EPSON
EXCEED YOUR VISION

S1S60000
Application Note

No.4 How To Use the TCP/IP Communication End Points

SEIKO EPSON CORPORATION

Rev.1.2

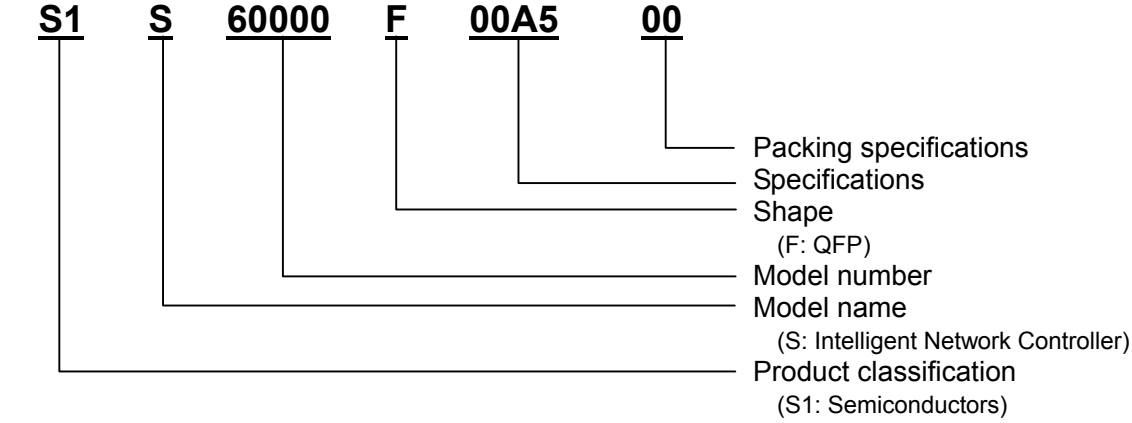
NOTICE

No part of this material may be reproduced or duplicated in any form or by any means without the written permission of Seiko Epson. Seiko Epson reserves the right to make changes to this material without notice. Seiko Epson does not assume any liability of any kind arising out of any inaccuracies contained in this material or due to its application or use in any product or circuit and, further, there is no representation that this material is applicable to products requiring high level reliability, such as, medical products. Moreover, no license to any intellectual property rights is granted by implication or otherwise, and there is no representation or warranty that anything made in accordance with this material will be free from any patent or copyright infringement of a third party. This material or portions thereof may contain technology or the subject relating to strategic products under the control of the Foreign Exchange and Foreign Trade Law of Japan and may require an export license from the Ministry of International Trade and Industry or other approval from another government agency.

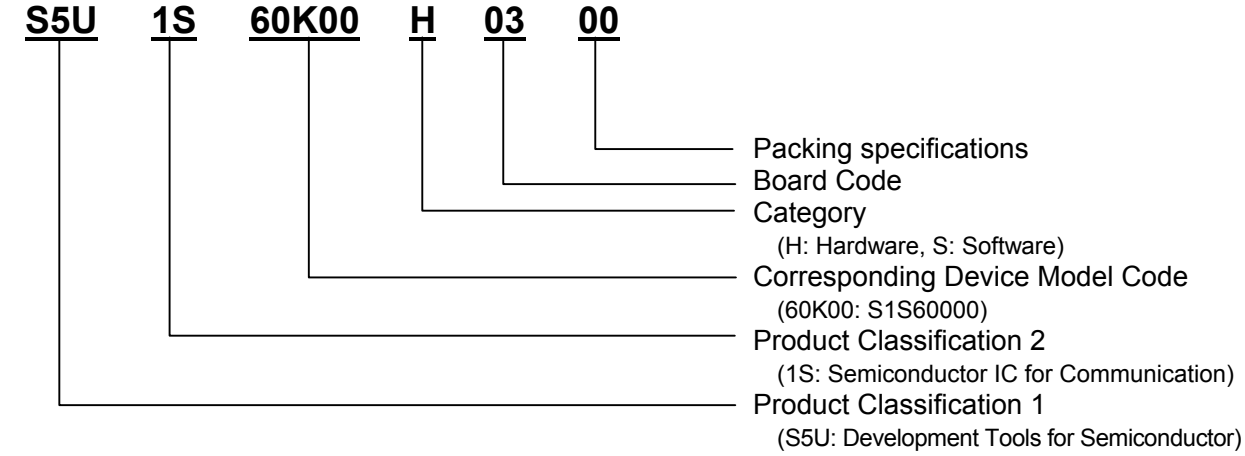
All other product names mentioned herein are trademarks and/or registered trademarks of their respective companies.

Configuration of product number

●DEVICES



●Evaluation Board



CONTENTS

1. DESCRIPTION

This document describes the basic procedure to open the TCP end points of the S1S60000 and to send and receive data to/from the TCP end points.
When the TCP end point of the S1S60000 is opened, TCP/IP data transmission is enabled between the S1S60000 and network devices.

