『Maple Bus 1.0』
Function Type Specifications
FT_9: Pointing Function

Revision 0.91

Created by:
CS Development and Product
SEGA ENTERPRISES Ltd.

Revision:

1999/04/09	0.10	Preview Version
1999/10/19	0.50	Preview Version 2 Substantial changes
1999/11/12	0.60	Preview Version 3 Data format changes
1999/11/18	0.70	Preview Version 4 Changes in 10-bit description
1999/11/30	0.80	First Edition
1999/12/21	0.81	Introduced categorization; substantial changes
2000/03/02	0.90	Button names changed
2000/03/23	0.91	Modified the accuracy value of the mouse coordinate axes to 10 bit

Contents:

1 OVERVIEW OF POINTING FUNCTION....4
1.1 Definition of pointing function....4
1.2 Characteristics and limitations of the pointing function....4
1.3 Default configuration for the pointing function....4
2 DEVICE ID....5
2.1 Configuration of the device ID....5
2.2 Function types....6
2.3 Function definition block....6
3 POINTING FUNCTION SPECIFICATIONS....9
3.1 Analog coordinate axes....9
3.1.1 Assigning coordinate axes....10
3.1.2 Absolute coordinates....11
3.1.3 Relative coordinates....12
3.2 Operating buttons....13
3.3 Categories....14
3.3.1 Mouse....14
3.3.2 Tablet....16
4 DATA FORMAT....18
4.1 Basic data format....18
4.2 Notes on coordinate values with 10-bit accuracy....20
5 COMMANDS....21
5.1 Control commands....21
5.1.1 Get Condition....21
5.1.2 Data Transfer....22
5.1.3 Device Reply....23
5.1.4 Device Request....23
5.1.5 All Status Request....24
5.1.6 Device Reset....24
5.1.7 Device Kill....25
5.1.8 Device All Status....25
5.1.9 Device Status....26
5.2 Error commands....27
5.2.1 Function Type Unknown....27
5.2.2 Command Unknown....27
5.2.3 Transmit Again....28
6 PROTOCOL FLOW....29
6.1 Processing flow for data reading....29
7 AFTERWORD....30

1 OVERVIEW OF POINTING FUNCTION

1.1 Definition of pointing function

A pointing function must provide a pointing device capable of indicating a position, and meet the following requirements.

(1) Provided the category limits are not exceeded, there are no restrictions on external appearance.
(2) There must be at least one coordinate axis (digital or analog) on which a point can be identified.
(3) The pointing function must conform to the "Maple Bus 1.0" Standard Specifications.

1.2 Characteristics and limitations of the pointing function

The pointing function has the following characteristics and limitations.

(1) The coordinate axes on which a point can be identified have digital (top-bottom-left-right) specification and analog specification, and either or both can be used.
(2) A pointing device must belong to one or other category within the function.
The categories are determined by the following specifications.
・Analog coordinate axis accuracy: 8-bit (256 levels) or 10-bit (1024 levels).
・Number of analog coordinate axes: Maximum 8 axes can be used.
・Analog coordinate axis system: 1) Absolute coordinates: indicate distance from origin.
2) Relative coordinates: indicate amount of movement from origin.
・External appearance and functionality such as to enable categories to be reasonably distinguished.
(3) Maximum of eight operating buttons can be used, including digital coordinate axes.
(4) Option parameters for wireless type, adapter type, and so on.

1.3 Default configuration for the pointing function

The default configuration for the pointing function is as follows.

(1) The coordinate values are at the origin.
(2) Option parameters are returned with their values at the time.

2 DEVICE ID

Device IDs conform with the device ID stipulations in the "Maple Bus 1.0" Standard Specifications.
The notation used is the memory image in the host.

2.1 Configuration of the device ID

In Maple Bus 1.0, device IDs are configured as shown below.

