”;
```

```
If f:4{Rv;<FCS>
```

S

Name

Request Resend of last serial message

Description

Requests reader to resend the last serial message. Ack reply is returned if no previous message is found.

Type

Flow control command with reply

Format

```
}S;0B><CRLF>
```

```
}S!<CRLF>
```

Return Message

```
{A;A7<CRLF>
```

```
(or) Last serial message
```

Tm

Name

Tag mute

Description

This is a tag specific command that sends an ISO 18000-6A SOF sequence to simulate a Tag Mute. This simulated Tag Mute leaves the reader in FST mode.

This command only applies to ISO 18000-6A tags.

Type

Tag command with reply

Format

```
}Tm;<79>;<CRLF>
```

```
}Tm!<CRLF>
```

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Tr

Name

Tag reset

Description

This is a protocol specific tag reset command.

Type

Tag command with reply

Format

```
}Tr;7E<CRLF>
```

```
}Tr!<CRLF>
```

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Ts

Name

Tag select

Description

For protocols that support tag selection, a Select Tag command is run. The SUID flag is set and SUID bytes are sent, regardless of the configuration write variable P6A settings.

Type

Tag command with reply

Format

```
}Ts,d:<tag ID>;<FCS><CRLF>
```

```
}Ts,d:<tag ID>!<CRLF>
```

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Tw

Name

Tag wakeup

Description

This is a tag specific command that runs an ISO 18000-6A Wake Up command. This command leaves the reader in FST mode.

This command only applies to ISO 18000-6A tags from manufacturers with proprietary extensions.

Type

Tag command with reply

Format

```
}Tw;83<CRLF>
```

```
}Tw!<CRLF>
```

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Wa

Name

Write user data to any tag in field

Description

This is a protocol specific tag command that writes user data to any tag in the field.

Type

Tag command with reply

Format

```
}Wa,b:<blk data>[,a:nn][,l:<# blks>];
<FCS><CRLF>
```

```
}Wa,b:<blk data>[,a:nn][,l:<# blks>]
!<CRLF>
```

where:

`a:<blk addr>` is the block address (0 by default).

`b: <blk data>` is the block data.

`l:<# blks>` is the number of blocks (1 by default).

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Wd

Name

Write DSFID

Description

This is a protocol specific tag command that writes the DSFID. If the "`d:`" sequence is not sent, the tag must first be selected.

Type

Tag command with reply

Format

```
}Wd,b:<blk data>[,d:<tag id>][,l:];
<FCS><CRLF>
```

```
}Wd,b:<blk data>[,l:] !<CRLF>
```

where:

`a:<blk addr>` is the block address (0 by default).

`b: <blk data>` is the block data.

`l:<# blks>` is the number of blocks (1 by default).

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Wf

Name

Write AFI

Description

For ISO 18000-6A tags, this command runs a Write AFI command. The SUID flag is set and the SUID bytes are sent. If the "`d:`" sequence is not sent, the tag must first be selected.

The reader remembers the AFI setting and sends the value with the subsequent Init_Round commands. The AFI value can be overwritten by performing a configuration write with variable P6A.

Type

Command

Format

`}Wf,b:xx[,d:<tag ID>];<FCS><CRLF>`

`}Wf,b:xx[,d:<tag ID>]!<CRLF>`

where:

`b:xx` is the AFI (one HEX byte, block data).

`d:<tag ID>` is the tag identification.

Return Message

`{A[,e:xx]*;<FCS><CRLF>`

Wo

Name

Write digital input and output

Description

SAMSys readers have four optically isolated input lines and four transistor buffered output lines. The lines are unidirectional. Writing a "1" to an output port is equivalent to energizing a relay or closing a switch.

The output ports require an external current source and pullups (one for each output line).

Refer to the reader owner's manual for deatiled information on the digital input/output lines.

This command accepts a 2-digit HEX value. The lower digit (second x) is written to the four output lines. The upper digit is ignored.

Type

Reader management command with reply

Format

`}Wo,d:<data>;<FCS><CRLF>`

`}Wo,d:<data>!<CRLF>`

where:

`d:<data>` is the model specific 8-bit digital data.

Return Message

`{A[,e:xx]*;<FCS><CRLF>`

Wt

Name

Write user data to a specific tag

Description

This command writes data to a specified block of a specified tag.

Type

Command with reply

Format

```
}Wt,d:<tag ID>[,a:nn],b:<blk data>
[l:<# blks>] [,t:<tag type>];<FCS><CRLF>
```

```
}Wt,d:<tag ID>[,a:nn],b:<blk data>
[l:<# blks>] [,t:<tag type>]!<CRLF>
```

where:

`d:<tag ID>` is the serial ID of the requested tag.

`a:nn` is the block address. Defaults to 0 if missing.

`b:<blk data>` is the block data.

`t:<tag type>` is the optional tag type (typically used to increase search speed).

`l: <# blks>` is the number of blocks (1 by default).

Return Message

```
{A[,e:xx]*;<FCS><CRLF>
```

Yl

Name

Query lock block status

Description

For protocols that support lock queries, this command reads the lock block status. If the "`d:`" sequence is not sent, the tag must first be selected.

Type

Tag command with reply

Format

```
}Yl[,d:<tag ID>];<FCS><CRLF>
```

```
}Yl[,d:<tag ID>]!<CRLF>
```

where:

`d:<tag ID>` is the serial ID of the requested tag.

Return Message