2. SETUP BEFORE OPENING THE TCP END POINTS

First, open the SYSTEM communication end point of the S1S60000. For details, see the "S1S60000 Application Note: No.2 Ping Reply Method." In this example, use the IP address of 192.168.0.254 to open the SYSTEM communication end point.

3. OPENING THE TCP END POINT

The S1S60000 has four TCP end points: TCP0 to TCP3. In this example, TCP0 (end point number 2) is used and the "passive open" mode is selected that the S1S60000 waits for a connection request from a remote machine. For detailed commands, status and data transmission between the host CPU and the S1S60000, see the "Ping Reply Method".
This section describes how to open the TCP end point and check its connection from the host CPU. The procedure is as follows.

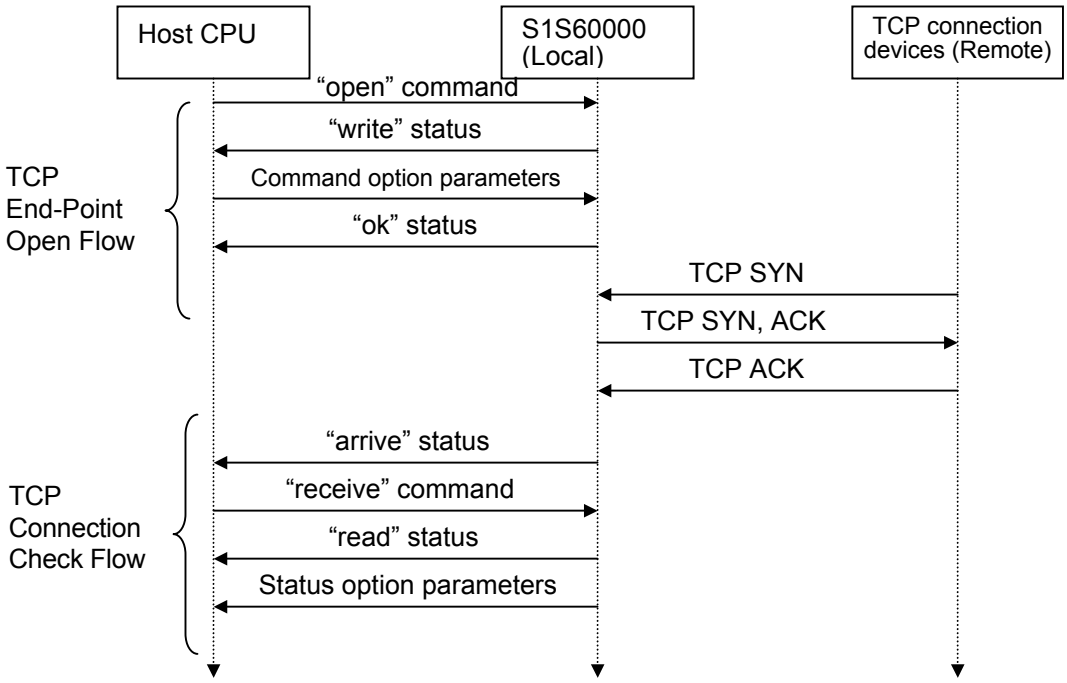


Fig.3.1 TCP End-Point Open Flow

3. OPENING THE TCP END POINT

3.1 Opening TCP End Point 0 (TCP0)

The host CPU issues "open" command (0x0120: sequence number 01, end point number 2 = TCP0, and command number 0 = Open) to the command port of the S1S60000.
The S1S60000 returns "write" status (0x0121: sequence number 01, end point number 2 = TCP0, and status number 1 = write). Read the "write" status from the Status port. In this example, the TCP end point is opened in "passive open" mode, the remote IP address is disabled, the remote port number is disabled, the local port number is enabled (49152), and the timeout value is disabled. Create the end-point open parameters in memory of the host CPU as follows, and write them into the data port.

Table 3.1 Command Option Parameter Set Values

BYTE	Write value	Contents	Comment
0	0x00	Fixed value	
1	0x00		
2	0x10	Flag	Bit 7 (passive open)=0, Bit 6 (Remote IP address disabled)=0, Bit 5 (Remote port number disabled)=0, Bit 4 (Local port number enabled)=1, Bit 3 (Reserved) =0, Bit 2 (Timeout value disabled)=0, Bit 1 (Reserved)=0, Bit 0 (Reserved)=0
3	0x00	Fixed value	
4	0x00	Remote IP address	Disabled (and omitted).
5	0x00		
6	0x00		
7	0x00		
8	0x00	Remote port number	Disabled (and omitted).
9	0x00		
10	0xc0	Local port number	49152(0xc000)
11	0x00		
12	0x00	Timeout value	Disabled timeout value 0
13	0x00		
14	0x00	Fixed value	
15	0x00		

When the command option parameters are stored in memory, BYTE 0 of Table 3.1 is set to the low-order addresses of memory.

