HT-4139-28/48 Scanner TWAIN Driver

Developer's Guide

HITACHI

Preface

Thank you very much for using our TWAIN Driver.

This manual explains the specification and the function of the triplet needed when the TWAIN driver is controlled from the application.

Read this manual carefully to use the TWAIN Driver properly.

Notes

- It is strictly prohibited to reprint or copy part or all of this manual without our permission.
- The content of this manual is subject to change without prior notice due to improvement.

In this manual following shortened forms are used.
- Hitachi scanner TWAIN driver (Hereinafter “TWAIN driver”)
- User interface of Hitachi scanner TWAIN driver (Hereinafter “User interface of TWAIN driver”)

Indication of non-Hitachi trademarks
Windows XP, Windows Vista,Windows7, Visual C++, and Visual Basic are registered trademarks of Microsoft Corporation in the United States and other countries.

Feb 2008 (First edition)
July 2009 HT-4134 connected with same PC
Jan 2010 Patchcode function
Mar 2010 Add Ean128 BarCode

Read Prior to Operation

This manual does not describe about the TWAIN standard,
It is necessary to understand the TWAIN standard to develop / test application programs.

Read “HT-4139-28/48 Scanner TWAIN Driver User’s Guide” first.

This manual does not describe about the TWAIN driver user interface.

(1) How to scan
There are 3 ways to scan images with this driver.

1) Displaying the user interface of the TWAIN driver and setting parameter to scan images. With this screen, multiple parameter setting can be saved in a parameter file.

2) Without displaying the user interface of the TWAIN driver, scan images using the parameters set by 1).

3) Without displaying the user interface of the TWAIN driver, set Capabilities by a triplet and scan images.

Please switch the parameter file referring to Chapter 2 when operating it by the method of 2)

The application that can display TWAIN driver's user interface because it sets the parameter in method of 3) is necessary.

(2)Stacker

There are Hitachi scanners that equip a dual stacker.

Stacker for sheet ejection will be selectable by using a custom triplet.

(3) Imprinting (Numbering)

Hitachi scanners support an imprinting function that can be used for scanning check.

Contents of Imprinting can be modified before printing by using a custom triplet.

Precautions for Use

Continuous scanning

If the application program does not support the continuous scanning, performance may become lower than expectation.
If sheets are too thin, copy papers, with glue, or mixed sizes, multiple feeding error might occur. Using options and/or the sheet check function, try to scan them correctly.

Scanned images

To confirm if the scanned data satisfy the level of customer's accuracy and quality, check the gradation and density of scanned images.
This TWAIN driver does not support pre-view function.

Scanned data

This device can detect sheet sizes and multiple feeding but it might fail to catch multiple feeding depending on conditions.
Inspect all scanned image data after scanning sheets and make sure that all sheets are scanned correctly.
Extra attention is required when check will not performed or sheets will be discarded.

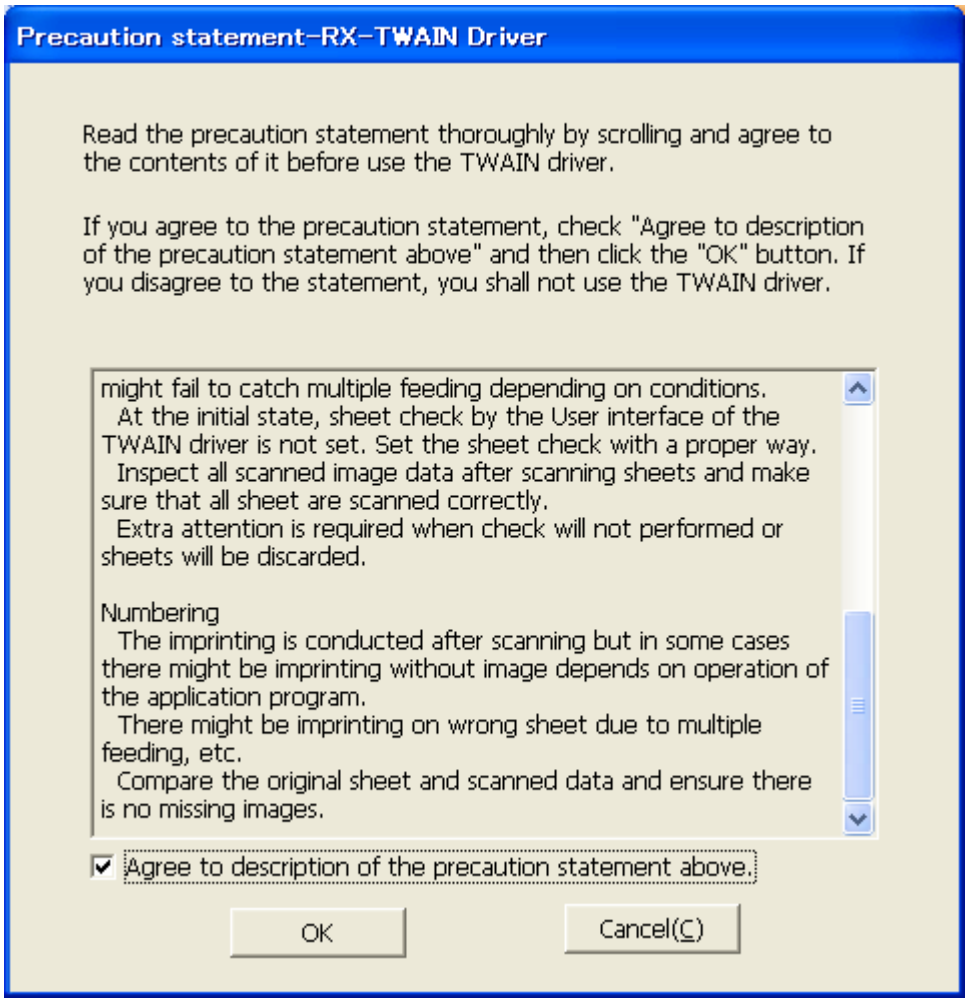
Imprinting (Numbering)

The imprinting is conducted after scanning but in some cases there might be imprinting without image depends on operation of the application program.
There might be imprinting on wrong sheet due to multiple feeding error, etc.
Compare the original sheet and scanned data and ensure there are no missing images.

Note for System Set up

Keep followings in your mind when you set up systems with the TWAIN driver.

When you use the driver first time, a dialog of “Precaution statement” will be displayed without displaying the TWAIN driver user interface.
Understand the contents and check mark Agree to “Precaution statement” and click the “OK” button. This dialog will not be shown later once you agree.（If you delete this program and reset it later, you will see this dialog again.）



When you will version up a driver from a previous version that you have not agreed, it will be displayed.

