DMC Co., Ltd.

**Projected Capacitive Touch Screen Controller Board
DUS Series Software Specifications**

Table of Contents

1. Applied Product

This specification sheet describes specifications for the command interface of the projected capacitive touch screen controller boards, DUS series.

2. Connection System

The communication interface with the host computer is defined below.

2-1. UART Interface

Interface	UART
Baud rate	57600
Data bit	8bits
Parity	None
Stop bit	1bit
Flow control	None

2-2. USB Interface

Interface	USB 2.0 HID 1.1 (Windows standard driver)
Transfer rate	Full-speed 12[Mbps]
Power supply	Bus-powered
Power type	Low power device (100[mA] or less)

2-3. HID ID

Windows7/XP support version

Item	Value	Content
Vendor ID	0x0AFA	2810 (DMC VID)
Product ID	0x07D2	2002
Release ID	0x0001	Value may vary according to FW

Windows8/7 support version

Item	Value	Content
Vendor ID	0x0AFA	2810 (DMC VID)
Product ID	0x07D3	2003
Release ID	0x0001	Value may vary according to FW

3. Report Format

Coordinate data transmitted from touch screen controller to the host computer is defined below.

Transmission of the coordinate data will start once the host computer sends instruction for starting via coordinate output control command.

3-1. Coordinate data for WindowsXP

Product ID=0x07D2 or 0x07D3 (Format will be the same for either Product ID)

Report ID is 0x01, and mouse format will be placed after it.

0	1	2	3	4	5
Report ID	SW	X coordinate lower	X coordinate Upper	Y coordinate lower	Y coordinate upper
1byte	1byte	2bytes		2bytes	

Report ID: 0x01fixed

SW: 0x00=up,   0x01=down

Format of the coordinate data will be the same between USB and UART.

3-2. Coordinate Data for Windows7

Firmware of Product ID=0x07D2

Report ID will be 0x04, then mouse format will be placed after it.

When touched with 2 fingers, coordinate will be informed one by one. (Serial version)

0	1	2	3	4	5
Report ID	SW	X coordinate Lower	X coordinate Upper	Y coordinate Lower	Y coordinate Upper
1 byte	1 byte	2bytes		2bytes	

Report ID：   0x04 fixed

SW: b7:   1 fixed

　　 b6:   0=Packet Connection、1=Delimiter（1 Packet Ends）

　　 b5-b2:   Finger ID(start from 0）Touch ID can be specified in multi-touch operation.

　　 b1:   1 fixed

　　 b0:   0=up, 1=down

Format of the coordinate data will be the same between USB and UART.

3-3. Coordinate Data for Windows 8 / 7

Firmware of Product ID=0x07D3

Report ID will be 04h, and coordinate will be placed in touch format.

Because of hybrid mode transfer, coordinate of multiple fingers can be assigned to 1 packet. If number of simultaneous touched points exceeds the maximum finger number per packet (max 5 fingers per packet), the touched coordinate points will be reported with multiple packets (e.g. if 10 touch points are detected, the coordinate points will be reported with 2 packets).

[Multi-touch hybrid mode transfer format]

0	Finger 1 information								
	1	2	3	4	5	6	7	8	9
Report ID	SW	X coordinate (Lower)	X coordinate (Upper)	Y coordinate (Lower)	Y coordinate (Upper)	Width (Lower)	Width (Upper)	Height (Lower)	Height (Upper)
1 byte	1 byte	2bytes		2 bytes		2bytes		2bytes	

(Finger 1 information continues to the below)

Finger n information （n : 2 ～ 5）								
(n-1)*9 +1	(n-1)*9 +2	(n-1)*9 +3	(n-1)*9 +4	(n-1)*9 +5	(n-1)*9 +6	(n-1)*9 +7	(n-1)*9 +8	(n-1)*9 +9
SW	X Coordinate (Lower)	X Coordinate (Upper)	Y Coordinate (Lower)	Y coordinate (Upper)	Width (Lower)	Width (Upper)	Height (Lower)	Height (Upper)
1byte	2bytes		2bytes		2bytes		2bytes	

(Finger n information continues to the below)

9n+1	9n+2	9n+3	9n+4	9n+5
Scan Time				Contact Count
(Lower)			(Upper)	
4bytes				1byte

Repot ID: 04h fixed

SW: b5-b2:   Finger ID (from 0)

     b0:   0= up, 1=down

     b7, b6, b1:   unused (1fixed)

X coordinate, Y coordinate: X/Y coordinates of touch contact by1 finger

Width, Height: Width and height of touch contact by 1 finger.