The write data row for each host CPU type is as follows.

Table 3.2 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x00,0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xc0,0x00,0x00,0x00,0x00,0x00
8-bit connection BigEndian	0x00,0x00,0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xc0,0x00,0x00,0x00,0x00,0x00
16-bit connection LittleEndian	0x0000,0x0010,0x0000,0x0000,0x0000,0x00c0,0x0000,0x0000
16-bit connection BigEndian	0x0000,0x1000,0x0000,0x0000,0x0000,0xc000,0x0000,0x0000

* Access sequence during 8-bit connection: From low-order ports to high-order ports

When the “ok” status (0x0123: sequence number 01, end point number 2 = TCP0, and status number 3 = ok) is returned, the “open” command processing has completed. If the “working” status is returned instead of the “ok” status, the end point has already issued the “open” command or it is continuing asynchronous processing after the command issue. Issue “status” command (0x0227: sequence number 02, end-point number 2 = TCP0, and command number 7 = status) for this end point and make sure that the end point has been “timed out” or “closed.”

3.2 Checking the Established Connection

The connection to the TCP end point is NOT established yet even when the “ok” status is returned in response to the “open” command. The TCP connection is established by the S1S60000 automatically. When it is established, the S1S60000 outputs “arrive” status (0x002d: sequence number 00, end-point number 2 = TCP0, and status number d = arrive). From the host CPU, issue “receive” command (0x0322: sequence number 03, end-point number 2 = TCP0, and command number 2 = receive). When “read” status (0x0322: sequence number 03, end-point number 2 = TCP0, and status number 2 = read) is returned in response to the “receive” command, read the “read” status option parameters from the data port.
If the connection is established and if the remote machine has IP address 192.168.0.2 and port number 1039, the following data row is read.

Table 3.3 Readout of “receive-read” Status Option Parameters

BYTE	Write value	Contents	Comment
0	0x00	Received data length	The TCP control flag is received and, therefore, no data is usually contained.
1	0x00		
2	0x6a	Flag	Bit 7 (End of data)=0, Bit 6 (Remote IP address enabled)=1, Bit 5 (Remote port number enabled)=1, Bit 4 (unicast)=0, Bit 3 (Control flag enabled)=1, Bit 2 (Reserved)=0, Bit 1 (Data offset enabled)=1, Bit 0 (Overall data length disabled)=0
3	0x00	Fixed value	
4	0xc0	Remote IP address	192.168.0.2 (0xc0.0xa8.0x00.0x02)
5	0xa8		
6	0x00		
7	0x02		
8	0x04	Remote port number	1039 (0x040f)
9	0x0f		
10	0x12	Control flag	TCP control flag Bit 4 (ACK flag)=1, Bit 1 (SYN flag)=1
11	0x00		
12	0x00	Data offset	
13	0x00		
14	0x00	Overall data length	
15	0x00		

When the command option parameters are stored in memory, BYTE 0 of Table 3.3 is set to the low-order addresses of memory.

The read data row for each host CPU type is as follows.

Table 3.4 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x00,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x12,0x00,0x00,0x00,0x00,0x00
8-bit connection BigEndian	0x00,0x00,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x12,0x00,0x00,0x00,0x00,0x00
16-bit connection LittleEndian	0x0000,0x006a,0xa8c0,0x0200,0x0f04,0x0012,0x0000,0x0000
16-bit connection BigEndian	0x0000,0x6a00,0xc0a8,0x0002,0x040f,0x1200,0x0000,0x0000

* Access sequence during 8-bit connection: From low-order ports to high-order ports

In the BYTE10 setting of SYN flag (Bit1=1) and the absence of FIN flag (Bit0=0) indicate that the TCP connection has been established. As the receive data length is 0 byte, no receive data follows after the “read” status option.

4. RECEIVING DATA

This section explains how to read the data on the host CPU when the S1S60000 receives data from the network. The procedure is as follows.