Default is set without function. Add needed functions for operation. And make sure they are installed correctly.

Hitachi scanners can check scanning status with the multiple feeding error detection and the imprinting function but it is not always perfect. Set up systems so that it can check if images in sheets are the same as scanned ones.

Scanning without displaying the TWAIN driver user interface and you do not use the parameter files, the sheet checking will not be performed except a device equips a ultrasonic sensor

Scanning without displaying the TWAIN driver user interface and using the parameter files, setting of Capabilities by the triplet will be ignored.

Items that can be set are shown in the table below.

No.	Function		Triplet Set Capabilities	TWAIN User Interface Or Use parameter file
1	Sheet setting	Standard size	support	support
2		Width	support	support
3		Height	support	support
4		Thickness	no	support
5		Scanning width	no	support
6		Unit	support	support
7	－	Continuous scanning	support	support
8	Sheet check	Length check	no	support
9		Length check	no	support
10		Width check	no	support
11		Thickness check（Transmitted light, Roller displacement）	no	Support(note1)
12		Thickness check（Ultrasonic）	support(note2)	support(note1)
13		Skew check	support	support
14	Scanning image setting	Only surface side	support	support
15		Only back side	no	support
16		Surface + Back side	support	support
17		Multiple faces	no	support
18		Image check	no	support
19		Alternate process of 4-point detection error	no	support
20		Surface side barcode	support	support
21		Back side barcode	no	support

No.	Function		Triplet Set **Capabilities**	TWAIN User Interface Or Use parameter file
22	Input	Image kind	no	support
23		Image setting	Support (note3) (note4)	support(note4)
24		Drop out	Support	support
25		Density（simple）	Support	support
26		Density（detail）	No	support
27		Resolution	support(note5)	support
28		Expand definition setting	no	support
29	Edit	Rotation	no	support
30		Backgroud cropping	specified size only	support
31		Position correction	no	support
32		Skew correction	no	support
33		Gradation reverse	no	support
34		Output resolution conversion	no	support
35		Delete background	no	support
36		Bordering	no	support
37	Feedin	Auto scanning	no	support
38	g	Ejection confirmation	no	support
39		Continuous feeding	no	support
40		Error processing	no	support
41		Scanning speed	no	support
42		Hopper position	no	support
43	Imprint`	Imprint	support(note6)	support
44		Erase imprint	no	support

（note1）Require scanner option

（note2）Require scanner option, default of ultrasonic sensor.

Disabled when [SCANOPTION]ULTRASONIC=NO of RxTwainMng.ini

（note 3）Color / gray require option. Some type of scanner does not support halftone.

（note 4）Not support half tone

（note 5）Not support for 240 DPI

(note 6)It is Only 1Place Printing. There is a limitation in the character that can be printed.

Contens

Preface 1
Notes 2
Read Prior to Operation 3
Precautions for Use 5
Note for System Set up 6

Chapter 1. Application Development 1-1
1.1 Application Control 1-1
1.2 Event loop of application 1-2
1.3 Continuous Scanning 1-3

Chapter 2. Parameter Switching 2-1
2.1 Non-UI Mode Parameter Settings Dialog 2-2
2.2 How to Start the TWAIN Default Setting Tool 2-3
2.3 File of Parameter Specification 2-5

Chapter 3. Operation Triplets 3-1
3.1 List of Operation Triplets 3-1
3.2 Details of Operation Triplets 3-3
3.3 Data Structures 3-3
3.4 Custom Triplets 3-4
3.5 Details of Custom Triplets 3-5

Chapter 4. List of Functions 4-1
4.1 List of Supported Functions 4-1

Appendix A-1
A-1 MSG_XFERREADY at Image Scanning A-1
A-2 Error Message A-2
A-3 Notes on File Output A-3
A-4 Notes on the Barcode A-4

Chapter 1. Application Development

When you develop an application program that controls a TWAIN scanner, it is necessary to understand the TWAIN standard (TWAIN Release v1.9) fully.

1.1 Application Control

Followings should be prepared to start and control a session through the TWAIN application.

This TWAIN driver complies the TWAIN standard and consists of 7 protocol states.

State 1 - Pre-Session
State 2 - Source Manager Loaded
State 3 - Source Manager Open
State 4 - Source Open
State 5 - Source Enabled
State 6 - Transfer is Ready
State 7 - Transferring

Proceed application as follows.

Load the Source Manager and Get the DSM_Entry (State 1 to 2)
. Open the Source Manager (State 2 to 3)
. Select the Source (during State 3)
. Open the Source (State 3 to 4)
. Negotiate Capabilities with the Source (during State 4)
. Request the Acquisition of Data from the Source (State 4 to 5)
. Recognize that the Data Transfer is Ready (State 5 to 6)
. Start and Perform the Transfer (State 6 to 7)
. Conclude the Transfer (State 7 to 6 to 5)
. Disconnect the TWAIN Session (State 5 to 1 in sequence)

Points to notice on control will be described from the next chapter.

1.2 Event loop of application

This driver reports followings when it is required to get data from a source, depending on the status of a scanner.

MSG_XFERREADY：Completion of data transfer preparation
MSG_CLOSEDSREQ：Require to disable a source user interface

Event loop of application should be able to handle the report, MSG_NULL、MSG_XFERREADY、MSG_CLOSEDSREQ.

(Note)
However it is possible to process only MSG_NULL、MSG_XFERREADY by following A-1 in the appendix of this manual but it is not recommended.

1.3 Continuous Scanning

This driver can get multiple images by continuous scanning both side scanning, or single side scanning. In this case, keep issuing DG_IMAGE/DAT_IMAGE***XFER/MSG_GET until TW_PENDINGXFERS.Count=0.

If you want to suspend transferring image, issue DG_CONTROL/DAT_PENDINGXFERS/MSG_RESET first.

Issue CAP_AUTOFEED for continuous scanning and use a document feeder.

It is possible to scan sheets on the feeder by issuing DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS multiple times but every time a stacker moves downward.

If a stacker moves downward every time, it would cause degradation of process efficiency and skew. Also it would be a cause of noise and injury.

Chapter 2. Parameter Switching

It is necessary to save a parameter file by the TWAIN driver user interface for parameter switching.

There are 3 ways to switch parameter files.
(1) Use a mode parameter specification tool (without UI)
(2) Rewrite parameter specification file
(3) Use a custom triplet

Process of (1) and (2) is the same, switch parameters by writing in the parameter specification file.

Switching by the parameter specification file is inevitably applied when you use this TWAIN driver.

Refer to 2.5.1 for the way of (3).