bit	7	6	5	4	3	2	1	0
1st Data	FT_{31}	FT_{30}	FT_{29}	FT_{28}	FT_{27}	FT_{26}	FT_{25}	FT_{24}
2nd Data	FT_{23}	FT_{22}	FT_{21}	FT_{20}	FT_{19}	FT_{18}	FT_{17}	FT_{16}
3rd Data	FT_{15}	FT_{14}	FT_{13}	FT_{12}	FT_{11}	FT_{10}	FT_9	FT_8
4th Data	FT_7	FT_6	FT_5	FT_4	FT_3	FT_2	FT_1	FT_0
5th Data	$FD1_{31}$	$FD1_{30}$	$FD1_{29}$	$FD1_{28}$	$FD1_{27}$	$FD1_{26}$	$FD1_{25}$	$FD1_{24}$
6th Data	$FD1_{23}$	$FD1_{22}$	$FD1_{21}$	$FD1_{20}$	$FD1_{19}$	$FD1_{18}$	$FD1_{17}$	$FD1_{16}$
7th Data	$FD1_{15}$	$FD1_{14}$	$FD1_{13}$	$FD1_{12}$	$FD1_{11}$	$FD1_{10}$	$FD1_9$	$FD1_8$
8th Data	$FD1_7$	$FD1_6$	$FD1_5$	$FD1_4$	$FD1_3$	$FD1_2$	$FD1_1$	$FD1_0$
9th Data	$FD2_{31}$	$FD2_{30}$	$FD2_{29}$	$FD2_{28}$	$FD2_{27}$	$FD2_{26}$	$FD2_{25}$	$FD2_{24}$
10th Data	$FD2_{23}$	$FD2_{22}$	$FD2_{21}$	$FD2_{20}$	$FD2_{19}$	$FD2_{18}$	$FD2_{17}$	$FD2_{16}$
11th Data	$FD2_{15}$	$FD2_{14}$	$FD2_{13}$	$FD2_{12}$	$FD2_{11}$	$FD2_{10}$	$FD2_9$	$FD2_8$
12th Data	$FD2_7$	$FD2_6$	$FD2_5$	$FD2_4$	$FD2_3$	$FD2_2$	$FD2_1$	$FD2_0$
13th Data	$FD3_{31}$	$FD3_{30}$	$FD3_{29}$	$FD3_{28}$	$FD3_{27}$	$FD3_{26}$	$FD3_{25}$	$FD3_{24}$
14th Data	$FD3_{23}$	$FD3_{22}$	$FD3_{21}$	$FD3_{20}$	$FD3_{19}$	$FD3_{18}$	$FD3_{17}$	$FD3_{16}$
15th Data	$FD3_{15}$	$FD3_{14}$	$FD3_{13}$	$FD3_{12}$	$FD3_{11}$	$FD3_{10}$	$FD3_9$	$FD3_8$
16th Data	$FD3_7$	$FD3_6$	$FD3_5$	$FD3_4$	$FD3_3$	$FD3_2$	$FD3_1$	$FD3_0$

Table 2.1 Device IDs

FT : Function type the peripheral is equipped with.
FD1 : 1st function definition block.
FD2 : 2nd function definition block.
FD3 : 3rd function definition block.

(1) FT_{31} to FT_0 : Function type
Defines the type of function the peripheral is equipped with.
There are 32 function types altogether.

(2) FD_{31} to FD_0 : Function definition block
These blocks define the individual elements constituting the function.
(1 peripheral can be equipped with 3 different functions)

2.2 Function types

This section describes the function type FT within the device ID.

The Pointing Function function type is defined by FT_9 = '1.'

bit	7	6	5	4	3	2	1	0
1st Data	FT_{31}	FT_{30}	FT_{29}	FT_{28}	FT_{27}	FT_{26}	FT_{25}	FT_{24}
2nd Data	FT_{23}	FT_{22}	FT_{21}	FT_{20}	FT_{19}	FT_{18}	FT_{17}	FT_{16}
3rd Data	FT_{15}	FT_{14}	FT_{13}	FT_{12}	FT_{11}	FT_{10}	1	FT_8
4th Data	FT_7	FT_6	FT_5	FT_4	FT_3	FT_2	FT_1	FT_0

Table 2.2 Function Type for the Pointing Function

For example, in the case of a peripheral device for which only the Pointing Function is implemented, the function type is defined by FT = 00-00-02-00h.

If other functions are implemented in a peripheral device, the function type bit that corresponds to that function is set to '1.'

2.3 Function definition block

This section describes the function definition block (FD) within the device ID.

The function definition block is a 32-bit data table that is inherent to each function. The elements that comprise a function, the data transmission and reception methods, etc., are all determined on the basis of this data.

The following table shows the configuration of the function definition block for the Pointing Function.

bit	7	6	5	4	3	2	1	0
1st Data	0	0	0	0	CTC_3	CTC_2	CTC_1	CTC_0
2nd Data	OBD_7	OBD_6	OBD_5	OBD_4	OBD_3	OBD_2	OBD_1	OBD_0
3rd Data	AS_7	AS_6	AS_5	AS_4	AS_3	AS_2	AS_1	AS_0
4th Data	FD_7	FD_6	FD_5	FD_4	FD_3	FD_2	FD_1	FD_0

Table 2.3 Pointing Function Definition Block Configuration

CTC : Category code

Indicates the category to which the pointing device belongs.

There are 16 categories.