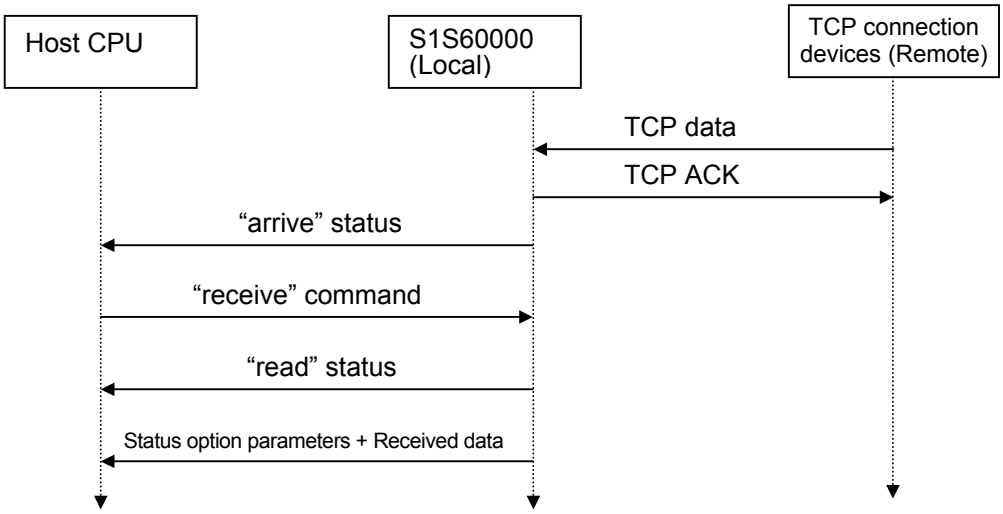


Fig.4.1 Flow of Host CPU Data Reception

When receiving data from the network, the S1S60000 outputs “arrive” status (0x002d: sequence number 00, end-point number 2 = TCP0, and status number d = arrive) to the host CPU. The host CPU issues “receive” command (0x0422: sequence number 04, end-point number 2 = TCP0, and command number 2 = receive). When “read” status (0x0422: sequence number 04, end-point number 2 = TCP0, and status number 2 = read) is returned in response to the “receive” command, read the received data from the data port.
The “read” status option parameters have the same format as those used when the TCP connection is established.
The first two bytes of “read” status option is the received data length. The length is shown in bytes. If the received data length is “n” bytes, the received data is read from the data port in units of 16 bits for “n/2” times if “n” is an even number or “(n+1)/2” times if “n” is an odd number. The received data is read in units of 8 bits for “n” times if “n” is an even number or “(n+1)” times if it is an odd number. Read it from the data port for the required number of times.
If the received data is the 11-byte character codes of “0123456789A” character string, the following data row may be read.

Table 4.1 “receive-read” Status Options and Readout of Received Data

BYTE	Write value	Contents	Comment
0	0x00	Received data length	11-byte data is received.
1	0x0b		
2	0x6a	Flag	Bit 7 (End of data)=0, Bit 6 (Remote IP address enabled)=1, Bit 5 (Remote port number enabled)=1, Bit 4 (unicast)=0, Bit 3 (Control flag enabled)=1, Bit 2 (Reserved)=0, Bit 1 (Data offset enabled)=1, Bit 0 (Reserved)=0
3	0x00	Fixed value	
4	0xc0	Remote IP address	192.168.0.2 (0xc0.0xa8.0x00.0x02)
5	0xa8		
6	0x00		
7	0x02		
8	0x04	Remote port number	1039 (0x040f)
9	0x0f		
10	0x18	Control flag	TCP control flag Bit 4 (ACK flag)=1, Bit 3 (PSH flag)=1
11	0x00		
12	0x00	Data offset	This is meaningless at the TCP end point.
13	0x00		
14	0x00	Overall data length	
15	0x00		
16	0x30	Received data	The 11-byte character codes (hex.) of character string “0123456789A” is received.
17	0x31		
18	0x32		
19	0x33		
20	0x34		
21	0x35		
22	0x36		
23	0x37		
24	0x38		
25	0x39		
26	0x41		
27	0xXX	padding	It is padded as the data is in odd-numbered bytes.

When the command option parameters are stored in memory, BYTE 0 of Table 4.1 is set to the low-order addresses of memory.

The read data row for each host CPU type is as follows.

Table 4.2 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x0b,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x18,0x00,0x00,0x00,0x00,0x00 0x30,0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38,0x39,0x41,0xXX
8-bit connection BigEndian	0x00,0x0b,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x18,0x00,0x00,0x00,0x00,0x00 0x30,0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38,0x39,0x41,0xXX
16-bit connection LittleEndian	0x0b00,0x006a,0xa8c0,0x0200,0x0f04,0x0018,0x0000,0x0000, 0x3130,0x3332,0x3534,0x3736,0x3938,0xXX41
16-bit connection BigEndian	0x000b,0x6a00,0xc0a8,0x0002,0x040f,0x1800,0x0000,0x0000, 0x3031,0x3233,0x3435,0x3637,0x3839,0x41XX

* Access sequence during 8-bit connection: From low-order ports to high-order ports

5. SENDING DATA

This section explains the procedure you perform on the host CPU to transmit data from the S1S60000 to the network. The procedure is as follows.