Scan Time: Scan time for touch reporting. The initial Down detection is 0, then elapsed time will be set by 100usec. (Up detection will ends the timing measurement, and the next Down detection will reset the measured result). *1

Contact Count: Touch number of simultaneous touch detected *2

Operation will comply with multi-touch hybrid transfer mode of Windows8 as below.

*1 When multiple packets are transmitted, the same time will be set for all the packets (multiple packets will be treated as the same time frame)

*2 When multiple packets are transmitted, all the detected simultaneous touched points will be set to the first packet, then Contact Count in subsequent packet will be set to 0.

Coordinate at UART connection will be **serial mode transfer format** (the same specs as for 3-2. Windows7).

4. Command

Commands transmitted from the host PC to touch screen controller are defined in this section.

Commands are the same between USB and UART unless otherwise described.

*Note

At UART connection, issue a command only after 100ms or longer time passed from booting or reset
-canceling.

4-1. Basic Format

All the commands will be transmitted and received in binary data.

Basic format of command and response data

Header	Command	Byte to send	Argument	Data
1 byte	1 byte	1 byte	1 byte	N byte

Number of the byte to send = Argument + Data

4-2. Command List

Command: ［Host PC → TPC］

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Parameter [4]	Function	Response from controller
0x02	0x4C('L')	0x02	0x04	0x00	Acquisition of version information	Yes
0x02	0x4C('L')	0x02	0x06	0x00	Detail information of firmware	Yes
0x02	0x4C('L')	0x02	0x80	data	Setting up	No
0x02	0x4C('L')	0x02	0x81	data	Control of Coordinate output	No
0x02	0x4C('L')	0x01	0x17	-	Adjust Offset	Yes
0x02	0x4C('L')	0x01	0x01	-	Calibrate Offset	Yes

<Usage examples of the commands>

The host computer will perform the processing below, after booting.
a. Issue the version information acquisition command
   Action by the host computer: Check the self-diagnosis information at booting state and confirm if it is in a
   good condition.

b. Acquire detail information of firmware
   Width(X) and height(Y) of "Logical maximum value" outputted from the firmware can be acquired from
this information. Convert the acquired values to your logical coordinate system. The logical coordinate
values (maximum values) outputted from the controller vary according to sizes of the touch screens.

c. Issue the setup command to select coordinate data format
     *1 This process can be skipped in case of Win7 mode.
     *2 Select "Windows7 mode" in case of multi-touch operation.
d. Issue the coordinate output start command
     The controller will be in the mode to transmit coordinate data at touch.
    *3 Coordinate data will not be transmitted at touch before completion of the above processing.
       (Unless connected via USB under Windows7/8 environment)

4-3. Command Definition

4-3-1. Version Information Acquisition Command

This command is for acquiring version information of touch screen firmware.

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]
0x02	0x4C('L')	0x02	0x04	0x00（Backup）

4-3-2. Firmware Detail Information Command

This command is for acquiring detail information of touch screen firmware.

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]
0x02	0x4C('L')	0x02	0x06	0x00（Backup）

4-3-3. Setup Command (only for UART)

This command is for configuring setup data of touch screen firmware.

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]
0x02	0x4C('L')	0x02	0x80	data

data：Setup data

　0x03：Host OS=Windows7

　0x04：Host OS=WindowsXP

　* The host OS is Windows7 in the default setting.

　　In case of WindowsXP, configure the setting with this command before starting coordinate output.

4-3-4. Coordinate Output Control Command (only for UART)

This command is for starting and stopping coordinate output

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]
0x02	0x4C('L')	0x02	0x81	data

data： Start and stop

　0x00：Stop

　0x01：Start

　*Default status after booting is stop status.

　　After sending the coordinate output start command, coordinates will be outputted at touch on the touch screen.

4-3-5. Offset Calibration

After embedding the touch screen into the machine, execute an offset calibration by following the procedures below. The controller will learn the strength of capacitance couplings between the touch screen and bezel/chassis, and write the learned information into its ROM.

Note:

Make sure to perform the Offset Calibration described here before operating touch screen. Otherwise, touch screen may not operate correctly. The Offset Calibration has the same effect as executing TPOffset.exe (application software for Windows) provided by DMC.

1. Stop coordinate outputs by the Coordinate Output Control Command (4-3-4).
2. Issue the Adjust Offset Command.

Header [0]	Command [1]	Byte to send [2]	Argument [3]
0x02	0x4C('L')	0x01	0x17

*It will take few seconds until this command is received. The duration to take will vary according to size of the touch screen.

Confirm that the response as below will be received, then proceed to the next step.

Response

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]	Note
0x02	0x4C('L')	0x02	0x17	0x01	Normal end
0x02	0x4C('L')	0x02	0x17	0x00	Abnormal end

3. Issue the Calibrate Offset Command

Header [0]	Command [1]	Byte to send [2]	Argument [3]
0x02	0x4C('L')	0x01	0x01

*It will take approximately 5 to 8 seconds until this command is received. The time depends on size of the touch screen.