Notes for using parameter files
- Changes of Capabilities (size, etc.) by the application will be invalid.
- Not scanning automatically by parameter setting.
- Setting of continuous scanning is specified by CAP_AUTOFEED.
- When other scanners are used or the scanner option is changed, the parameter file cannot be used.

Notes when HT-4139 Image scanner and HT-4134 Image scanner are
- Process of (1)and(2) becomes common setting of HT-4139 and HT-4134.
  When the scanner used is switched, it is necessary to set it again.

2.1 Non-UI Mode Parameter Settings Dialog

In the mode without TWAIN user interface, the TWAIN driver operates according to the specified values. With this tool, however, you can work with the parameters where the settings are already saved.

Starting this tool displays the Non-UI Mode Parameter Settings Dialog as shown in Figure 2.1.

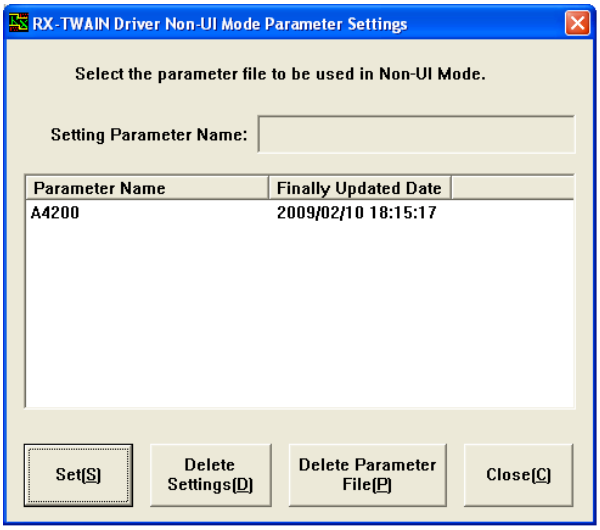


Figure 2.1 Non-UI Mode Parameter Settings Dialog

Table 2.1 Settings and Functions of the Non-UI Mode Parameter Settings Dialog

No.	Item	Description
1	Setting Parameter Name	Displays the parameter name used in the mode without TWAIN user interface. Note: At starting, the parameter name set currently is displayed.
2	Set	Sets the parameter name specified in the Parameter Name field can be used in the mode without TWAIN user interface.
3	Delete Settings	Deletes the parameter name setting.
4	Delete Parameter File	Deletes the parameter file.
5	Close	Terminates the operation.

Changing the size and other settings from the application is disabled.

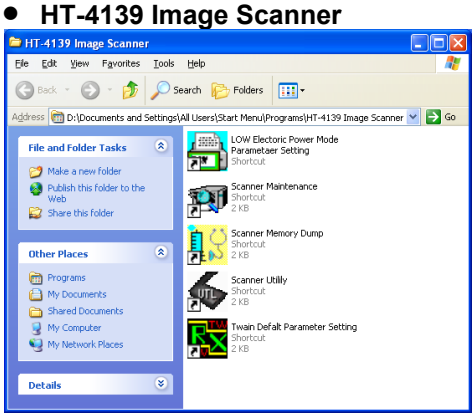
It is fixed that no automatic scanning is executed in the parameter setting.

Continuous input follows the CAP_AUTOFEED specification.

2.2 How to Start the TWAIN Default Setting Tool

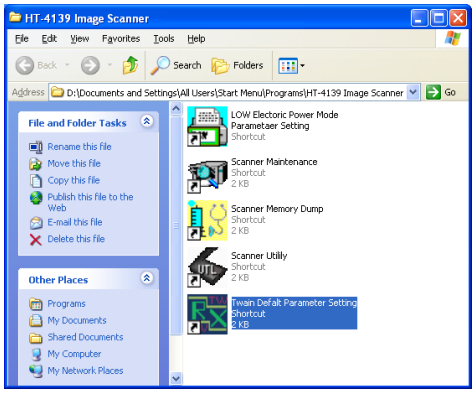
(1) When starting from the HT-4139 Image Scanner folder

- **HT-4139 Image Scanner**



1 Select the “HT-4139 ImageScanner “folder.

- **Twain Defalt Parameter Setting**

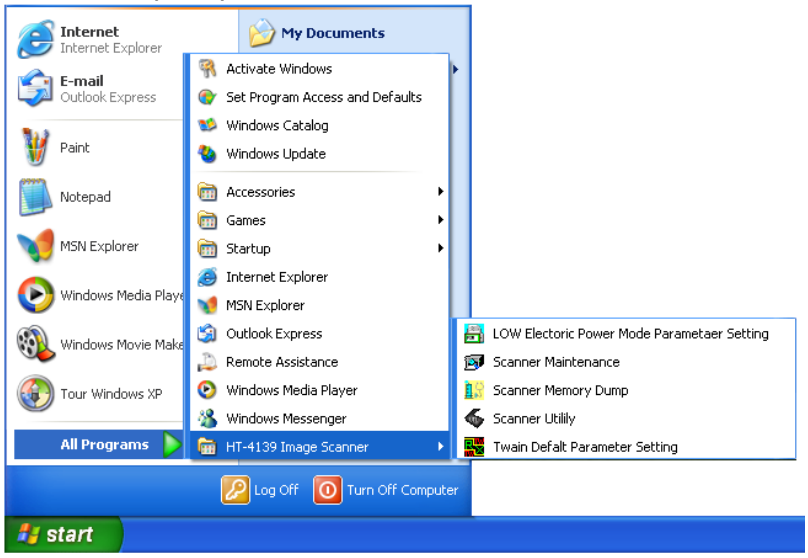


2 Double-click the “Twain Defalt Parameter Setting” icon in the “HT-4139 ImageScanner” folder.

This starts the Non-UI Mode Parameter Settings Tool.

(2) When starting from the “Start” menu

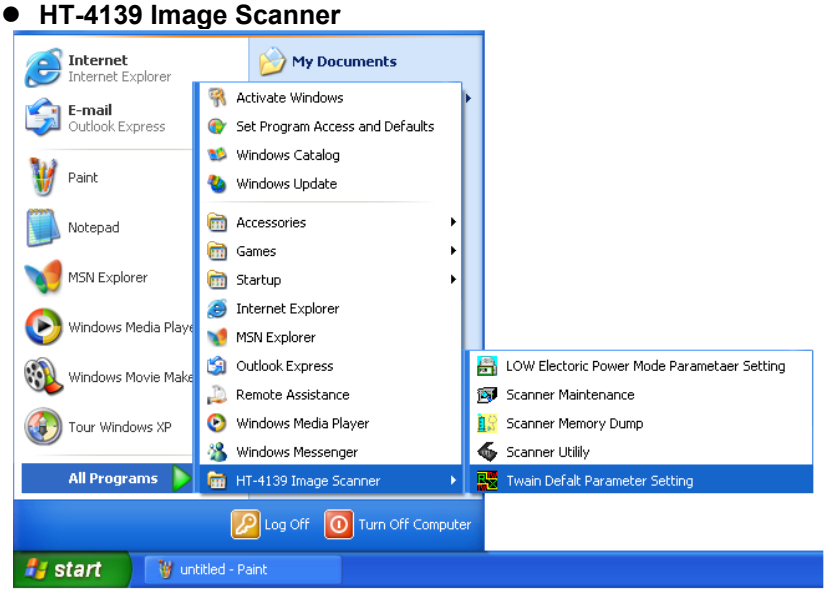
- **Taskbar (Start)**



1 Select “Start” of the taskbar.

2 Then select “All Programs”.

- **HT-4139 Image Scanner**



3 Additionally, select “HT-4139 ImageScanner”.

4 Select “Twain Defalt Parameter Setting”.

This starts the Non-UI Mode Parameter Settings Tool.