```
{Yl,f:<lock block info>;<FCS><CRLF>
```

where:

`f:<lock block info>` is the data for the lock status.

Ys

Name

Query tag system information

Description

For protocols that support tag system queries, this command runs a Get System Information command. If the ‘‘`d:`’’ sequence is not sent, the tag must first be selected.

Type

Tag command with reply

Format

`}Ys[,d:<tag ID>];<FCS><CRLF>`

`}Ys[,d:<tag ID>]!<CRLF>`

where:

`d:<tag ID>` is the serial ID of the requested tag.

Return Message

`{Ys[,d:<8 bit oid>]`

`[,bd:xx]` `DSFID` (if supported)

`[,bf:xx]` `AFI` (if supported)

`[,bs:xx] block size`

`[,bn:xx] number of blocks`

`;<FCS><CRLF>`

`{Ys[,e:xx]*;<FCS><CRLF>`

Chapter 3

Reader Configuration

This chapter provides information on configuring a SAMSys reader. Topics discussed in this chapter include the following:

- Reader Configuration Overview
- Default Reader Configuration
- Configuration Word Descriptions
  - General Configuration Word (GCW)
  - Serial Configuration Word (SCW)
  - Serial Multidrop Address (SMA)
  - Multiplexer Configuration Word (MCW)
  - Multiplexer Select Word (MSW)
  - Protocol Configuration Word (PCW)
  - Protocol Select Word (PSW)
  - Protocol Selection Mask (PSM)
- Updating Reader Firmware

Reader Configuration Overview

SAMSys reader software is easily configured for different operating modes and protocols. Typical reader functions that can be configured include the following:

- CHUMP programming mode
- RF communication mode (polled or continuous)
- Serial communication mode (polled or continuous)
- Serial multidrop addressing
- Protocol configuration
- Protocol selection

To setup a SAMSys reader, the Configuration Write (Cw) command is used to write a three-letter configuration variable name and a 32-bit word to the reader. This word contains the individual setup parameters required by the reader.

Commonly used configuration variables include the following:

- GCW – General Configuration Word
- SCW – Serial Configuration Word (CHUMP 1.31 and above)
- SMA – Serial Multidrop Address
- MCW – Multiplexer Configuration Word (UHF reader)
- MSW – Multiplexer Select Word (MSW)
- PCW – Protocol Configuration Word
- PSW – Protocol Select Word
- Pxx – Specific Protocol Configurations

Configuration variables are described in the following section.

For specific syntax information on the Configuration Write command refer to Chapter 2.

Default Reader Configuration

SAMSys proximity readers are factory set with the following configurations:

- RF Continuous
- Serial Continuous
- 9600 Baud
- 8 Data Bits
- 1 Stop Bit
- No Parity
- All supported protocols enabled

The MP9320 UHF Long-Range reader is factory set with the following additional parameters:

- 1 antenna (antenna port 1)
- 1 inventory round per antenna

Configuration Word Descriptions

Warning

The SAMSys MP9320 UHF Reader uses internal registers to provide a high degree of flexibility in setting operational parameters. The use of these registers is described in the following sections. However, improper setting of these registers can result in reader inoperability or permanent damage.

In particular, the TPC, TP0, TP1, TP2 and TP3 registers control the transmit power of the reader. The user should not modify these registers unless specifically directed by SAMSys personnel. Trained, professional installers should only modify the registers as part of the transmit calibration procedure described in the Field Installation Guide.

General Configuration Word (GCW)

The GCW is used to set the reader operation to continuous or polled and to set the serial communication mode. For UHF readers, the FCC or ETSI mode can also be set.

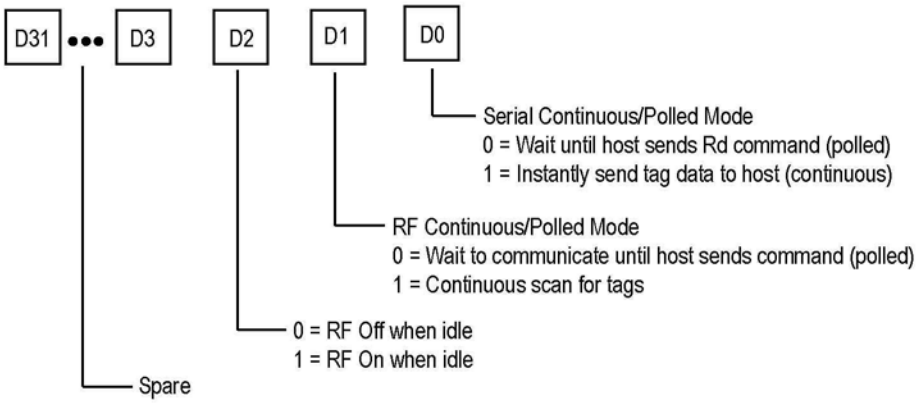


Figure 1 - *General Configuration Word*

NOTE: *The "RF On when idle" bit can be set in any mode to maintain power to the antenna and any tags in the antenna field.*

Serial Configuration Word (SCW)

SAMSys readers have up to three UARTs to control serial communications. The SCW sets the baud rate, word length, parity, and stop bits for each UART (see following note).

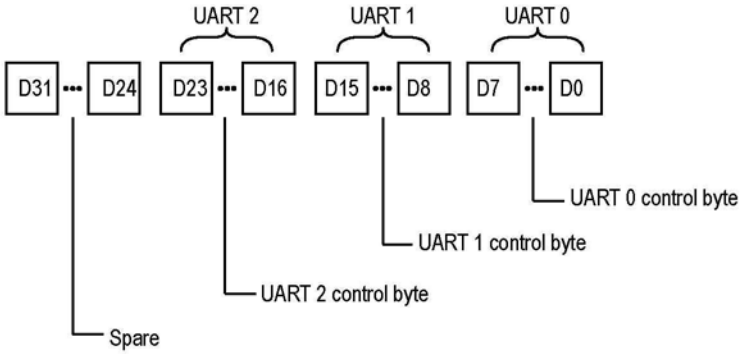


Figure 2 - *Serial Configuration Word (Bytes)*

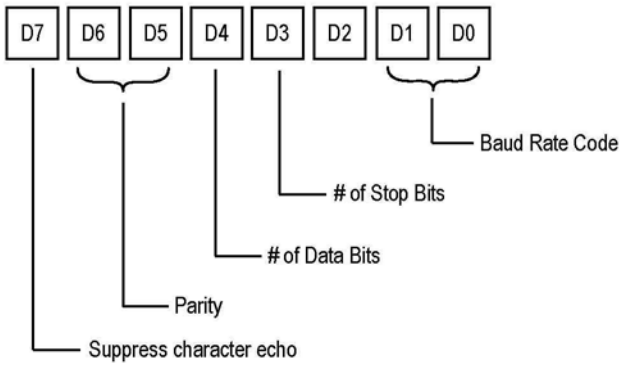


Figure 3 - *Serial Configuration Word (Bits for UART 0)*

Correct method: `}cw,d:scw,b:00131313!`

Incorrect method: `}cw,d:scw,b:13!`

NOTE: *.Parameters for UARTS 0, 1, and 2 must programmed concurrently (see following example commands). Trying to configure only one UART will make the other UARTS unusable. However, each UART can be set to different speeds. If the reader becomes unresponsive over the RS-232 or RS-485 links, refer to the following procedure to "unlock" the communication link.*

1. To "unlock" the communication link, perform the following:
2. Configure a terminal to 9600 Baud, No parity, 8 data bits, and 1 stop bit.
3. Connect the terminal to the reader.
4. Power up the reader.
5. While the reader is in the power-up sequence, hit the Enter key more than three (3) times.
6. The reader will revert to the SCW default setting of 00131313.

The following table shows the bit values for each UART.

Bit(s)	Parameter	Bit Value	Description
7	Suppress echo	1	Suppress character echo
6,5	Parity	00	None
		01	Odd
		10	Even
4	Word length	0	7 bits
		1	8 bits
3	Stop bits	0	1 stop bit
		1	2 stop bits
2...0	Baud rate	000	1200 bps
		001	2400 bps
		010	4800 bps
		011	9600 bps (default)
		100	19.2 Kbps
		101	38.4 Kbps
		110	57.6 Kbps
		111	115.2 Kbps

The following examples are commonly used bit settings:

00010011 = 0x13 for 9600, 8 data bits, No parity, 1 stop bit

00010101 = 0x15 for 38400, 8 bits, No parity, 1 stop bit

Serial Multidrop Address (SMA)

The SMA assigns the multidrop address to the reader. Valid addresses are any hexadecimal value between 01 - FE.

NOTE: *The address value "00" is used as the multidrop broadcast address and should not be programmed into the reader. The value "FF" is used to set the reader for single drop operation.*

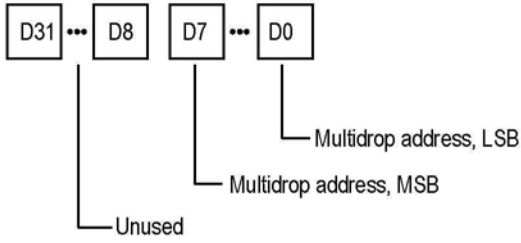


Figure 4 - *Serial Multidrop Address*

Multiplexer Configuration Word (MCW)

The MCW configures the MP9320 UHF reader. Parameters that can be set include the following:

- Enable/suppress antenna hopping
- Number of antennas
- Number of inventory operations before antenna hop
- Enable/suppress MUX failure messages

The bit settings for the MCW are shown in Figure 5.

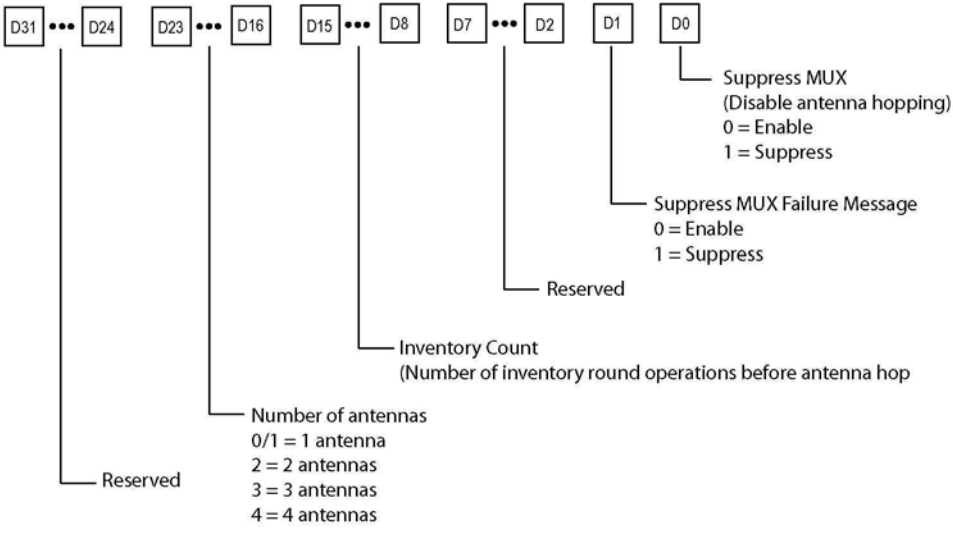


Figure 5 - *MCW Bit Configuration*

The following example command selects four antennas with one Inventory Read Round per antenna:

```
}cw,d:mcw,b:40100,f:1!
```

Multiplexer Select Word (MSW)

The MSW selects which antennas are active on the MP9320 UHF reader. The variable also selects the activation order (LSB to MSB). The bit settings for the MSW are shown in Figure 5.

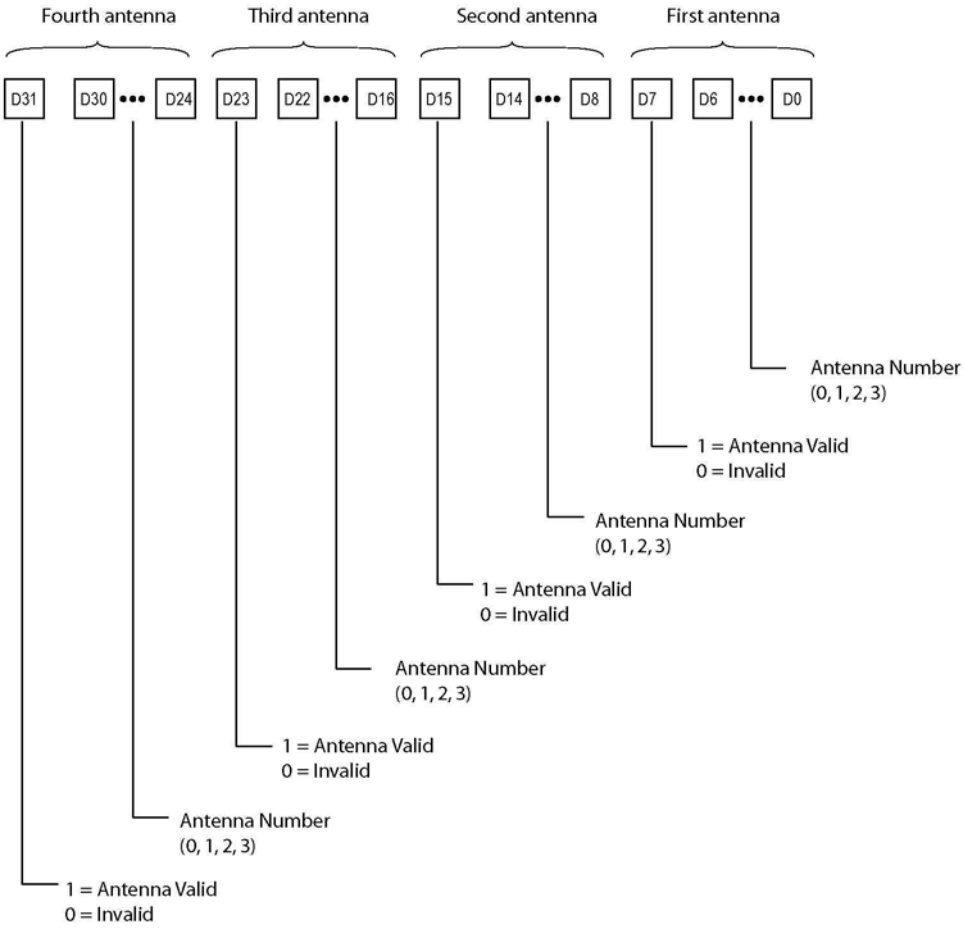


Figure 6 - *MSW Bit Configuration*

The following example commands illustrate how the MCW and MSW registers can be used to configure different antenna combinations:

Select antenna 1 only:

```
}cw,d:mcw,b:00010200,f:1!
}cw,d:msw,b:80!
```

Select antenna 2 only:

```
}cw,d:mcw,b:00010200,f:1!
}cw,d:msw,b:81!
```

Select antenna 2 and 3:

```
}cw,d:mcw,b:00020200,f:1!
}cw,d:msw,b:00008281!
```

Select all 4 antennas:

```
}cw,d:mcw,b:00040200,f:1!
}cw,d:msw,b:83828180!
```

Use all 4 antennas in reverse order:

```
}cw,d:mcw,b:00040200,f:1!
}cw,d:msw,b:80818283!
```

NOTE: *Adding "f:1" to the end of the command makes the setting permanent. Remove "f:1" for temporary command settings.*

Protocol Configuration Word (PCW)

The PCW controls the multiprotocol behavior of the reader.

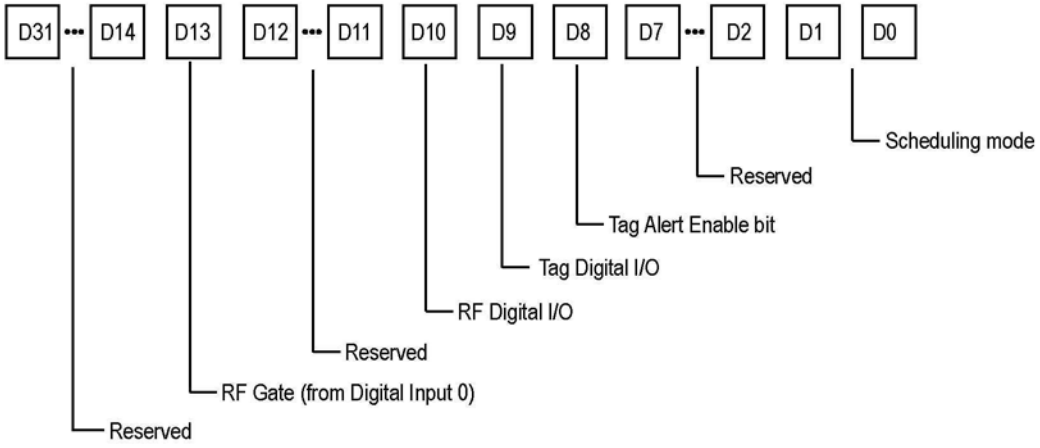


Figure 7 - *Protocol Configuration Word*

A value of "00" (default) sets the reader to Round Robin mode and suppresses the beep when tags are detected.

Tag Alert enables the buzzer (horn) on tag activity.

Tag Digital I/O enables DIO_0 (OC_OUT0) on tag activity.

RF Digital I/O enables DIO_1 (OC_OUT1) to indicate RF_ON/ RF_OFF.

RF Gate uses Digital Input 0 to gate Auto mode Inventory operations.

Protocol Select Word (PSW)

The PSW selects the protocols to be scanned by the reader. The data word is a bitmap of the tag types to be scanned in RF Continuous mode and in response to serial commands.

To select a tag type, set the appropriate bit to "1". The bit assignments for the various tag types are shown in the following table. The name is the tag type name returned by the reader after executing the "Rd" command.

Tag Type	Name	Bit #	Freq
EMM 4102	T_EM	0	125 KHz
EMM 4025 32cpb	EM432	1	125 KHz
Philips HiTag1	PHT1	2	125 KHz
Texas Instruments TIRIS, Single-Part, Read Only	TRSR	3	134 KHz
Texas Instruments TIRIS, Single-Part, Writable	TRSW	3	134 KHz
Texas Instruments TIRIS, Multi-Part	TRSM	3	134 KHz
Temic (Atmel) e555x	T_EM	4	125 KHz
Texas Instruments TagIt	TGT	5	13.56 MHz
ISO 15693	IS15	6	13.56 MHz
Philips 13.56 I-CODE	ICP	8	13.56 MHz
Philips EPC	IEPC	9	13.56 MHz
ISO 18000-6A	IS186A	10	UHF
ISO 18000-6B, Intermec Intellitag®,	IS186B	11	UHF
EM 4022, EM 4222	STG	12	UHF
Alien EPC	EPC1	13	UHF
Matrics EPC	EPC2	14	UHF
Other or Unknown	unk	—	

NOTE: *I-CODE in this manual refers to the Philips 13.56 MHz protocol and is only applicable to the MP9210 13.56 MHz reader. To read Philips I-CODE HSL on the MP9320 UHF reader, configure the reader for ISO 18000-6B.*

Protocol Selection Mask (PSM)

The PSM configuration variable determines the protocols supported by the reader. The bit settings for the PSM are the same as those in the Protocol Select Word (PSW) and are set at the factory.

The PSM settings are locked by a protection key based on the Electronic Serial Number (ESN) of the reader. The ESN is a software encoded number and is not the serial number found on the reader enclosure. When a new protocol is purchased for the reader, a new PSM mask and protection key are provided.

If a new protocol is to added, provide the ESN to SAMSys at the time of purchase. The ESN can be obtained by executing the **Rv** command. As shown in the following example, the ESN is the **s:** parameter returned by the command.

```
}Rv!
```

```
{Rv,d:"EPCReader_1.34.14+ 224[EPC1pBpAEM]
Copyright (c) 2003 SAMSys Technologies
Inc.",s:E9000008333B5501;AA
```

After receiving a new mask and key, load the values into the reader using the **Cw** command. As shown in the following example, the new mask is the value of the **b:** parameter and the key is the value of the **s:** parameter.