Category	CTC	CTC_3	CTC_2	CTC_1	CTC_0
Mouse	0h	0	0	0	0
Tablet	1h	0	0	0	1
Undefined (Reserved)	2h	0	0	1	0
Undefined (Reserved)	3h	0	0	1	1
Undefined (Reserved)	4h	0	1	0	0
Undefined (Reserved)	5h	0	1	0	1
Undefined (Reserved)	6h	0	1	1	0
Undefined (Reserved)	7h	0	1	1	1
Undefined (Reserved)	8h	1	0	0	0
Undefined (Reserved)	9h	1	0	0	1
Undefined (Reserved)	Ah	1	0	1	0
Undefined (Reserved)	Bh	1	0	1	1
Undefined (Reserved)	Ch	1	1	0	0
Undefined (Reserved)	Dh	1	1	0	1
Undefined (Reserved)	Eh	1	1	1	0
Undefined (Reserved)	Fh	1	1	1	1

Table 2.4 Bit definitions for category codes

The maximum number of coordinate axes, the coordinate axis accuracy, and the coordinate system depends on the category used.

OBD : Operating button definition

For each button, this indicates whether it is used or unused.

OBD	OBD_7	OBD_6	OBD_5	OBD_4	OBD_3	OBD_2	OBD_1	OBD_0
Button	R	L	D	U	W	A	B	C
Used	1	1	1	1	1	1	1	1
Unused	0	0	0	0	0	0	0	0

Table 2.5 Bit definitions for operating buttons

- Digital coordinate axes
  - R : Indicates rightward button.
  - L : Indicates leftward button.
  - D : Indicates downward button.
  - U : Indicates upward button.
- Buttons
  - W : Indicates W button. (wheel button: center)
  - A : Indicates A button. (left button: OK)
  - B : Indicates B button. (right button: Cancel)
  - C : Indicates C button.

AS : Analog coordinate axis definition

Indicates whether each analog coordinate axis is used or unused.

AS	AS_7	AS_6	AS_5	AS_4	AS_3	AS_2	AS_1	AS_0
Analog coordinate axis	AC8	AC7	AC6	AC5	AC4	AC3	AC2	AC1
Not present	0	0	0	0	0	0	0	0
Present	1	1	1	1	1	1	1	1

Table 2.6 Definition of analog coordinate axes

The bit for each analog coordinate axis that is used is ‘1.’

FD : Reserved

This value is set to ‘0.’

3 POINTING FUNCTION SPECIFICATIONS

This section describes the elements of the pointing function.
For the coordinate axes, either or both of the digital coordinate axes and the analog coordinate axes can be used.

3.1 Analog coordinate axes

These are coordinate axes in which the coordinate value changes linearly with the direction.

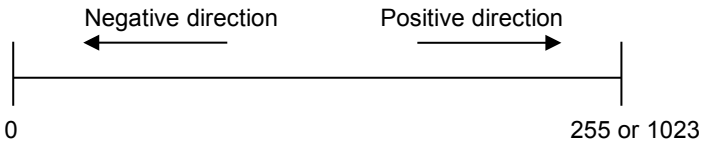

Fig. 3.7 Analog coordinate axes

The coordinate axes can be shown with the following two levels of accuracy (coordinate values).

1) 8-bit accuracy : 0 to 255, origin 128 or 0
2) 10-bit accuracy : 0 to 1023, origin 512 or 0

The points can be indicated with the following two coordinate axis systems.

1) Absolute coordinates: indicate distance from origin.
2) Relative coordinates: indicate amount of movement and direction in each time interval. There is no origin zero setting.

Indicating the coordinates for one point can be done using one, or two or more analog coordinate axes.

Example: P(X,Y), P(X,Y,Z)

By combining the coordinate axes, within one function a plurality of points can be represented.

Example: P1(X1), P2(X2,Y2,Z2)

For the coordinate values of a single point, different accuracies can be used together.

Example: P(X,Y,Z) X,Y = 8-bit accuracy, Y = 10-bit accuracy

For the coordinate axes of a single point, different coordinate systems can be used together.

Example: P(X,Y,Z,U,V,W) X,Y,Z = absolute coordinates, U,V,W = relative coordinates

Also, the coordinate axes of a single point, different coordinate systems and accuracies can be used together.

Example: P(X,Y,Z,U,V,W) X,Y,Z = absolute coordinates, U,V,W = relative coordinates, X,Y = 10-bit accuracy, Z,U,V,W = 8-bit accuracy

For the correspondence between indicated points and coordinate axes and origin, refer to the category specification and the specification for the corresponding peripheral device.

3.1.1 Assigning coordinate axes

There can be a maximum of eight coordinate axes (AC1 to AC8).

The function definition block (AS) contains the indication of which analog coordinate axes can be used.

The data sizes which can be used for 8-bit and 10-bit accuracy coordinate axes are as follows.

Analog coordinate axis No.	Analog coordinate axis	10-bit accuracy coordinate axis data size	8-bit accuracy coordinate axis data size
No.1	AC1	2 bytes	1 byte
No.2	AC2	2 bytes	1 byte
No.3	AC3	2 bytes	1 byte
No.4	AC4	2 bytes	1 byte
No.5	AC5	2 bytes	1 byte
No.6	AC6	2 bytes	1 byte
No.7	AC7	2 bytes	1 byte
No.8	AC8	2 bytes	1 byte

Table 3.8 Data sizes for analog coordinate axes

The coordinate axis accuracy, the number of coordinate axes, and the correspondence with analog coordinate axes are determined by the category.