Confirm that the response as below will be received, then proceed to the next step.

Response

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4]	Note
0x02	0x4C('L')	0x02	0x01	0x01	Normal end
0x02	0x4C('L')	0x02	0x01	0x00	Abnormal end

4. Start coordinate output by the Coordinate Output Control Command (4-3-4).

4-4. Response Data

Response Data: ［Touch Screen Controller → Host Computer］

4-4-1. Version Information Acquisition

(1) Normal Response

Response by touch screen controller to the version information acquisition command

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Data [4 - n]
0x02	0x4C('L')	Variable	0x04	Described below

Each version of touch screen firmware will be transmitted（DMC's administrative information）

Bytes to send are variable. All the data are ASCII codes.

Byte to send

  The sum of argument (1byte) and data

Data format

    "nnnnnnnnnnnnnn PID-iiii:PROG-pppp DATA-dddd TEST=xx"

nnnnnnnn:	Product name
iiii:	Product ID
pppp:	Program version
dddd:	Data version
xx:	Self-diagnostic result

                                      Normal ：OK

                                        Error ：NG (error code) *Refer to self-diagnostic Test (section 5)

*There may not be output of TEST=xx in some FW versions.

For reference information, there will be no output of TEST=xx in FWs created after middle of Dec, 2013.

If you want to check the result of self-diagnostic test, refer to the section 5. [Self-diagnostic Test].

a. The result of self-diagnostic: Example of normal response

02,4C,36,04,44,55,53,31,30,30,30,5F,35,70,37,5F,45,58,43,30,35,37,42,30,36,30,41,3A,50,52,4F,47,2D,31,30,36,31,20,44,41,54,41,2D,56,30,36,2E,30,30,20,54,45,53,54,3D,4F,4B

  The data part (blue letters) converted into Ascii-code is indicated as below.

  DUS1000_5p7_EXC057B060A:PROG-1061 DATA-V06.00 TEST=OK

b. The result of self-diagnostic: Example of error response

02,4C,3C,04,44,55,53,31,30,30,30,5F,35,70,37,5F,45,58,43,30,35,37,42,30,36,30,41,3A,50,52,4F,47,2D,31,30,36,31,20,44,41,54,41,2D,56,30,36,2E,30,30,20,54,45,53,54,3D,4E,47,28,30,78,33,31,29

  02,4C,0E,06,18,12,00,01,02,FF,17,FF,11,D5,01,63,01

The data part (blue letters) converted into Ascii-code is indicated as below

  DUS1000_5p7_EXC057B060A:PROG-1061 DATA-V06.00 TEST=NG(0x31)

*Byte of the data [4 - n] included in response is not fixed. It may vary according to models.

4-4-2. Firmware Detail Information Command

（1）Normal Response

Response to the firmware detail information command

Header [0]	Command [1]	Byte to send[2]	Argument [3]	Data			
				[4]	[5]	[6]	[7]
0x02	0x4C('L')	0x0E or 0x10	0x06	Number of X electrodes	Number of Y electrodes	Unused	Unused

Data									
[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	
Number of Touch Points	Logical maximum value Width(X)		Logical maximum value Height(Y)		Physical maximum value Width(X)		Physical maximum value Height(Y)		
	(Low)	(High)	(Low)	(High)	(Low)	(High)	(Low)	(Height)	

Data					
[17]	[18]	[19]	[20]	[21]	[22]
W_Z value presence/ absence	Backup	Control items for DMC use （DATA Flash version）			
		(Low)			(High)

Number of X/Y Electrode: Number of X-axis/Y-axis electrodes

Number of Touch Points: Maximum number of multiple simultaneous touches

Logical Maximum Value Width(X)/Height(Y):

Logical maximum values of Width(X) and Height (Y) will be indicated in pixel.

Physical Maximum Value Width(X)/Height(Y):

Actual panel size of width(X) and height(Y) will be indicated in inch (x.xx inch).

If the byte to send is 0x10, data[17] ～[18] will be added.

W_Z value presence/absence: 0=W_Z absent

1= W_Z present

Backup: 0x00

If number of bytes to send is 0x14, [19]～[22] will be added (for some FWs created after Oct, 2013).Control items for DMC use

(2) Error response

If an error occurred, the all the bytes will be 0.

4-4-3. Setup Command

    There is no response data to setup command.

4-4-4. Coordinate Output Control Command

    There is no response data to coordinate output control command.

5. Self-diagnostic Test

The self-diagnostic test will be performed automatically at booting.