```
}Cw,d:PSM, b:0x00000015,s:3047FE91,f:1!
```

If the update is valid, no error codes will be generated during the acknowledgement. An invalid update will return an error code of 45 in the acknowledgement.

NOTE: *Updating the Protocol Selection Mask (PSM) does not automatically update the Protocol Select Word (PSW). After updating the PSM, the PSW musr be updated with a new value. Refer to the previous section for information on updating the PSW.*

Specific Protocol Configurations

Many individual tag protocols can be configured for specific behaviors. These behaviors and functions include the following:

- Auto-quiet, quiet tag after detection
- Get and display tag size information
- Order of bytes in tag data
- Order of bits in tag data bytes
- Number of timeslots to use

Refer to Chapter 4 for detailed configuration word bit settings for specific protocols and tag types.

Updating Reader Firmware from a Terminal

Periodically, SAMSys releases updated reader firmware that incorporates new protocols, enhanced features, and updated FlashROM images. SAMSys readers can be automatically updated with new firmware using a terminal emulation program such as HyperTerminal®. The firmware file is typically included in a zipped release file such as "UHFReader_1.34.00.zip".

To configure HyperTerminal, perform the following:

7. Select:
   File ⇨ Properties ⇨ Connect to ⇨ Configure

8. Set to the following parameters:
   - Baud rate = 9600
   - Data bits = 8
   - Parity = None
   - Stop bits = 1
   - Flow Control = None

9. Select:
   File ⇨ Properties ⇨ Settings ⇨ ASCII Setup

10. Set Line delay to 100 milliseconds

11. Initiate CHUMPloader. Enter:
    }cu!
    }cu!

12. Start the firmware upload. Select:
    Transfer ⇨ Send Text File...

13. Browse and select the updated firmware file "*******.chp".

14. The upload will take approximately 3 minutes. After upload, reboot the reader. Enter:
    }cx!

Updating Reader Firmware with the ICCM

In addition to a terminal emulation program, SAMSys readers can be automatically updated with new firmware using the Interrogator Control and Concentrator Module (ICCM).

To upgrade the reader firmware using the ICCM, verify that the ICCM is connected to the reader via the RS-485 communication port and that a PC or other host computer is connected to the ICCM via the RS-232 port. See Figure 8.

NOTE: *If you do not have an ICCM, contact SAMSys for additional firmware upload instructions.*

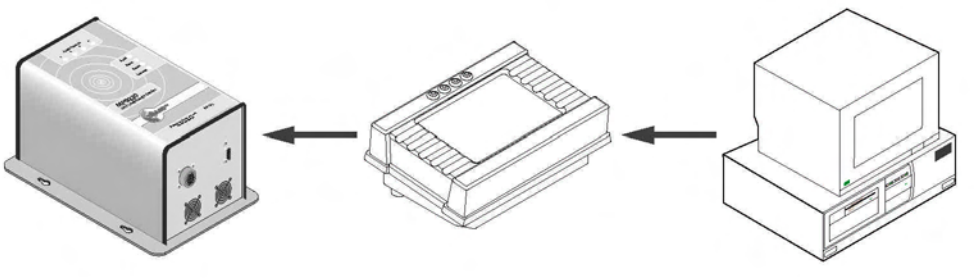


Figure 8–*Firmware Upload System Configuration*

Receiving Updated Firmware from SAMSys

Updated MP9320 reader firmware is available on CD or via download from the SAMSys website at www.samsys.com. The upgrade files include the reader application code which has a **`.mot`** extension and a FlashROM image file which has a **`.mcs`** extension.

The following files are example upgrade files for an MP9320 reader. Your files may have different names, but the extensions will be the same.

`UHFrdr_1.30d.mot`

`UHFrdr_r2.mcs`

Copy the files to a PC or other host computer connected to the ICCM.

Transferring the Files to the ICCM

Once the upgrade files are loaded to your PC, use the **Transfer Files** function on the ICCM to transfer the files to the ICCM. For detailed information on the Transfer Files function, refer to the *Interrogator Control and Concentrator Module User's Guide*.

Uploading the File to the Reader

After transferring the files to the ICCM, use the **Upload Reader Firmware** function to upload the reader application file to the reader. Use the **Upload Reader FlashROM** function to upload the Forth word definitions file to the reader. For detailed information on the Upload Reader functions, refer to the *Interrogator Control and Concentrator Module User's Guide*.