3.1.2 Absolute coordinates

The coordinates of a specified point P show the distance from the origin.
The coordinate values are following below:
For 10-bit accuracy : 000h to 3FFh, origin 200h or 0h
For 8-bit accuracy : 00h to FFh, origin 80h or 0h

* The coordinate determination does not depend on the timing of reading of the coordinates by the host.

For example, in the following figure, the coordinates when the point P moves are 200h at P_0, next 2CAh at P_1, and next 194h at P_2.

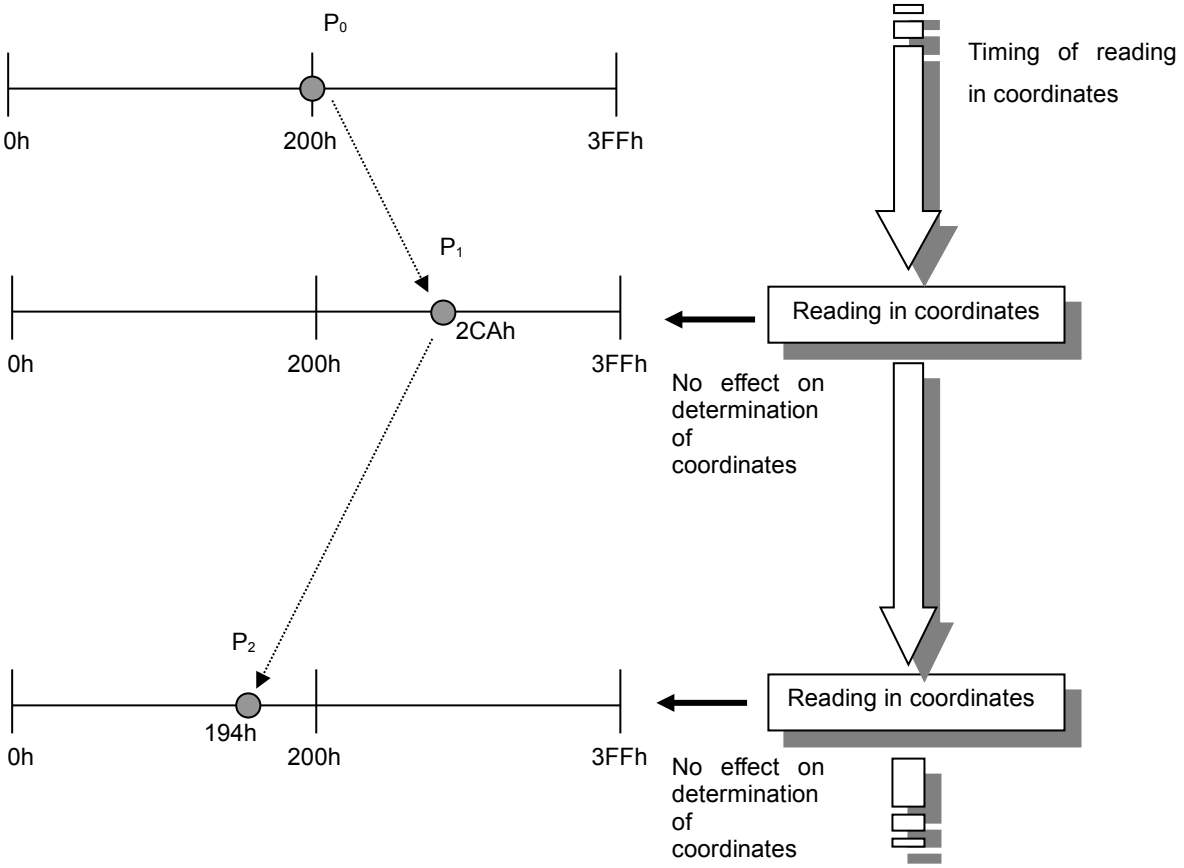

Fig. 3.9 Absolute coordinates

3.1.3 Relative coordinates

The coordinates of a specified point P show the movement relative to the last point as origin.

The coordinate values are following below:

For 10-bit accuracy : 000h to 3FFh, origin 200h

For 8-bit accuracy : 00h to FFh, origin 80h

The direction is given as positive when the coordinates are greater than the origin value, and negative when less.

* The coordinate determination depends on the timing of reading of the coordinates by the host.
  The host must therefore read the coordinates at a fixed interval.

For example, in the following figure, the coordinates when the point P moves are 200h at P_0, next 2CAh at P_1, and next BDh at P_2.