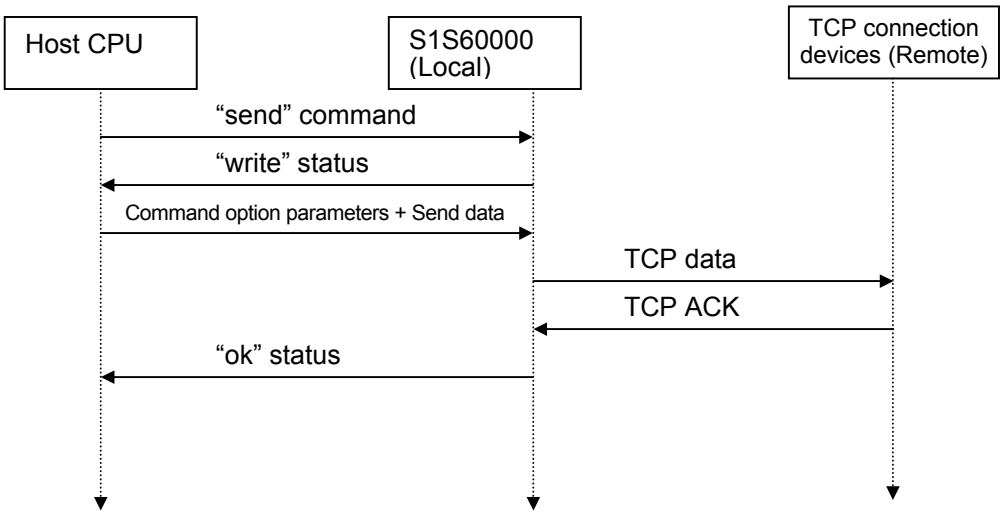


Fig.5.1 Flow of Host CPU Data Transmission

From the host CPU, issue “send” command (0x0521: sequence number 05, end point number 2 = TCP0, and command number 1 = send). When the S1S60000 returns “write” status (0x0521: sequence number 05, end point number 2 = TCP0, and status number 1 = write), write both the command option parameters and the send data (in this order) into the data port. This example sends 11-byte character codes (hex.) of character string “0123456789A.”

Table 5.1 Command Option Parameters and Send Data Values

BYTE	Write value	Contents	Comment
0	0x00	Send data length	11-byte data is sent.
1	0x0b		
2	0x00	Flag	Bits 7-6 (Reserved)=0, Bit 3 (Control flag disabled)=0, Bits 2-0 (Reserved)=0
3	0x00	Fixed value	
4	0x00		
5	0x00		
6	0x00		
7	0x00		
8	0x00		
9	0x00		
10	0x00	Control flag	TCP control flag No flag is set.
11	0x00		
12	0x00	Fixed value	
13	0x00		
14	0x00		
15	0x00		
16	0x30	Send data	The 11-byte character codes (hex.) of character string “0123456789A” is sent.
17	0x31		
18	0x32		
19	0x33		
20	0x34		
21	0x35		
22	0x36		
23	0x37		
24	0x38		
25	0x39		
26	0x41		
27	0xXX	padding	It is padded as the data is in odd-numbered bytes.

When the command option parameters are stored in memory, BYTE 0 of Table 5.1 is set to the low-order addresses of memory.

5. SENDING DATA

The write data row for each host CPU type is as follows.

Table 5.2 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x0b,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00 0x30,0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38,0x39,0x41,0xXX
8-bit connection BigEndian	0x00,0x0b,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00 0x30,0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38,0x39,0x41,0xXX
16-bit connection LittleEndian	0x0b00,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000, 0x3130,0x3332,0x3534,0x3736,0x3938,0xXX41
16-bit connection BigEndian	0x000b,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000, 0x3031,0x3233,0x3435,0x3637,0x3839,0x41XX

* Access sequence during 8-bit connection: From low-order ports to high-order ports

The number of times of writing the send data in the unit of 16 bits is “n/2” times when the send data length “n”(bytes) is an even number or “(n+1)/2” times if it is an odd number. Here, as the send data length (n) is 11 bytes, the send data is written in units of 16 bits into the data port for “(n+1)/2=6” times (total 14 times together with the command option of “16 bytes/2”=8). The send data is written in units of 8 bits for “n” times if “n” is an even number or “(n+1)” times if it is an odd number. From the host CPU, write the send data into the data port for the required number of times. The TCP data transmission is processed automatically by the S1S60000. When the S1S60000 has verified the data reception at the remote machine, the S1S60000 returns “ok” status (0x0523: sequence number 05, end-point number 2 = TCP0, command number 3 = ok).
If the “send” command has failed after retries of data transmission, the S1S60000 returns “cancel” status (0x0528: sequence number 05, end-point number 2 = TCP0, status number 8 = cancel) instead of the “ok” status. When receiving the “cancel” status, release the connection because further communication may fail. Usually, you can use the “close” command to release the connection. However, if the “cancel” status is returned, use the “abort” command because you may fail to communicate with the remote machine. The release of the connection is detailed in the subsequent “Releasing the TCP Connection” section.