2.3 File of Parameter Specification

The file for saving the specified contents is the “OCR32¥ETC32¥RxTwainUser.ini” file, which is saved in the following format.

[RXTWAIN]
GUILESSFILENAME=XXXXXXXX XXXXXXXX: file-name
(No extension)

File name is that specified when parameter is saved.

By changing this specification before opening the TWAIN driver, you can operate the file with the parameters that have the settings.

GetPrivateProfileString,WritePrivateProfileString of WindowsAPI can be used for read/write file.

Note:
If invalid names are set to this parameter, it will be a cause of errors.

Chapter 3. Operation Triplets

3.1 List of Operation Triplets

The list of operation triplets supported by the TWAIN driver is shown in Table 3.1 to Table 3.3.

Table 3.1 List of Operation Triplets for Control Information (from Application to Source)

No.	Data group	Data type	Message
1	DG_CONTROL	DAT_CAPABILITY	MSG_GET MSG_GETCURRENT MSG_GETDEFAULT MSG_QUERYSUPPORT MSG_RESET MSG_SET
2	DG_CONTROL	DAT_EVENT	MSG_PROCESSEVENT
3	DG_CONTROL	DAT_PENDINGXFERS	MSG_ENDXFER MSG_GET MSG_RESET
4	DG_CONTROL	DAT_SETUPMEMXFER	MSG_GET
5	DG_CONTROL	DAT_STATUS	MSG_GET
6	DG_CONTROL	DAT_USERINTERFACE	MSG_DISABLEDS MSG_ENABLEDS
7	DG_CONTROL	DAT_XFERGROUP	MSG_GET
8	DG_CONTROL	DAT_SETUPFILEXFER	MSG_GET MSG_GETDEFAULT MSG_RESET MSG_SET

Table 3.2 List of Operation Triplets for Image Information (from Application to Source))

No.	Data group	Data type	Message
1	DG_IMAGE	DAT_IMAGEINFO	MSG_GET
2	DG_IMAGE	DAT_IMAGELAYOUT	MSG_GET MSG_GETDEFAULT MSG_RESET
3	DG_IMAGE	DAT_IMAGEMEMXFER	MSG_GET
4	DG_IMAGE	DAT_IMAGENATIVEXFER	MSG_GET
5	DG_IMAGE	DAT_IMAGEFILEXFER	MSG_GET
6	DG_IMAGE	DAT_EXIMAGEINFO	MSG_GET
7	DG_IMAGE	DAT_IMAGEFILEXFER	MSG_GET
8	DG_IMAGE	DAT_EXIMAGEINFO	MSG_GET (Note 1)

Table 3.3 Control Information from Source Manager to Source

No.	Data group	Data type	Message
1	DG_CONTROL	DAT_IDENTITY	MSG_CLOSEDS MSG_GET MSG_OPENDS

Note 1: Only when ICAP_EXTIMAGEINFO is TRUE

3.2 Details of Operation Triplets

Details of the operation triplets comply with the TWAIN Release v1.9 specifications

3.3 Data Structures

The data structures comply with the TWAIN Release v1.9 specifications.

3.4 Custom Triplets

Control Information from Application to Source

No.	Data group	Data type	Message
1	DG_CONTROL	DAT_NOUIPARMFILE	MSG_SET MSG_RESET
2	DG_CONTROL	DAT_DIVSCANCTL	MSG_SET MSG_RESET
3	DG_CONTROL	DAT_EJECTCTL	MSG_SET MSG_RESET
4	DG_CONTROL	DAT_EJECTNUMBERING	MSG_GET MSG_SET MSG_RESET

Constants

DAT_NOUIPARMFILE	0x8001
DAT_DIVSCANCTL	0x8002
DAT_EJECTCTL	0x8003
DAT_EJECTNUMBERING	0x8004

3.5 Details of Custom Triplets

3.5.1 DG_CONTROL/DAT_NOUIPARMFILE/MSG_SET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_NOUIPARMFILE,MSG_SET,
pNoUIParmFile);
pNoUIParmFile:pTW_NOUIPARMFILE

typedef struct{
TW_STR128 NoUIParmFile; //Parameter file name
}TW_NOUIPARMFILE, FAR * pTW_NOUIPARMFILE;

Enabled status:
Only 4

Explanation:
The source reads a specified setting file in the non-UI mode.
A failure in reading the setting file specified here results in starting by default.
The parameter file name must be set to 128 bytes or less.

Notes on the application:
The application must perform this specification before issuing
DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS.
This setting ignores the setting by the TWAIN Default Setting Tool.

Return code

TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_BADVALUE When the setting file name is 128 bytes or more
TWCC_SEQERROR Invalid sequence

3.5.2 DG_CONTROL/DAT_NOUIPARMFILE/MSG_RESET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_NOUIPARMFILE,MSG_RESET,
pNoUIParmFile);
pNoUIParmFile:pTW_NOUIPARMFILE

Enabled status:
From 4 to 7

Explanation:
The setting is that the source reads no parameter files in the non-UI mode.
There is no need to specify pNoUIParmFile, either.

Notes on the application
There are no particular settings or operations.
This specification is not indispensable, either.