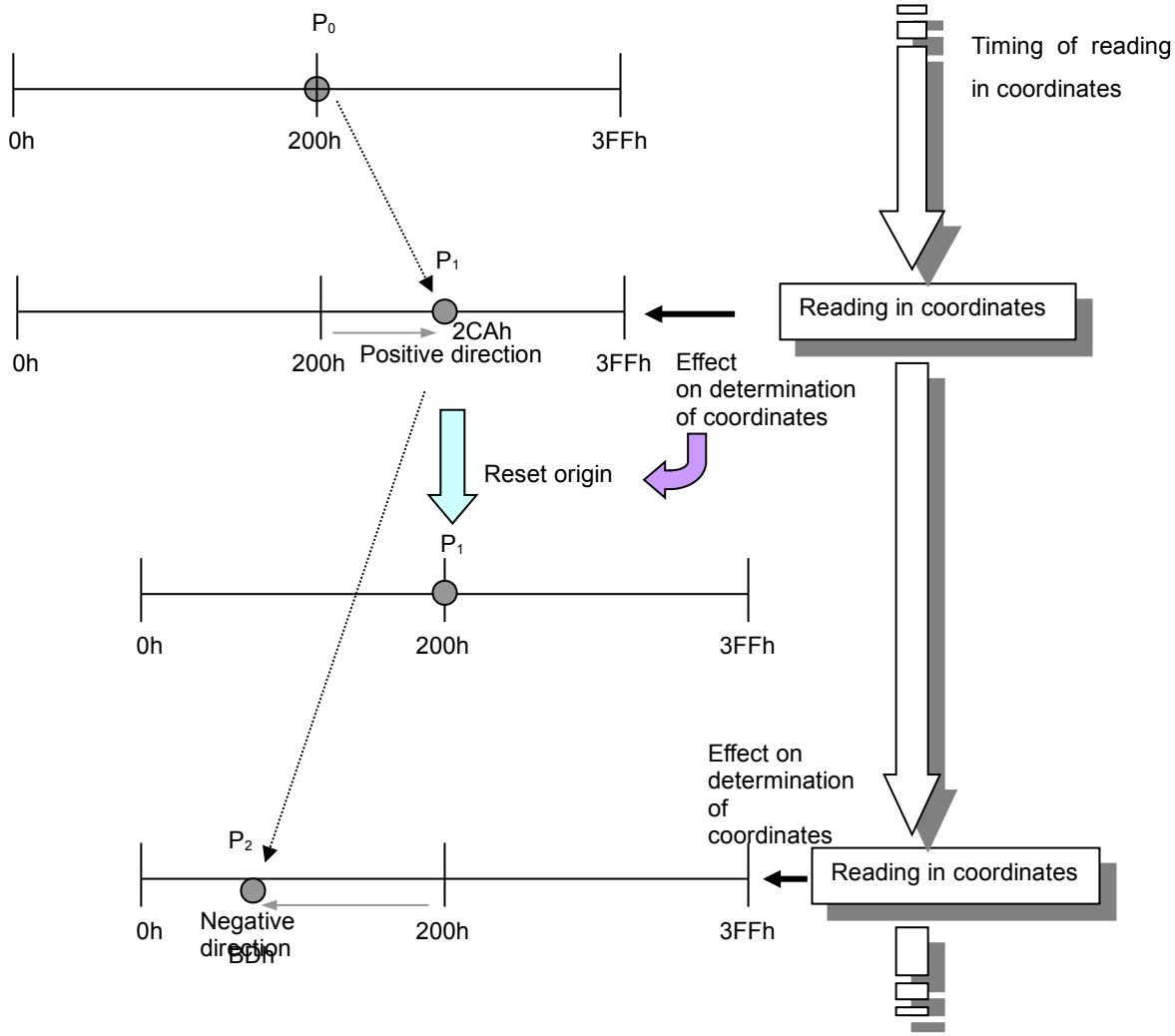

Fig. 3.10 Relative coordinates

3.2 Operating buttons

This section describes the button (switches) for operation on an indicated coordinate point.

The function definition block (OBD) shows which buttons can be used.

bit	7	6	5	4	3	2	1	0
BTN	R	L	D	U	W	A	B	C
OFF	1	1	1	1	1	1	1	1
ON	0	0	0	0	0	0	0	0

Table 3.11 Button data

Digital coordinate axes

R : Indicates rightward button.

L : Indicates leftward button.

D : Indicates downward button.

U : Indicates upward button.

Buttons

W : Indicates wheel button.

A : Indicates A button.

B : Indicates B button.

C : Indicates C button.

When a button is pressed it is ON, and when it is not pressed it is OFF.

The function arranges that R and L or D and U are not ON simultaneously. (As standard)

Operation indications (OK: left button) principally use the A button, and operation cancel (Cancel: right button) principally uses the B button.