Chapter 4

Tag Configuration

This chapter provides configuration variables and detailed configuration word bit settings for specific tag types and protocols. Protocols that are covered include the following:

- HITAG 1 (PHT)
- I-Code (PPI)
- I-Code EPC (PEP)
- Tag-it™ (PTA)
- TEMIC (PET)
- TIRIS (PTI)
- ISO 15693 (P15)
- ISO 18000-6A (P6A)
- ISO 18000-6B, Intellitag®
- EPC

Philips HITAG 1 Protocol Configuration (PHT)

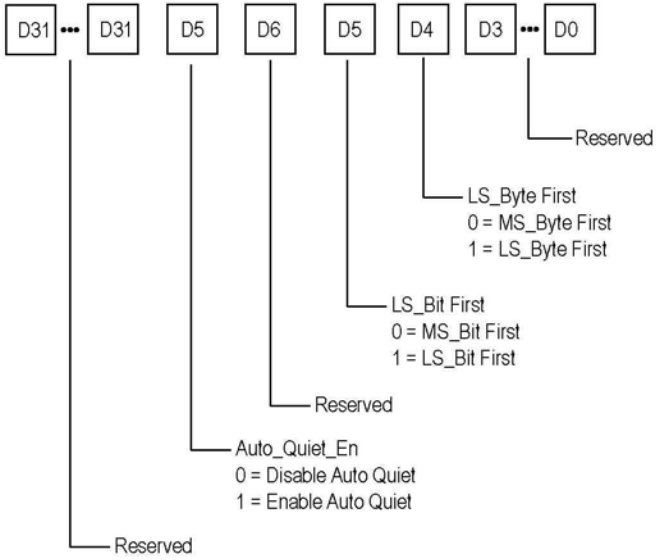


Figure 9 - *HITAG 1 Protocol Configuration*

Philips I-Code (HF) Protocol Configuration (PPI)

The default value for this configuration variable is “0000013”.

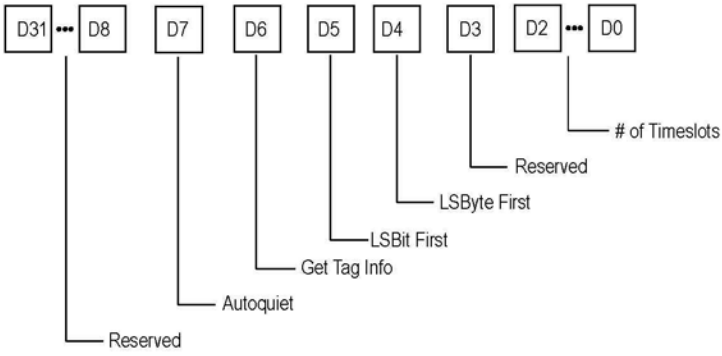


Figure 10 - *Philips I-Code Protocol Configuration*

Philips I-Code EPC (HF) Protocol Configuration (PEP)

The default value for this configuration variable is “00000004”.

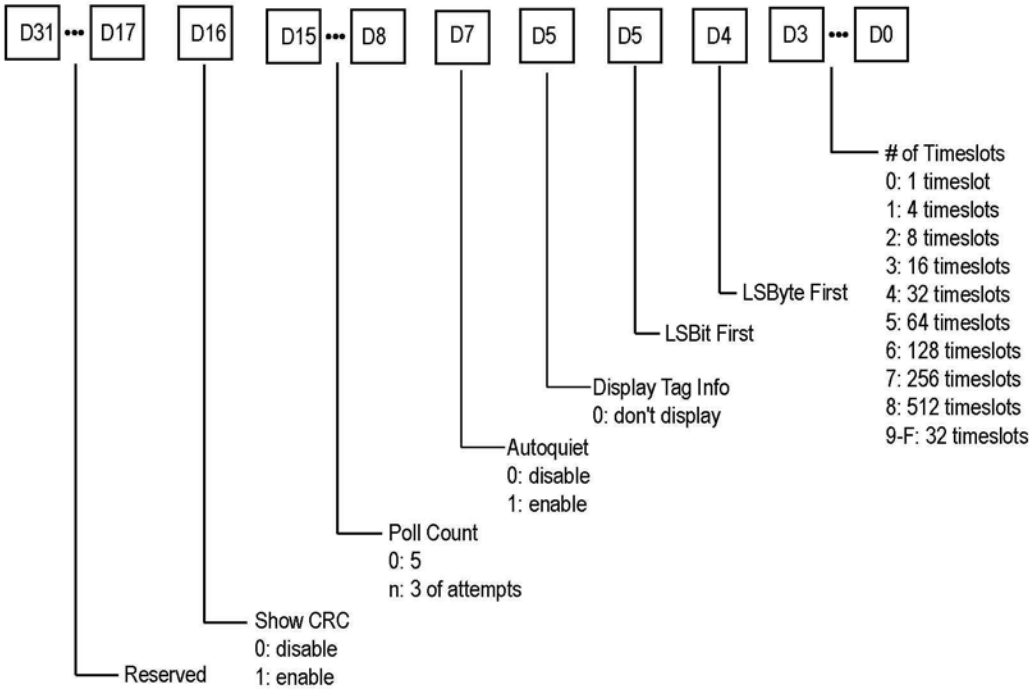


Figure 11 - *Philips I-Code EPC Protocol Configuration*

Typical PEP modes are as follows:

- Fast 1 tag reporting: 0x00000080
- Many tags w/ Auto-Quiet: 0x00000684
- Many tags w/ AQ and debug: 0x00001684

TI Tag-it™ Protocol Configuration (PTA)

The default value for this configuration variable is “0000040”.

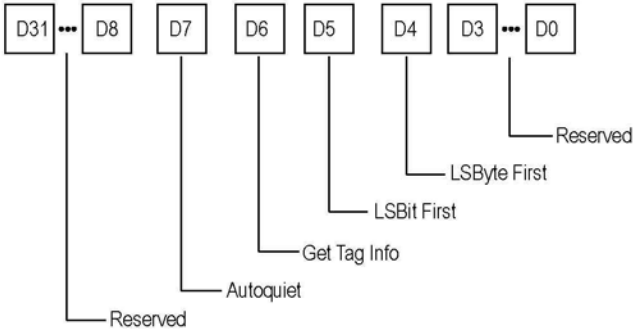


Figure 12 - *TI Tag-it™ Protocol Configuration*

Atmel TEMIC Protocol Configuration (PET)

The default value for this configuration variable is “0000004”.

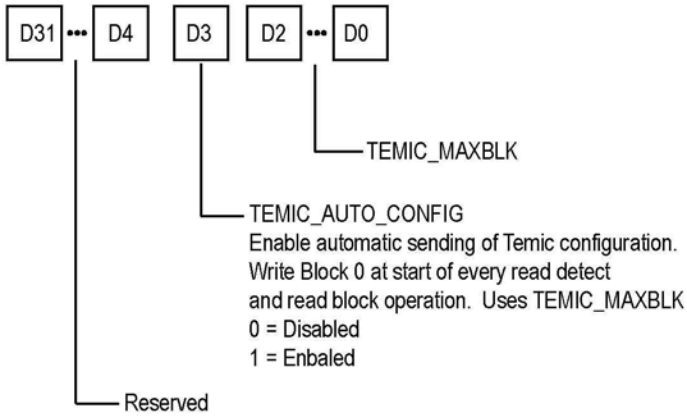


Figure 13 - *TEMIC Protocol Configuration*

TI TIRIS Protocol Configuration (PTI)

The default value for this configuration variable is “0000003”.

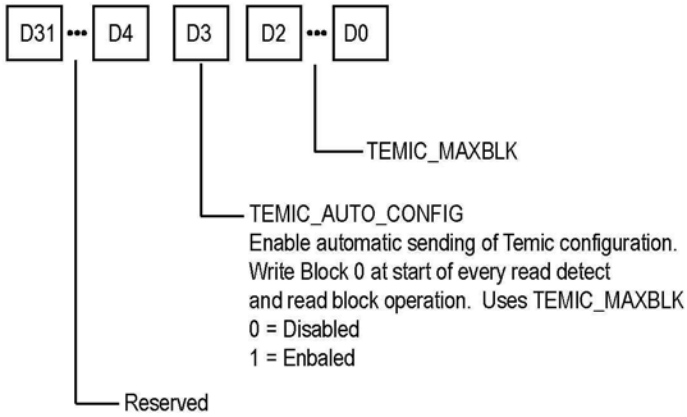


Figure 14 - *TIRIS Protocol Configuration*

ISO 15693 Protocol Configuration (P15)

The default value for this configuration variable is “0000041”.

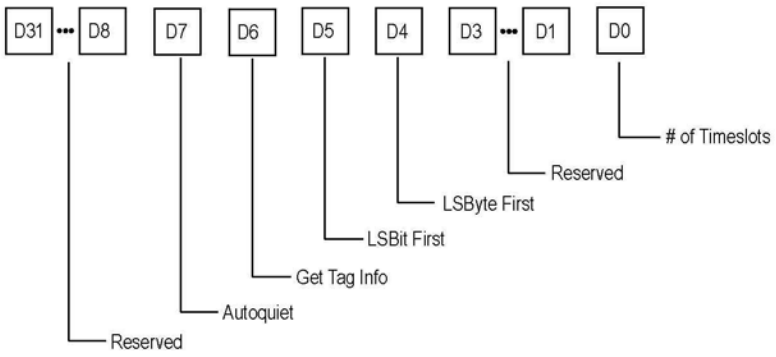


Figure 15 - *ISO 15693 Protocol Configuration*

ISO 18000-6A Protocol Configuration (P6A)

The default value for this configuration variable is “00001024”. The AFI is “00”, 16 slots, and SUID set on inventory commands.

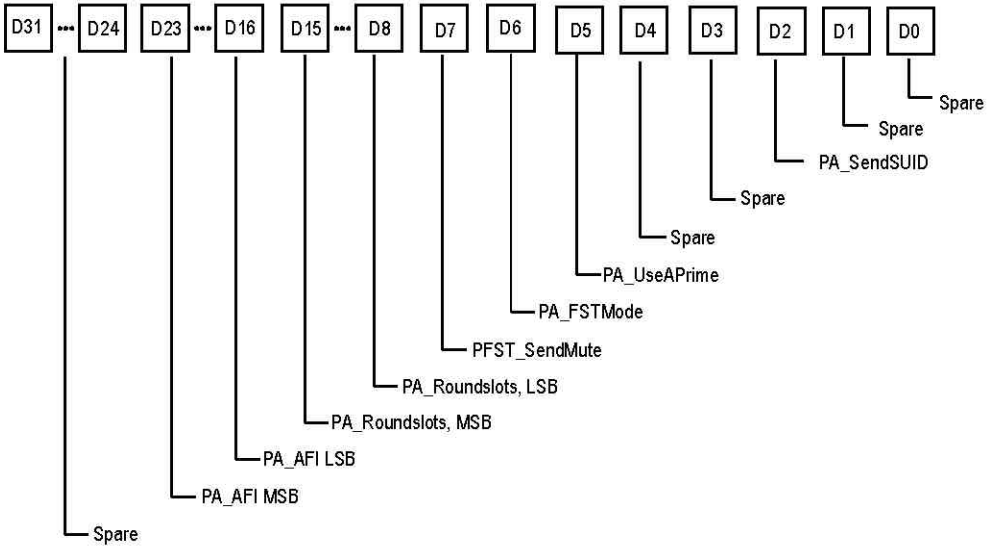


Figure 16 - *ISO 18000-6A Protocol Configuration*

ISO 18000-6B and Intellitag® Configuration

The ISO 18000-6B and Intellitag®protocol is highly versatile. As a result, SAMSys readers can be programmed with several ISO 18000-6B variables to support the different configurations.

ISO 18000-6B Selection Configuration Word (P6S)

The default value for ISO 18000-6B is “00010017”.

For Intellitag® , P6S must be set to “00”. This sets GROUP_SELECT_EQ_FLAGS Data Exchange Status Bit (DE_SB) equal to zero.

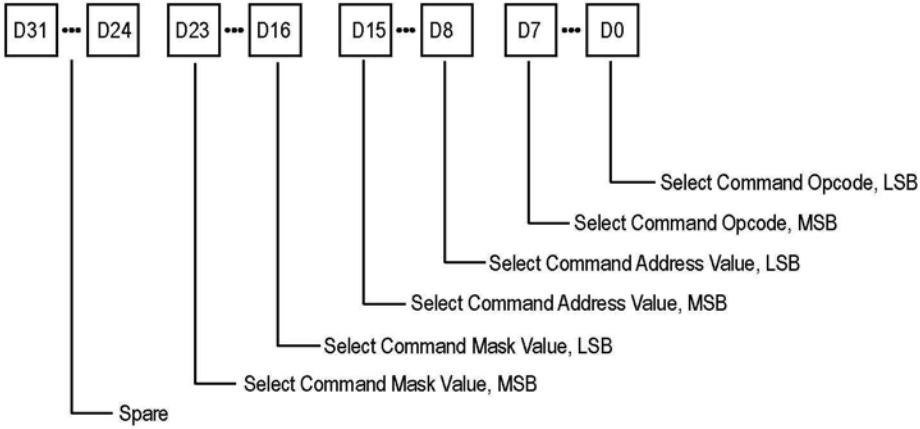


Figure 17 - *ISO 18000-6B Selection Configuration*

IS0 18000-6B Selection Data High (P6H)

The default value for this configuration is “00”. All data fields are set to “0” to select all tags in the RF field.

IS0 18000-6B Selection Data Low (P6L)

The default value for this configuration is “00”. All data fields are set to “0” to select all tags in the RF field.

ISO 18000-6B Unselection Configuration Word (P6U)

The default value for this configuration is “00”. The unselect command is GROUP_UNSELECT_EQ with all data fields set to “0” to deselect all tags in the RF field.

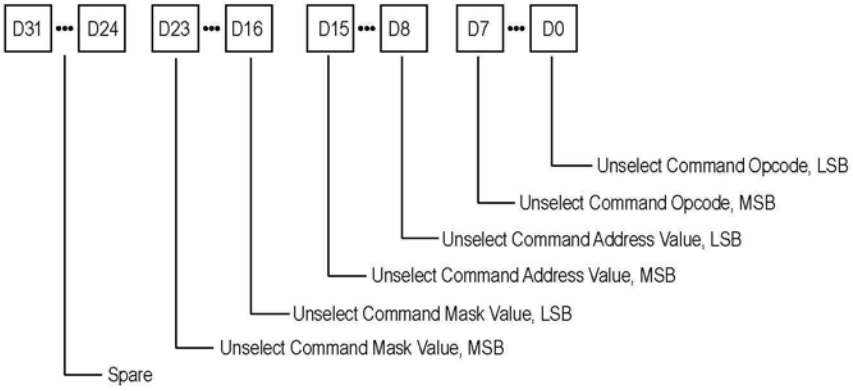


Figure 18 - *ISO 18000-6B Unselection Configuration*

IS0 18000-6B Unselection Data High (P6H)

The default value for this configuration is “00”. The unselect command is GROUP_UNSELECT_EQ with all data fields set to “0” to deselect all tags in the RF field.

IS0 18000-6B Unselection Data Low (P6L)