6. RELEASING THE TCP CONNECTION

This section explains the standard procedure to release the TCP connection by issuing the release request from the host CPU and the procedure you must perform when a release of connection is requested from a remote machine.

6.1 Releasing the Connection from the Host CPU

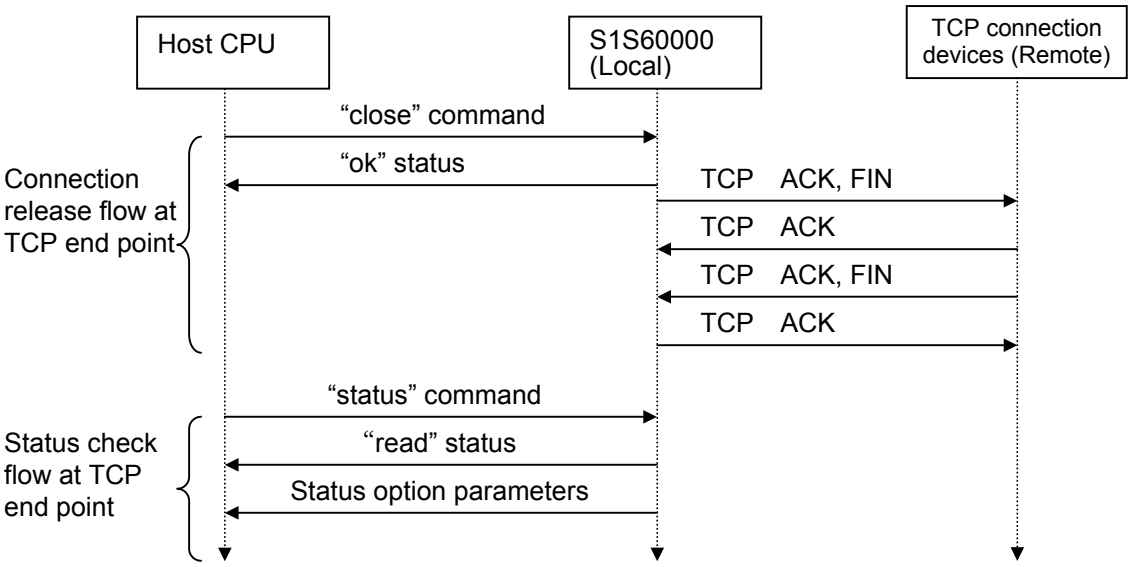


Fig.6.1 Connection Release Flow at TCP End Point

To release the existing TCP connection, issue "close" command (0x0624: sequence number 06, end-point number 2 = TCP0, and command number 4 = close). The S1S60000 returns "ok" status (0x0623: sequence number 06, end-point number 2 = TCP0, and status number 3 = ok). Then, the S1S60000 processes connection releasing asynchronously. If depends on the response time of the remote machine until the connection release processing completes. If the end point is "closed," you can "open" the connection again. You can issue "status" command (0x0727: sequence number 07, end-point number 2 = TCP0, command number 7 = status) to this end point and check the establishment of TCP connection. When the S1S60000 returns "read" status (0x0722: sequence number 07, end-point number 2 = TCP0, status number 2 = read) in response to the "status" command, get the "read" status option parameters from the data port. If the end point is "closed," the following data row may be read.

Table 6.1 Readout of “status-read” Status Option Parameters

BYTE	Write value	Contents	Comment
0	0x00	Fixed value	
1	0x00		
2	0x00	Flag	It is “0” because the status is “closed.”
3	0x00	Fixed value	
4	0x00	Remote IP address	It is “0” because the status is “closed.”
5	0x00		
6	0x00		
7	0x00		
8	0x00	Remote port number	
9	0x00		
10	0x00	Local port number	
11	0x00		
12	0x00	Timeout	
13	0x00		
14	0x01	Status of end point	“0x01=close” status for the TCP end point
15	0x00	Fixed value	

When the command option parameters are stored in memory, BYTE 0 of Table 6.1 is set to the low-order addresses of memory.
The read data row for each host CPU type is as follows.

Table 6.2 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x00
8-bit connection BigEndian	0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x00
16-bit connection LittleEndian	0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0001
16-bit connection BigEndian	0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0000,0x0100

* Access sequence during 8-bit connection: From low-order ports to high-order ports

You can issue the “abort” command to request for releasing of the TCP connection from the host CPU. In this case, the S1S60000 returns the “ok” status again. Unlike the “close” command, the S1S60000 simply sends an RST flag to the remote and changes the end point into the “closed” status.
The release of connection by the “abort” command is the abnormal procedure. Use the “abort” command if there is a compelling reason to do so such as no response by the remote.