3.3 Categories

A pointing device belongs to one of 16 categories.
The applications, and general appearance and functionality depend on the category.
The categories are now described.

3.3.1 Mouse

* A mouse must have AC1 and AC2 (x- and y-axes), and the A button and B button.

A mouse includes a mouse and a device whose operation is compatible with that of a mouse.
This is shown by CTC = 0h.

Number of analog coordinate axes : Maximum eight can be used
Coordinate axis accuracy : All 8-bit accuracy
Coordinate systems : Relative coordinate systems

・Mouse data format

bit	7	6	5	4	3	2	1	0
1st-Data	R	L	D	U	S	A	B	C
2nd-Data	OP_7	OP_6	OP_5	OP_4	OP_3	OP_2	OP_1	OP_0
3th-Data	AOV_7	AOV_6	AOV_5	AOV_4	AOV_3	AOV_2	AOV_1	AOV_0
4rd-Data	RES_7	RES_6	RES_5	RES_4	RES_3	RES_2	RES_1	RES_0
5th-Data	$AC1_7$	$AC1_6$	$AC1_5$	$AC1_4$	$AC1_3$	$AC1_2$	$AC1_1$	$AC1_0$
6th-Data	$AC2_7$	$AC2_6$	$AC2_5$	$AC2_4$	$AC2_3$	$AC2_2$	$AC2_1$	$AC2_0$
7th-Data	$AC3_7$	$AC3_6$	$AC3_5$	$AC3_4$	$AC3_3$	$AC3_2$	$AC3_1$	$AC3_0$
8th-Data	$AC4_7$	$AC4_6$	$AC4_5$	$AC4_4$	$AC4_3$	$AC4_2$	$AC4_1$	$AC4_0$
9th-Data	$AC5_7$	$AC5_6$	$AC5_5$	$AC5_4$	$AC5_3$	$AC5_2$	$AC5_1$	$AC5_0$
10th-Data	$AC6_7$	$AC6_6$	$AC6_5$	$AC6_4$	$AC6_3$	$AC6_2$	$AC6_1$	$AC6_0$
11th-Data	$AC7_7$	$AC7_6$	$AC7_5$	$AC7_4$	$AC7_3$	$AC7_2$	$AC7_1$	$AC7_0$
12th-Data	$AC8_7$	$AC8_6$	$AC8_5$	$AC8_4$	$AC8_3$	$AC8_2$	$AC8_1$	$AC8_0$
13th-Data	$AC5_7$	$AC5_6$	$AC5_5$	$AC5_4$	$AC5_3$	$AC5_2$	$AC5_1$	$AC5_0$
14th-Data	$AC5_{15}$	$AC5_{14}$	$AC5_{13}$	$AC5_{12}$	$AC5_{11}$	$AC5_{10}$	$AC5_9$	$AC5_8$
15th-Data	$AC6_7$	$AC6_6$	$AC6_5$	$AC6_4$	$AC6_3$	$AC6_2$	$AC6_1$	$AC6_0$
16th-Data	$AC6_{15}$	$AC6_{14}$	$AC6_{13}$	$AC6_{12}$	$AC6_{11}$	$AC6_{10}$	$AC6_9$	$AC6_8$
17th-Data	$AC7_7$	$AC7_6$	$AC7_5$	$AC7_4$	$AC7_3$	$AC7_2$	$AC7_1$	$AC7_0$
18th-Data	$AC7_{15}$	$AC7_{14}$	$AC7_{13}$	$AC7_{12}$	$AC7_{11}$	$AC7_{10}$	$AC7_9$	$AC7_8$
19th-Data	$AC8_7$	$AC8_6$	$AC8_5$	$AC8_4$	$AC8_3$	$AC8_2$	$AC8_1$	$AC8_0$
20th-Data	$AC8_{15}$	$AC8_{14}$	$AC8_{13}$	$AC8_{12}$	$AC8_{11}$	$AC8_{10}$	$AC8_9$	$AC8_8$

Table 3.12 Mouse data format

1st to 4th-Data: see "4 data format."

5th, 6th-Data	AC1: X axis data
7th, 8th-Data	AC2: Y axis data
9th, 10th-Data	AC3: Z axis data (wheel)
11th, 12th-Data	AC4: Other axis data
13th, 14th-Data	AC5: Other axis data
15th, 16th-Data	AC6: Other axis data
17th, 18th-Data	AC7: Other axis data
19th, 20th-Data	AC8: Other axis data