The default value for this configuration is “00”. The unselect command is GROUP_UNSELECT_EQ with all data fields set to “0” to deselect all tags in the RF field.

EPC Configuration

The following variables are used to configure the reader to support different EPC configurations.

UCODE V1.19 Enable 4X Return Link Word (P6B)

The default value is “00” which sets the reader to the 40000 bps return link. To enable UCODE V1.19 four (4) times return link (160000 bps), set the register to “01”. To return to the default return link of 40000 bps, set the register to “00”.

This configuration variable applies to the ISO 18000-6B and the Philips UCODE tag.

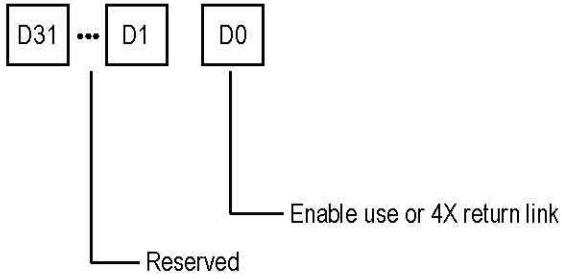


Figure 19 - *ISO 18000-6B Enable 4XReturn Link*

NOTE: *This register only applies to the MP9320 EPC reader.*

EPC Bit Length Configuration Word (PEC)

Default value is 0x0000. A setting of 0x200 will set the reader for EPC 0/1 = 64 bits and optimize for large populations.

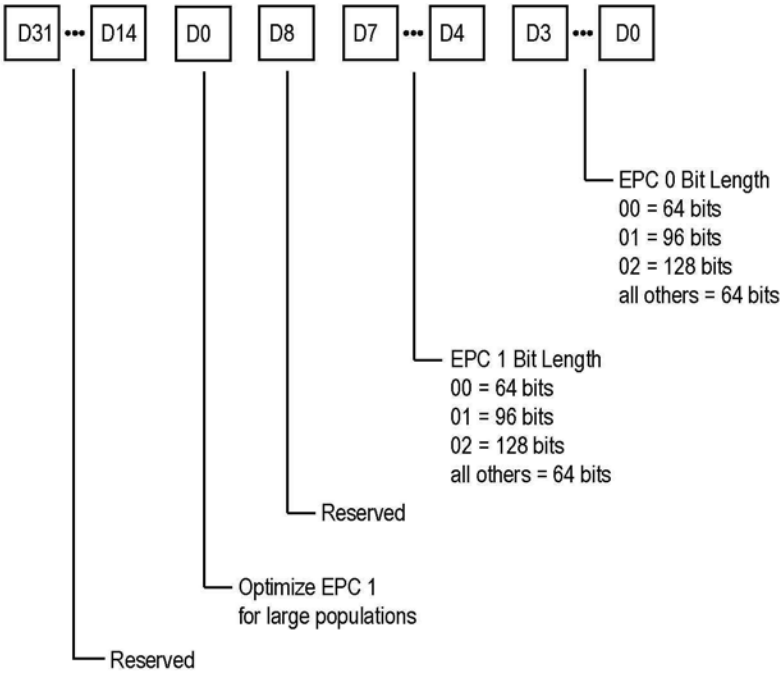


Figure 20 - *EPC Bit Length Configuration Word (PEC)*

Appendix A

Error Codes

This chapter provides descriptions of the error codes returned by the reader. Error codes described in this chapter include the following:

- Serial Protocol Errors
- Board Support Errors
- Radio and Tag Errors
- Application Errors

Serial Protocol Errors

Error Code	Description
0x01	Bad message: point-to-point only. Problem with message head or tail sequence.
0x03	Message Synch: point-to-point only. Current message head or “}” character interrupted a previous message.
0x04	Bad CRC: point-to-point only. The last valid message string had a bad CRC.
0x05	Unprintable bytes rejected.
0x08	Bad message head.
0x09	Bad message tail.
0x0a	Bad or unsupported command.
0x0b	Bad or unsupported subcommand.
0x0c	Unsupported command parameter. Command rejected.
0x0d	Unsupported command parameter. Command proceeding.
0x0e	High level serial error 1
0x0f	High level serial error 2

Board Support Errors

Error Code	Description
0x21	Temperature sensor error.
0x22	RAM error.
0x23	ROM error.
0x24	Serial number chip error.
0x25	Antenna missing or bad
0x26	Bad antenna mux
0x27	RXD reflective power too high. Check antenna or cable connection.
0x2f	Other self-test error.

Radio and Tag Errors

Error Code	Description
0x31	Radio controller does not respond or general RF section failure.
0x33	Antenna failure.
0x35	No RF field.
0x36	No tag in field (Rd, Rt, Wa, Wt commands).
0x37	Tag data received, but fails integrity checks (Rd, Rt, Wa, Wt commands).
0x38	Good tag data available, but specific requested tag not found (Rt, Wt commands).
0x39	Attempt to write to a read-only tag or locked block.
0x3a	Radio lost lock (UHF synthesizer error).

Application Errors

Error Code	Description
0x41	Tag data lost (in serial polled or RF continuous mode, incoming new tag data has overwritten old tag data between polls).
0x42	Non-volatile memory data element does not exist or not found [when requesting a read of a specific element in Non-volatile memory (NVM)].
0x43	NVM is full (when requesting a write to a specific element in NVM).
0x44	No NVM present
0x45	Invalid configuration variable
0x46	Failure of application specific hardware

Appendix B

External Interfaces

This chapter provides information and software routines to aid users and system integrators developing external applications that interface to SAMSys RFID readers.

- Sample Checksum Generation Code (in C)
- Sample Checksum Verification Code (in C)
- Sample Checksum Code (in Visual Basic)

Sample Checksum Generation Code

In order for an external device to communicate with a SAMSys reader, a CHUMP command must be sent to the reader followed by a checksum. The following code sample is a “C” routine to generate the checksum.