・Data Flash（Parameter region、Calibration region）

・Raw Data

Result of self-diagnostic test will be acquired.

Header [0]	Command [1]	Byte to send[2]	Argument [3]	Data [4]
0x02	0x4C('L')	0x02	0x09	0x00（Backup）

Response to the command for acquiring the result of self-diagnostic test

Header [0]	Command [1]	Byte to send [2]	Argument [3]	Processing result [4]	Diagnostic result [5]
0x02	0x4C('L')	0x03	0x09	0x01: Normal 0x00: Error	0x00: Normal Not 0x00: Error

Refer to the below for the error codes.

5-1. Data Flash

(1)Parameter Region

Parameter region in the data flash (information used for scanning electrodes) will be checked.

・Version Check (version inconsistency between the basic firmware and parameter firmware, or writing error on data region)

・LENGTH Check （Improper data）

・SUM Check （Improper data）

・MODE Check （Improper data）

No.	Error Code	Detail of the error	Remark
1	0x11	Version Error	
2	0x12	Length Error	
3	0x13	Sum Error	
4	0x14	Mode Error	

(2) Calibration Region

Calibration region in data flash will be checked.

・Version Check （Version inconsistency between the basic firmware and calibration information, or unperformed calibration.）

・LENGTH Check （Improper calibration information）

・SUM Check （Improper calibration information）

・MODE Check （Improper calibration information）

・FLASH ROM ERROR （Calibration failure due to Flash ROM error）

No.	Error code	Detail of the error	Remark
1	0x21	Version Error	
2	0x22	Length Error	
3	0x23	Sum Error	
4	0x24	Mode Error	
5	0x25	Flash Rom Error	only when calibration failed

*If an error was found in calibration region, calibration would be retried.

If calibration was not performed properly, touch detection would not work properly.

5-2. Raw Data

Electrodes on touch screen will be scanned, and values of the raw data will be checked.

Judged as error If the detected values of the Raw data are outside the range between -255 and 255.

（False detection may occur if booted when the touch screen is being touched.）

No.	Error code	Error detail	Remark
1	0x31	Raw Data Error	

*It may work almost properly despite of abnormal Raw Data, but coordinates close to the abnormal electrodes will not be detected properly.

6. Change History

Ver0.1 (June 13, 2012)

Provisional specification sheet was issued.

Ver0.2 （July 17, 2012）

3. Coordinate data for Report format Windows7/XP was added.

4-3-3.「Setup」section was added.

4-3-4.「Coordinate output control」section was added.

（Note）　　Normal end of version information acquisition command starts coordinate output

　　　→　Issuing coordinate output control command starts coordinate output.

Ver1.0 （July 23, 2012）

First edition

Changes from the ver0.3 are as below.

2.connection method "Interface：RS232C" was deleted.

3-1 "Coordinate data for Windows7 USB" was deleted.

3-2. "Coordinate data for WindowsXP USB" was deleted.

4-2. Command List　　Unnecessary descriptions were deleted.

4-2. Command List　　Coordinate output control command was added.

4-3-4. Coordinate output control「After sending the coordinate output start command, coordinate will be outputted at touch on the touch screen.」was added.

4-4-1. Version information acquisition TEST =xx　　Self-diagnostic result was added.

5. Self-diagnostic test section was added.

Document number was assigned.

Ver1.1 （October 1, 2012）

3-1　Report　format

Explanation for "finger ID" was added.

4-2 Command List

"Usage example of command" was added.

4-4-1. Version information acquisition

Examples of responses were added.

Ver1.2 （June 19, 2013）

4-2 Command List

"Adjust Offset" and "Calibration Offset" were added.

4-3-5. Offset Calibration

A usage example of the offset calibration was added.

Ver1.3 （January 20, 2014.）

Title of this sheet was changed to [DUS Series Software Specifications].

3-3. Coordinate data for Windows 8/7 was added.

    Commands for USB/UART became the same.

    Notes were added P5, P7

4-4-2. Detail information of firmware and response commands were added.

5. Commands for self-diagnostic test were added.

7. Precautions for Use

§   The contents of this document are subject to change without notice.

§   The manufacturer or sales representatives will not be liable for any damages or loss arising from use of this product.

§   This product is intended for use in standard applications (computers, office automation, and other office equipment, industrial, communications, and measurement equipment, personal and household devices, etc.) Please avoid using this product for special applications where failure or abnormal operation may directly affect human lives, or cause physical injury or property damage, or where extremely high levels of reliability are required (such as aerospace systems, vehicle operating control, atomic energy controls, medical devices for life support, etc.).

§   Any semiconductor devices have inherently a certain rate of failure. The user must protect against injury, damage, or loss from such failures by incorporating safety design measures into the user's facility and equipment.