・Coordinate direction

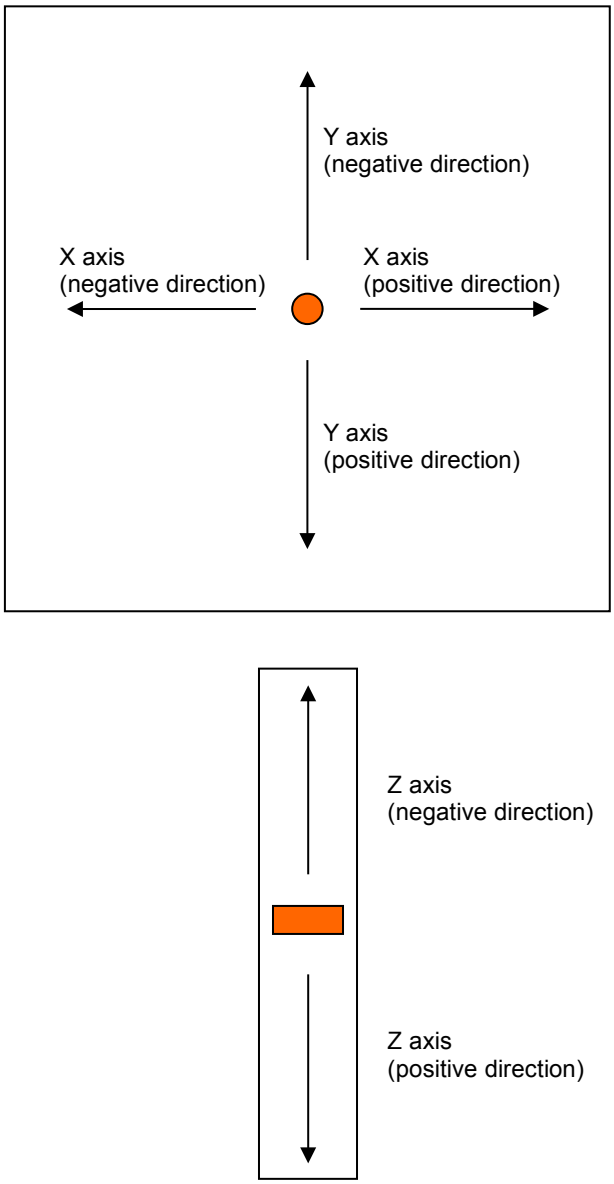

Fig. 3.13 Mouse coordinate direction

3.3.2 Tablet

* A tablet must have AC1 and AC2 (x- and y-axes), and the A button.

A tablet includes a tablet and a device whose operation is compatible with that of a tablet.
This is shown by CTC = 1h.

Number of analog coordinate axes	: Maximum eight can be used
Coordinate axis accuracy	: All 10-bit accuracy
Coordinate systems	: Absolute coordinate systems

・Tablet data format

bit	7	6	5	4	3	2	1	0
1st-Data	R	L	D	U	S	A	B	C
2nd-Data	OP_7	OP_6	OP_5	OP_4	OP_3	OP_2	OP_1	OP_0
3th-Data	AOV_7	AOV_6	AOV_5	AOV_4	AOV_3	AOV_2	AOV_1	AOV_0
4rd-Data	RES_7	RES_6	RES_5	RES_4	RES_3	RES_2	RES_1	RES_0
5th-Data	$AC1_7$	$AC1_6$	$AC1_5$	$AC1_4$	$AC1_3$	$AC1_2$	$AC1_1$	$AC1_0$
6th-Data	$AC1_{15}$	$AC1_{14}$	$AC1_{13}$	$AC1_{12}$	$AC1_{11}$	$AC1_{10}$	$AC1_9$	$AC1_8$
7th-Data	$AC2_7$	$AC2_6$	$AC2_5$	$AC2_4$	$AC2_3$	$AC2_2$	$AC2_1$	$AC2_0$
8th-Data	$AC2_{15}$	$AC2_{14}$	$AC2_{13}$	$AC2_{12}$	$AC2_{11}$	$AC2_{10}$	$AC2_9$	$AC2_8$
9th-Data	$AC3_7$	$AC3_6$	$AC3_5$	$AC3_4$	$AC3_3$	$AC3_2$	$AC3_1$	$AC3_0$
10th-Data	$AC3_{15}$	$AC3_{14}$	$AC3_{13}$	$AC3_{12}$	$AC3_{11}$	$AC3_{10}$	$AC3_9$	$AC3_8$
11th-Data	$AC4_7$	$AC4_6$	$AC4_5$	$AC4_4$	$AC4_3$	$AC4_2$	$AC4_1$	$AC4_0$
12th-Data	$AC4_{15}$	$AC4_{14}$	$AC4_{13}$	$AC4_{12}$	$AC4_{11}$	$AC4_{10}$	$AC4_9$	$AC4_8$
13th-Data	$AC5_7$	$AC5_6$	$AC5_5$	$AC5_4$	$AC5_3$	$AC5_2$	$AC5_1$	$AC5_0$
14th-Data	$AC5_{15}$	$AC5_{14}$	$AC5_{13}$	$AC5_{12}$	$AC5_{11}$	$AC5_{10}$	$AC5_9$	$AC5_8$
15th-Data	$AC6_7$	$AC6_6$	$AC6_5$	$AC6_4$	$AC6_3$	$AC6_2$	$AC6_1$	$AC6_0$
16th-Data	$AC6_{15}$	$AC6_{14}$	$AC6_{13}$	$AC6_{12}$	$AC6_{11}$	$AC6_{10}$	$AC6_9$	$AC6_8$
17th-Data	$AC7_7$	$AC7_6$	$AC7_5$	$AC7_4$	$AC7_3$	$AC7_2$	$AC7_1$	$AC7_0$
18th-Data	$AC7_{15}$	$AC7_{14}$	$AC7_{13}$	$AC7_{12}$	$AC7_{11}$	$AC7_{10}$	$AC7_9$	$AC7_8$
19th-Data	$AC8_7$	$AC8_6$	$AC8_5$	$AC8_4$	$AC8_3$	$AC8_2$	$AC8_1$	$AC8_0$
20th-Data	$AC8_{15}$	$AC8_{14}$	$AC8_{13}$	$AC8_{12}$	$AC8_{11}$	$AC8_{10}$	$AC8_9$	$AC8_8$