6.2 Releasing the TCP Connection from Remote Machine

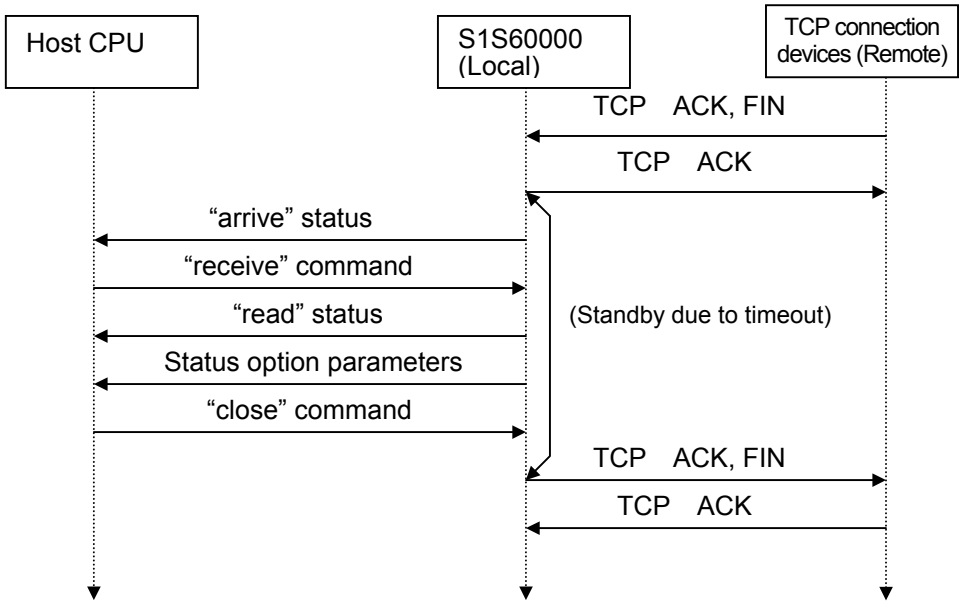


Fig.6.2 Connection Release Flow at TCP End Point

If the release of TCP connection is requested by a remote machine, S1S60000 outputs “arrive” status (0x002d: sequence number 00, end-point number 2 = TCP0, and status number d = arrive) in the same way as for the checkout of connection establishment and for data reception. From the host CPU, issue “receive” command (0x0822: sequence number 08, end-point number 2 = TCP0, and command number 2 = receive). When the S1S60000 returns “read” status (0x0822: sequence number 08, end-point number 2 = TCP0, and status number 2 = read) in response to the “receive” command, get the “read” status option parameters from the data port. If a remote machine requests for a release of the connection and if the S1S60000 receives a FIN flag, the data row of Table 6.3 can be read.
Also, after getting the “read” status option parameters, issue “close” command (0x0924: sequence number 09, end-point number 2 = TCP0, and command number 4 = close) from the host CPU. When receiving the “close” command, the S1S60000 notifies the release of TCP connection to the remote machine. However, if the “close” command is NOT issued, the S1S60000 waits for the specified period of timeout and automatically notifies the release of the connection to the remote machine.

Table 6.3 Readout of “receive-read” Status Option Parameters

BYTE	Write value	Contents	Comment
0	0x00	Received data length	The TCP control flag is received and therefore, no data is usually contained.
1	0x00		
2	0x6a	Flag	Bit 7 (End of data)=0, Bit 6 (Remote IP address enabled)=1, Bit 5 (Remote port number enabled)=1, Bit 4 (unicast)=0, Bit 3 (Control flag enabled)=1, Bit 2 (Reserved)=0, Bit 1 (Data offset enabled)=1, Bit 0 (Reserved)=0
3	0x00	Fixed value	
4	0xc0	Remote IP address	192.168.0.2 (0xc0.0xa8.0x00.0x02)
5	0xa8		
6	0x00		
7	0x02		
8	0x04	Remote port number	1039 (0x040f)
9	0x0f		
10	0x11	Control flag	TCP control flag Bit 4 (ACK flag)=1, Bit 0 (FIN flag)=1
11	0x00		
12	0x00	Data offset	
13	0x00		
14	0x00	Overall data length	
15	0x00		

When the command option parameters are stored in memory, BYTE 0 of Table 6.3 is set to the low-order addresses of memory.

The read data row for each host CPU type is as follows.