```
char compute_checksum(const char* pData, int nLen
{
   char returnValue = 0;
   for (int i=0; i<nLen; i++)  {
      returnValue += pData[i];
   }
   return returnValue;
}
```

Sample Checksum Verification Code

The following code sample is a “C” routine to verify a checksum.

```
BOOL checksum_verify(char const* buffer, int nLen)
{
   char sentCRC = pullHexValue(buffer+nLen-4);
   char cs = compute_checksum(buffer, nLen-4);
   return (cs == sentCRC);
}
```

Sample Checksum Code (Visual Basic)

The following code sample is a Visual Basic routine to generate a checksum.

```
Dim checksum As Long
Dim chumpedString As String

checksum = ASC(“}”)
checksum = checksum + ASC(“R”)
checksum = checksum + ASC(“v”)
checksum = checksum + ASC(“,”)
checksum = checksum + ASC(“f”)
checksum = checksum + ASC(“:”)
checksum = checksum + ASC(“1”)
chumpedString = “}Rv,f:1”
checksum = checksum + Asc (“1”)
chumpedString = chumpedString + “;” +
                Right(Hex(checksum), 2)
```

Appendix C

ISO 18000 Data Elements

This appendix provides information on specific data elements used within the ISO 18000 protocol.

- Unique Identifier (UID)
- Sub-Unique Identifier (SUID)
- Application Family Identifier (AFI)
- Data Storage Format Identifier (DSFID)

Unique Identifier (UID)

ISO 18000 tags are identified by a 64-bit Unique Identifier that is used for tag addressing. As shown in Figure 22, the UID consist of the following:

- 8 MSBs defined as “EO”
- 8-bit manufacturer code
- 48-bit unique serial number

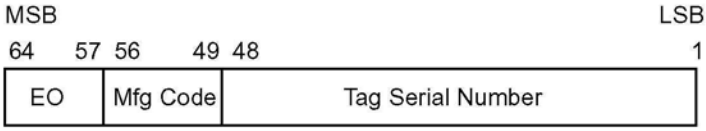


Figure 21 - *Unique Identifier Format*

Sub-Unique Identifier (SUID)

Within the ISO 18000-6A protocol, only part of the 64-bit Unique Identifier (UID) is used. Referrred to as the Sub-Unique Identifier (SUID), this 40-bit code is transmitted in most tag commands and in tag replies during collision arbitraton. The SUID is made up of an 8-bit manufacturer code followed by the 32 LSBs of the manufacturer serial number.

NOTE: *The manufacturer serial number in the 64-bit UID is typically 48 bits. However, the 16 MSBs (33-48) are set to 0.*

Figure 22 shows the mapping of the 64-bit UID to the 40-Bit SUID and back to the UID.

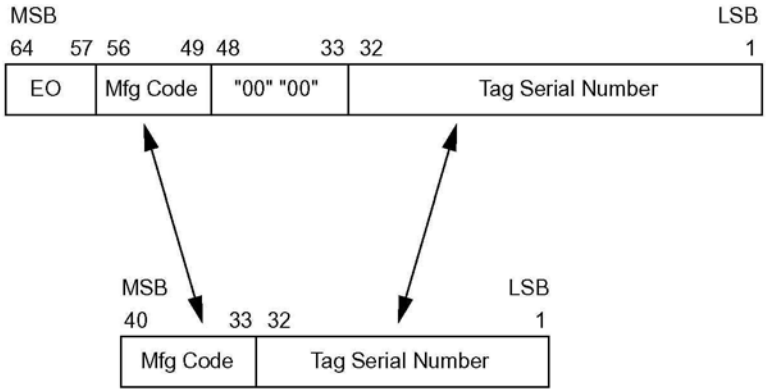


Figure 22 - *Sub-Unique Identifier Mapping*

Application Family Identifier (AFI)

Tag application families are identified by a 1-byte identifier. This byte consist of a 4-bit application family identifier and a 4-bit application sub-family (ASF) identifier.

The AFI identifies the type of application targeted by the reader and allows the reader to target only those tags meeting the required application criteria.

For more information on the AFI, refer to the ISO/IEC CD 18000-6 specification.

Data Storage Format Identifier (DSFID)

The DSFID is a 1-byte code that indicates how data is structured in tag memory. For more information on coding the DSFID, refer to the ISO/IEC 15962 and ISO/IEC specification.

Appendix D

Glossary of Terms

This appendix provides definitions of commonly used acronyms, abbreviations, and terms.

ABM	Asynchronous Balanced Mode (HDLC mode)
ACK	Acknowledge
AFI	Application Family Identifier (family code)
ARM	Asynchronous Response Mode (HDLC mode)
ASCII	American Standard Code for Information Exchange
ASF	Application Sub-Family identifier
Bps	Bits per second
BIT	Built-In test
chksm	Check Sum byte in CHUMP message tail
CHUMP	Comprehensivie Heuristic Unified Messaging Protocol
Cpb	Cycles per bit
CR	Carriage Return
CRC	Cyclic Redundancy Check
DBP	Differential BiPhase
DSFID	Data Storage Format Identifier
dst	Destination
ETSI	European Telecommunications Standards Institute
FCC	Federal Communications Commission
FCS	Frame Check Sum
FST	Fast SuperTag

HDLC	High-level Data Link Control
HF	High Frequency (in RFID, 13.56 MHz)
ICCM	Interrogator Control and Concentrator Module
I/O	Input/Output
LED	Light-Emitting Diode
LF	Line Feed
LF	Low Frequency (in RFID, 125 or 134 KHz)
LSB	Least Significant Bit
MSB	Most Significant Bit
Multidrop	Serial communication link with multiple devices sharing a common wired connection (typically RS-485).
NACK	Negative Acknowledge
NRM	Normal Response Mode (HDLC mode)
NVM	Non-Volatile Memory
OSI	Open Systems Interface
POR	Power On Reset
RF	Radio Frequency
RF Continuous	Reader automatically scans for tags in RF field
RFID	Radio Frequency Identification
RF Polled	Reader only scans for tags in RF field on command
SARM	Stand-Alone Reader Multiple frequency

Serial Continuous	Reader only sends tag data to host on command
Serial Polled	Reader only sends tag data to host on command
SOAP	Simple Object Access Protocol
SOF	Start of frame
src	Source
SUID	Sub-Unique Identifier
tag	RFID transponder (can be active or passive)
UART	Universal Asynchronous Receiver-Transmitter
UHF	Ultra High Frequency (in RFID, 869.55 MHz under ETSI rules and 902-928 MHz under FCC rules
UID	Unique Identifier

SAMSys Technologies, Inc.

Part Number: HI471-CH-RF-V3.2
Printed in U.S.A.