Return code

TWRC_SUCCESS	Success
TWRC_FAILURE	
TWCC_SEQERROR	Invalid sequence

3.5.3 DG_CONTROL/DAT_DIVSCANCTL/MSG_SET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL, DAT_DIVSCANCTL,MSG_SET,
pDivScanCtl);
pDivScanCtl:pTW_DIVSCANCTL

typedef struct{
TW_BOOL DivScanFlg; //Is it necessary to divide scan and ejection?
}TW_DIVSCANCTL, FAR * pTW_DIVSCANCTL;

Enabled status:
Only 4

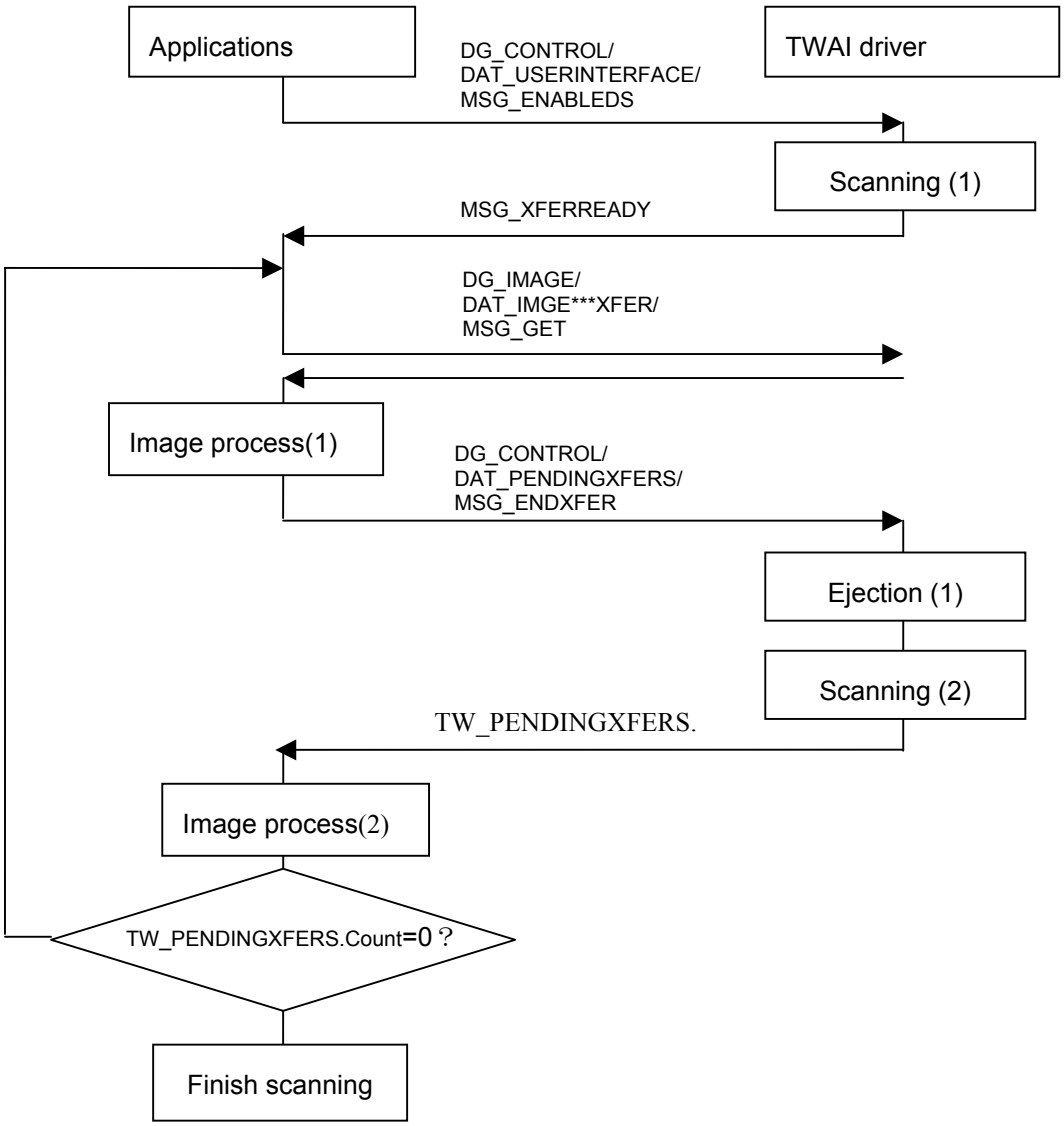
Explanation:
This specification is specified when you want to divide scan and ejection. The source performs scan and ejection according to this specification.
The DAT_EJECTCTL and DAT_EJECTNUMBERING specifications are enabled only when this specification requires scan and ejection to be divided for execution.

Notes on the application
The application must perform this specification before issuing DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS.
If this specification requires the division of scan and ejection, the source ejects a sheet being scanned while receiving DG_CONTROL/DAT_PENDINGXFERS/MSG_ENDXFER.
If the scan and ejection are not divided, the source is ejected automatically after scanning.

Return code
TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_BADVALUE If the division of scan and ejection is TRUE other than in the extension mode
TWCC_SEQERROR Invalid sequence

Note for when scanning process and ejection process are separated

The operation when scanning process and ejection process are separated is shown below.



There are 2 timings for scanned image process, image process (1) and image process (2). Operations are different against the ejection (1) after ENDXFER and error at scanning (2).

Image process (1)
Image will be processed even fail at ejection (1).
Image will be processed even fail at scanning(2).

There is a case that sheet will not be ejected even it is processed.
(There is a case that imprinting will not be done even it is processed.)

Image process (2)
Image will NOT be processed even fail at ejection (1).
Image will NOT be processed even fail at scanning (2).

There is a case that it will be not-processed even it is ejected.
(There is a case that it will be not-processed even it is imprinted.)

Consider cases above when you develop an application. And consider how to handle errors for these cases.

3.5.4 DG_CONTROL/DAT_DIVSCANCTL/MSG_RESET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL, DAT_DIVSCANCTL,MSG_RESET,
pDivScanCtl);
pDivScanCtl:pTW_DIVSCANCTL

Enabled status:
From 4 to 7

Explanation:
The specification for dividing scan and ejection is set to None.
There is no need to specify pNoUIParmFile, either.

Notes on the application
There are no particular settings or operations.
This specification is not indispensable, either.

Return code

TWRC_SUCCESS	Success
TWRC_FAILURE	
TWCC_SEQERROR	Invalid sequence

3.5.5 DG_CONTROL/DAT_EJECTCTL/MSG_SET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_EJECTCTL,MSG_SET,
pEjectCtl);
pEjectCtl:pTW_EJECTCTL

```
typedef struct{
TW_UINT16    EjectInstruct;    //Ejection indication flag
}TW_EJECTCTL, FAR * pTW_EJECTCTL;
```

Enabled status:
From 6 to 7

Explanation:
Specifies the sheet ejection indication. According to this specification, the source makes ejection to a normal or abnormal stacker.
For ejection to a normal stacker, specify 0 in the ejection instruction flag.
For ejection to an abnormal stacker, specify 1 in the ejection instruction flag.
Once a sheet is ejected, the content of this specification is deleted and the source will determine the ejection instruction.