Table 6.4 Data Row for Each Host CPU Type

Host CPU type	Data row
8-bit connection LittleEndian	0x00,0x00,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x00,0x11,0x00,0x00,0x00,0x00
8-bit connection BigEndian	0x00,0x00,0x6a,0x00,0xc0,0xa8,0x00,0x02,0x04,0x0f,0x00,0x11,0x00,0x00,0x00,0x00
16-bit connection LittleEndian	0x0000,0x006a,0xa8c0,0x0200,0x0f04,0x1100,0x0000,0x0000
16-bit connection BigEndian	0x0000,0x6a00,0xc0a8,0x0002,0x040f,0x0011,0x0000,0x0000

* Access sequence during 8-bit connection: From low-order ports to high-order ports

EPSON

International Sales Operations

AMERICA

EPSON ELECTRONICS AMERICA, INC.
HEADQUARTERS
150 River Oaks Parkway
San Jose, CA 95134, U.S.A.
Phone: +1-800-228-3964 FAX: +1-408-922-0238

SALES OFFICES

West
1960 E.Grand Avenue Flr 2
El Segundo, CA 90245, U.S.A.
Phone: +1-800-249-7730 FAX: +1-310-955-5400

Central
101 Virginia Street, Suite 290
Crystal Lake, IL 60014, U.S.A.
Phone: +1-800-853-3588 FAX: +1-815-455-7633

Northeast
301 Edgewater Place, Suite 210
Wakefield, MA 01880, U.S.A.
Phone: +1-800-922-7667 FAX: +1-781-246-5443

Southeast
3010 Royal Blvd. South, Suite 170
Alpharetta, GA 30005, U.S.A.
Phone: +1-877-332-0020 FAX: +1-770-777-2637

EUROPE

EPSON EUROPE ELECTRONICS GmbH
HEADQUARTERS
Riesstrasse 15
80992 Munich, GERMANY
Phone: +49-89-14005-0 FAX: +49-89-14005-110

DÜSSELDORF BRANCH OFFICE
Altstadtstrasse 176
51379 Leverkusen, GERMANY
Phone: +49-2171-5045-0 FAX: +49-2171-5045-10

FRENCH BRANCH OFFICE
1 Avenue de l' Atlantique, LP 915 Les Conquerants
Z.A. de Courtaboeuf 2, F-91976 Les Ulis Cedex, FRANCE
Phone: +33-1-64862350 FAX: +33-1-64862355

BARCELONA BRANCH OFFICE
Barcelona Design Center
Edificio Testa, C/Alcalde Barnils 64-68, Modulo C 2a planta
E-08190 Sant Cugat del Vallès, SPAIN
Phone: +34-93-544-2490 FAX: +34-93-544-2491

UK & IRELAND BRANCH OFFICE
8 The Square, Stockley Park, Uxbridge
Middx UB11 1FW, UNITED KINGDOM
Phone: +44-1295-750-216/+44-1342-824451
FAX: +44-89-14005 446/447

Scotland Design Center
Integration House, The Alba Campus
Livingston West Lothian, EH54 7EG, SCOTLAND
Phone: +44-1506-605040 FAX: +44-1506-605041

ASIA

EPSON (CHINA) CO., LTD.
23F, Beijing Silver Tower 2# North RD DongSanHuan
ChaoYang District, Beijing, CHINA
Phone: +86-10-6410-6655 FAX: +86-10-6410-7320

SHANGHAI BRANCH
7F, High-Tech Bldg., 900, Yishan Road,
Shanghai 200233, CHINA
Phone: +86-21-5423-5522 FAX: +86-21-5423-5512

EPSON HONG KONG LTD.
20/F., Harbour Centre, 25 Harbour Road
Wanchai, Hong Kong
Phone: +852-2585-4600 FAX: +852-2827-4346
Telex: 65542 EPSCO HX

EPSON Electronic Technology Development (Shenzhen) LTD.
12/F, Dawning Mansion, Keji South 12th Road,
Hi-Tech Park, Shenzhen
Phone: +86-755-2699-3828 FAX: +86-755-2699-3838

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110
Phone: +886-2-8786-6688 FAX: +886-2-8786-6677

HSINCHU OFFICE
No. 99, Jiangong Road,
Hsinchu City 300
Phone: +886-3-573-9900 FAX: +886-3-573-9169

EPSON SINGAPORE PTE., LTD.
1 HarbourFront Place,
#03-02 HarbourFront Tower One, Singapore 098633
Phone: +65-6586-5500 FAX: +65-6271-3182

SEIKO EPSON CORPORATION
KOREA OFFICE
50F, KLI 63 Bldg., 60 Yoido-dong
Youngdeungpo-Ku, Seoul, 150-763, KOREA
Phone: +82-2-784-6027 FAX: +82-2-767-3677

GUMI OFFICE
2F, Grand B/D, 457-4 Songjeong-dong,
Gumi-City, KOREA
Phone: +82-54-454-6027 FAX: +82-54-454-6093

SEIKO EPSON CORPORATION
SEMICONDUCTOR OPERATIONS DIVISION

IC Sales Dept.
IC Marketing Group
421-8, Hino, Hino-shi, Tokyo 191-8501, JAPAN
Phone: +81-42-587-5814 FAX: +81-42-587-5117

Document Code: 410503800
First Issue January 2006
Printed in JAPAN Ⓗ