Notes on the application
The source ejects a sheet being scanned while receiving DG_CONTROL/DAT_PENDINGXFERS/MSG_ENDXFER.
The application must perform this specification before issuing DG_CONTROL/DAT_PENDINGXFERS/MSG_ENDXFER.
This can be specified only when DG_CONTROL/DAT_DIVSCANCTL/MSG_SET is used to specify the division of scan and ejection for execution.

Return code
TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_BADVALUE When the ejection instruction flag is other than 0 and 1
TWCC_SEQERROR Invalid sequence

3.5.6 DG_CONTROL/DAT_EJECTCTL/MSG_RESET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_EJECTCTL,MSG_RESET,
pEjectCtl);
pEjectCtl:pTW_EJECTCTL

Enabled status:
From 4 to 7

Explanation:
The specification for the sheet ejection indication is set to None.
There is no need to specify pEjectCtl, either.

Notes on the application
There are no particular settings or operations.
This specification is not indispensable, either.

Return code

TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_SEQERROR Invalid sequence

3.5.7 DG_CONTROL/DAT_EJECTNUMBERING/MSG_GET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_EJECTNUMBERING,MSG_GET,
pNumberingCnt);
pNumberingCnt:pTW_UINT16

Enabled status:
From 6 to 7

Explanation:
For numbering at the ejection of a sheet, you can acquire the maximum value for the specifiable number of numbering print characters. When you specify numbering characters at the ejection of a sheet, they must be not greater than the number of numbering print characters entered by this specification.
The number of numbering print characters is returned to pNumberingCnt.

Notes on the application
The number of numbering characters returned by this content is the number of numbering print characters in the UI-specified numbering settings.

Return code
TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_SEQERROR Invalid sequence

Limitations:
This operation is supported only by the front side numbering.
When the back side numbering is set, this operation returns a value of 0 as the number of characters.

3.5.8 DG_CONTROL/DAT_EJECTNUMBERING/MSG_SET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_EJECTNUMBERING,MSG_SET,
pEjectNumbering);
pEjectNumbering:pTW_EJECTNUMBERING
typedef struct {
TW_BOOL NumberingFlg; //Whether to perform imprinter numbering
TW_STR128 NumberingChar; //Number of characters for imprinter numbering
}TW_EJECTNUMBERING, FAR * pTW_EJECTNUMBERING;

Enabled status:
From 6 to 7

Explanation:
The source performs numbering print according to the specified settings at the time of sheet-ejection numbering.
However, this is limited to the UI-specified numbering print.
You cannot change any of the following items if they are UI-specified: printing side, printing place, canceling numbering place, canceling digit number, printing orientation, printable check, printing position, and printing character control.

The number of digits for numbering is that specified by UI when you use UI or parameter.

The digit number that allow numbering print is the UI-specified digit number.

Other cases, it will be the number that specified by CAP_PRINTERSTRING, CAP_PRINTERSUFFIX.
If the number of digits is larger than specified one or specified invalid characters, the setting of this triplet will be invalid and the numbers specified by UI, parameter, or CAP_PRINTERSTRING, CAP_PRINTERSUFFIX will be applied.

DG_CONTROL/DAT_EJECTNUMBERING/MSG_GET can be used to acquire the digit number of numbering. Specify the numbering print characters not exceeding the acquired digit number.
If NumberingFlg is set to FALSE, the source does not print numbering.
If NumberingFlg is set to TRUE, the source performs numbering print for NumberingChar.
The following table shows a list of characters that can be specified for numbering print.
Character strings that can be specified by scanner
HT-4139
A to Z, 0 to 9, one-byte space, one-byte symbols: !”#$%&'()*+,-./ :;<=>? @[¥]^_ `{|}~「」

(Note) The six symbols, which cover ^, *, #, ~, [, and], serve as special characters for numbering

print and each of them must be preceded by the “~” symbol as an escape sequence.
(Example) If you want to print “ABC#[123]”, you need to specify “ABC~#~ [123~]”.

If you specify *, the stacker symbol is printed.
If you specify ^, the erased character is printed.

Notes on the application
The source ejects a sheet being scanned for numbering print while receiving DG_CONTROL/DAT_PENDINGXFERS/MSG_ENDXFER.
The application must perform this specification before issuing DG_CONTROL/DAT_PENDINGXFERS/MSG_ENDXFER.
This can be specified only when DG_CONTROL/DAT_DIVSCANCTL/MSG_SET is used to specify the division of scan and ejection for execution.

Return code
TWRC_SUCCESS Success
TWRC_FAILURE
TWCC_BADVALUE The numbering print characters had invalid data.
The maximum specified number of printing characters was exceeded.
TWCC_SEQERROR Invalid sequence

Limitations:
This operation is supported only by the front side numbering.
When the back side numbering is set, this operation results in an error.

3.5.9 DG_CONTROL/DAT_EJECTNUMBERING/MSG_RESET

Format:
DS_Entry(pOrigin,pDest,DG_CONTROL,DAT_EJECTNUMBERING,MSG_RESET,
pEjectNumbering);
pEjectNumbering:pTW_EJECTNUMBERING

Enabled status:
From 4 to 7

Explanation:
The specification for numbering in the ejection mode is set to None.
This is no need to specify pEjectNumbering, either.

Notes on the application
There are no particular settings or operations.
This specification is not indispensable, either.

Return code

TWRC_SUCCESS	Success
TWRC_FAILURE	
TWCC_SEQERROR	Invalid sequence

Chapter 4. List of Functions

4.1 List of Supported Functions

The list of supported functions is shown in Tables 4.1 and 4.2.

Table 4.1 List of Supported Functions (Support Messages and Containers)

No.	Function	Message				
		MSG_GET	MSG_GETCURRENT	MSG_GETDEFAULT	MSG_RESET	MSG_SET
1	CAP_SUPPORTEDCAPS	OK	OK	OK	NG	NG
2	CAP_AUTOFEED	OK	OK	OK	NG	OK
3	CAP_DEVICEONLINE	OK	OK	OK	NG	NG
4	CAP_DUPLEX	OK	OK	OK	NG	OK
5	CAP_DUPLEXENABLED	OK	OK	OK	NG	OK
6	CAP_FEEDERENABLED	OK	OK	OK	NG	OK
7	CAP_FEEDERLOADED	OK	OK	OK	NG	NG
8	CAP_INDICATORS	OK	OK	OK	NG	OK
9	CAP_PAPERDETECTABLE	OK	NG	NG	NG	NG
10	CAP_UICONTROLLABLE	OK	NG	NG	NG	NG
11	CAP_XFERCOUNT	OK	OK	OK	OK	OK
12	ICAP_BARCODEDETECTIONENABLED	OK	OK	OK	NG	OK
13	ICAP_BARCODEMAXSEARCHPRIORITIES	OK	OK	OK	NG	OK
14	ICAP_BARCODESEARCHMODE	OK	OK	OK	NG	OK
15	ICAP_BARCODESEARCHPRIORITIES	OK	OK	OK	NG	OK
16	ICAP_BITDEPTH	OK	OK	OK	NG	OK
17	ICAP_BITORDER	OK	OK	OK	NG	NG
18	ICAP_COMPRESSION	OK	OK	OK	NG	NG
19	ICAP_FILTER	OK	OK	OK	NG	OK
20	ICAP_GAMMA	OK	OK	OK	NG	OK
21	ICAP_HIGHLIGHT	OK	OK	OK	NG	OK
22	ICAP_IMAGEFILEFORMAT	OK	OK	OK	OK	OK
23	ICAP_PHYSICALHEIGHT	OK	OK	OK	NG	NG
24	ICAP_PHYSICALWIDTH	OK	OK	OK	NG	NG
25	ICAP_PIXELFLAVOR	OK	OK	OK	OK	OK

Continued on Next Page

Continued

No.	Function	Message				
		MSG_GET	MSG_GETCURRENT	MSG_GETDEFAULT	MSG_RESET	MSG_SET
26	ICAP_PIXELTYPE	OK	OK	OK	OK	OK
27	ICAP_PLANARCHUNKY	OK	OK	OK	NG	NG
28	CAP_PRINTER	OK	OK	OK	NG	OK
29	CAP_PRINTERENABLED	OK	OK	OK	NG	OK
30	CAP_PRINTERINDEX	OK	OK	OK	NG	OK
31	CAP_PRINTERMODE	OK	OK	OK	NG	OK
32	CAP_PRINTERSTRING	OK	OK	OK	OK	OK
33	CAP_PRINTERSUFFIX	OK	OK	OK	OK	OK
34	ICAP_SHADOW	OK	OK	OK	NG	OK
35	ICAP_SUPPORTEDSIZES	OK	OK	OK	NG	NG
36	ICAP_SUPPORTEDBARCODETYPES	OK	NG	NG	NG	NG
37	ICAP_THRESHOLD	OK	OK	OK	NG	OK
38	ICAP_UNITS	OK	OK	OK	OK	OK
39	ICAP_XFERMECH	OK	OK	OK	OK	OK
40	ICAP_XRESOLUTION	OK	OK	OK	NG	OK
41	ICAP_YRESOLUTION	OK	OK	OK	NG	OK
42	ICAP_EXTIMAGEINFO	OK	OK	OK	NG	OK
43	ICAP_PATCHCODEDETECTIONENABLED	OK	OK	OK	NG	OK
44	ICAP_PATCHCODEMAXSEARCHPRIORITIES	OK	OK	OK	NG	OK
45	ICAP_PATCHCODESEARCHMODE	OK	OK	OK	NG	OK
46	ICAP_PATCHCODESEARCHPRIORITIES	OK	OK	OK	NG	OK
47	ICAP_SUPPORTEDPATCHCODETYPES	OK	NG	NG	NG	NG

Table 4.2 List of Supported Functions (Default and Capability Values)

No.	Function	Default value	Capability value (allowable value)
1	CAP_SUPPORTEDCAPS	None	All supported functions
2	CAP_AUTOFEED	FALSE	TRUE FALSE
3	CAP_DEVICEONLINE	TRUE	TRUE FALSE
4	CAP_DUPLEX	TWDX_1PASS DUPLEX	TWDX_1PASSDUPLEX
5	CAP_DUPLEXENABLED	FALSE	FALSE TRUE
6	CAP_FEEDERENABLED	TRUE	TRUE
7	CAP_FEEDERLOADED	TRUE	TRUE
8	CAP_INDICATORS	FALSE	FALSE TRUE
9	CAP_PAPERDETECTABLE	FALSE	FALSE
10	CAP_UICONTROLLABLE	TRUE	TRUE
11	CAP_XFERCOUNT	-1	-1 to 32767
12	ICAP_BARCODEDETECTIONE NABLED	FALSE	TRUE FALSE
13	ICAP_BARCODEMAXSEARCHP RIORITIES	1	1
14	ICAP_BARCODESEARCHMOD E	TWBD_HORZ	TWBD_HORZ TWBD_VERT TWBD_HORZVERT
15	ICAP_BARCODESEARCHPRIO RITIES	TWBT_3OF9	TWBT_3OF9 TWBT_2OF5INTERLEAVED TWBT_CODE128 TWBT_CODABAR TWBT_EAN8 TWBT_EAN13
16	ICAP_BITDEPTH	1	1,4,8,24
17	ICAP_BITORDER	TWBO_ MSBFIRST	TWBO_MSBFIRST
18	ICAP_COMPRESSION	TWCP_NONE	TWCP_NONE
19	ICAP_FILTER	TWFT_RED	TWFT_RED TWFT_GREEN TWFT_BLUE TWFT_NONE

Continued on Next Page

Continued

No.	Function	Default value	Capability value (allowable value)
20	ICAP_GAMMA	0	30 to 500
21	ICAP_HIGHLIGHT	0	-256 to 256
22	ICAP_IMAGEFILEFORMAT	TWFF_BMP	TWFF_TIFF TWFF_BMP TWFF_JFIF
23	ICAP_PHYSICALHEIGHT	305(mm)	305(mm) (*1)
24	ICAP_PHYSICALWIDTH	420(mm)	420(mm) (*1)
25	ICAP_PIXELFLAVOR	TWPF_ CHOCOLATE	TWPF_CHOCOLATE
26	ICAP_PIXELTYPE	TWPT_BW	TWPT_BW TWPT_GRAY TWPT_RGB
27	ICAP_PLANARCHUNKY	TWPC_ CHUNKY	TWPC_CHUNKY
28	CAP_PRINTER	TWPR_IMPRI NTERBOTTO MAFTER	TWPR_IMPRINTERBOTTOMA FTER
29	CAP_PRINTERENABLED	FALSE	FALSE TRUE (*2)
30	CAP_PRINTERINDEX	0	0-9999
31	CAP_PRINTERMODE	TWPM_SINGL ESTRING	TWPM_SINGLESTRING TWPM_COMPOUNDSTRING
32	CAP_PRINTERSTRING	““	Character string (*3)
33	CAP_PRINTERSUFFIX	““	%1,%2,%3,%4,%5,%6,%7 (*4)
34	ICAP_SHADOW	0	-128 to 128
35	ICAP_SUPPORTEDSIZES	TWSS_NONE	TWSS_NONE TWSS_USLETTER TWSS_USLEGAL TWSS_A3 TWSS_A4 TWSS_A5 TWSS_A6 TWSS_A7
36	ICAP_SUPPORTEDBARCODET YPES	-	TWBT_3OF9 TWBT_2OF5INTERLEAVED TWBT_CODE128 TWBT_CODABAR TWBT_EAN8 TWBT_EAN13

Continued on Next Page

Continued

No.	Function	Default value	Capability value (allowable value)
37	ICAP_THRESHOLD	128	0 to 255
38	ICAP_UNITS	TWUN_ INCHES	TWUN_INCHES TWUN_CENTIMETERS
39	ICAP_XFERMECH	TWSX_ NATIVE	TWSX_NATIVE TWSX_FILE TWSX_MEMORY
40	ICAP_XRESOLUTION	200(dpi)	100,150,200,300,400,600,240 (dpi) (*5, *6)
41	ICAP_YRESOLUTION	200(dpi)	100,150,200,300,400,600,240 (dpi) (*5, *6)
42	ICAP_EXTIMAGEINFO	FALSE	TRUE (*7) FALSE
43	ICAP_PATCHCODEDETECTION ENABLED	FALSE	TRUE FALSE
44	ICAP_PATCHCODEMAXSEARC HPRIORITIES	6	6
45	ICAP_PATCHCODESEARCHMO DE	TWBD_HORZ	TWBD_HORZ TWBD_VERT TWBD_HORZVERT TWBD_VERTHORZ
46	ICAP_PATCHCODESEARCHPR IORITIES	1:PATCH1 2:PATCH2 3:PATCH3 4:PATCH4 5:PATCH6 6:PATCHT	TWPCH_PATCH1 TWPCH_PATCH2 TWPCH_PATCH3 TWPCH_PATCH4 TWPCH_PATCH6 TWPCH_PATCHT
47	ICAP_SUPPORTEDPATCHCOD ETYPES	—	TWPCH_PATCH1 TWPCH_PATCH2 TWPCH_PATCH3 TWPCH_PATCH4 TWPCH_PATCH6 TWPCH_PATCHT

*1: The unit varies depending on the ICAP_UNITS settings.

*2: TRUE is only when the numbering mechanism exists.

*3: The following is a list of characters that can be specified.

Scanner	Characters strings that can be specified
HT-4139	A to Z, 0 to 9, one-byte space, one-byte symbols: !”#$%&'()*+,-./ :;<=>? @[¥]^_ `{|}~「」

*4: %1:YYYY(year) %2:YY(year) %3:MM(month) %4:DD(day) %5: Value (4-digit) calculated from CAP_PRINTERINDEX %6:stacker symbol %7:space
CAP_PRINTERMODE prints only the following: TWPM_COMPOUNDSTRING.

*5: If the application sets a resolution value outside the scanner's capability value range, set a value close to the supported value is set and TWRC_CHECHSTATUS is returned. Therefore, check that the application is the set value.

*6: 240 DPI performed reduction from 300 DPI.

*7: Only when it is TRUE, DG_IMAGE/DAT_EXTIMAGEINFO/MSG_GET is enabled.

A-1 MSG_XFERREADY at Image Scanning

This TWAIN driver will issue MSG_XFERREADY when an application requires DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS and will be ready status.
It will issue MSG_CLOSEDSREQ if an error happens and will not be ready status.

Depending an application, after issuing DG_CONTROL/DAT_USERINTERFACE/MSG_ENABLEDS it would receive MSG_XFERREADY only. As MSG_CLOSEDSREQ will be ignored, operation cannot be continued.

In these cases, set an option that forced to issue MSG_XFERREADY even error status.
This option is common to whole systems. (It can not be set for each application, user individually.)
When you set the option, ensure if an error occurs after receipt of MSG_XFERREADY.

MSG_XFERREADY will be issued even in Not-ready, operation after that can not be performed.

Set information of this option in registry.

Key :HKEY_LOCAL_MACHINE
subkey :SOFTWARE¥HITACHI¥SOFTOCR¥CurrentVersion¥TWAIN¥FORCEREADY
name of the value :No Name
type code :REG_SZ
data :YES or NO （case-sensitive）
Default:NO

CAUTION
In registry, important information for computer hardware and software to operate system are stored.
If there were an error in the registry, computer would be unstable or would not work.
Before you edit the registry be sure to back up it.

A-2 Error Message

This TWAIN driver reports error messages without UI when errors occur.

This driver displays error message on the most front layer of scanning process But if a screen of other job is front of that of scanning process, error message will be displayed behind the screen of other process.
If you prefer to see error message at front, set an option to support it. This option is common to whole systems. (It can not be set for each application, user individually.)

Set information of this option in registry.

Key :HKEY_LOCAL_MACHINE
subkey :SOFTWARE¥HITACHI¥SOFTOCR¥CurrentVersion¥TWAIN¥ERRACT
name of the value :No Name
type code :REG_SZ
data :YES or NO （case-sensitive）
Default:NO

CAUTION
In registry, important information for computer hardware and software to operate system are stored.
If there were an error in the registry, computer would be unstable or would not work.
Before you edit the registry be sure to back up it.

A-3 Notes on File Output

The image file to be output by the file transfer output (DG_IMAGE/DAT_IMAGEFILEXFER/MSG_GET) varies depending on the format set by DG_CONTROL/DAT_SETUPFILEXFER/MSG_SET and the image to be output.

You need to pay attention to the following files created by this driver.

Format of the Output Image File Requiring Attention

Format	Output image	Remarks
TWFF_TIFF	1	MMR compression
	4	Non-compression
	8	JPEG compression
	24	JPEG compression
TWFF_JFIF	1	Output after converting to 8 bits
	4	Output after converting to 8 bits

A-4 Notes on the Barcode

The bar code recognition does not always work perfectly.

You cannot acquire the barcode information if a part of the barcode is not search or if the total recognition characters exceed 255 bytes.
The following table shows the correspondence between the barcode type on the Scanning Image Settings Dialog and the type reported by TWEI_BARCODETYPE in DG_IMAGE/DAT_EXTIMAGEINFO/MSG_GET.

Barcode Type Correspondence Table

Barcode type in the Reading Image Settings	Type of TWEI_BARCODETYPE
EAN13	TWBT_EAN13
EAN8	TWBT_EAN8
Codabar	TWBT_CODABAR
Code39	TWBT_3OF9
Industrial 2 of 5	TWBT_2OF5INTERLEAVED
Code128	TWBT_CODE128
EAN128	TWBT_UCC128

DG_IMAGE/DAT_EXTIMAGEINFO/MSG_GET returns the same value even if you issue it many times when the same sheet is being scanned.