Table 3.14 Tablet data format

1st to 4th-Data: see "4 data format."

5th, 6th-Data	AC1: X axis data
7th, 8th-Data	AC2: Y axis data
9th, 10th-Data	AC3: Other axis data
11th, 12th-Data	AC4: Other axis data
13th, 14th-Data	AC5: Other axis data
15th, 16th-Data	AC6: Other axis data
17th, 18th-Data	AC7: Other axis data
19th, 20th-Data	AC8: Other axis data

・Coordinate direction

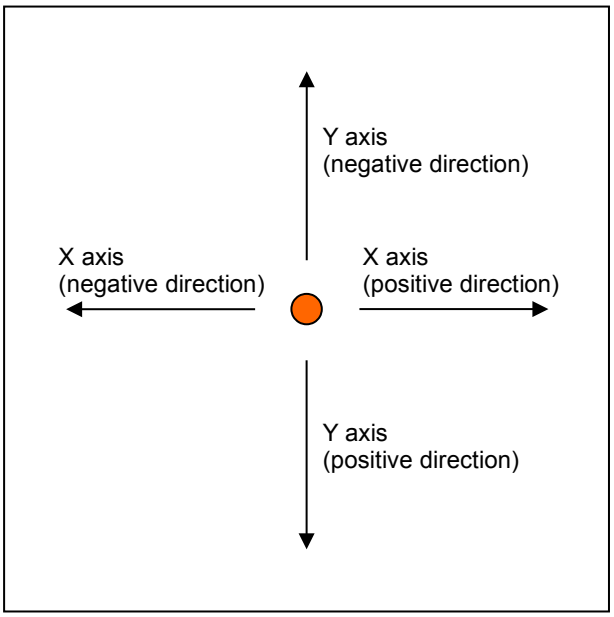

Fig. 3.15 Tablet coordinate direction

4 DATA FORMAT

This section describes the Pointing Function's data format.

4.1 Basic data format

This describes the basic data format for each category.

bit	7	6	5	4	3	2	1	0
1st-Data	R	L	D	U	S	A	B	C
2nd-Data	OP_7	OP_6	OP_5	OP_4	OP_3	OP_2	OP_1	OP_0
3th-Data	AOV_7	AOV_6	AOV_5	AOV_4	AOV_3	AOV_2	AOV_1	AOV_0
4rd-Data	RES_7	RES_6	RES_5	RES_4	RES_3	RES_2	RES_1	RES_0
5th-Data	AC	AC	AC	AC	AC	AC	AC	AC
6th-Data	AC	AC	AC	AC	AC	AC	AC	AC
7th-Data	AC	AC	AC	AC	AC	AC	AC	AC
8th-Data	AC	AC	AC	AC	AC	AC	AC	AC
9th-Data	AC	AC	AC	AC	AC	AC	AC	AC
10th-Data	AC	AC	AC	AC	AC	AC	AC	AC
11th-Data	AC	AC	AC	AC	AC	AC	AC	AC
12th-Data	AC	AC	AC	AC	AC	AC	AC	AC
13th-Data	AC	AC	AC	AC	AC	AC	AC	AC
14th-Data	AC	AC	AC	AC	AC	AC	AC	AC
15th-Data	AC	AC	AC	AC	AC	AC	AC	AC
16th-Data	AC	AC	AC	AC	AC	AC	AC	AC
17th-Data	AC	AC	AC	AC	AC	AC	AC	AC
18th-Data	AC	AC	AC	AC	AC	AC	AC	AC
19th-Data	AC	AC	AC	AC	AC	AC	AC	AC
20th-Data	AC	AC	AC	AC	AC	AC	AC	AC

Table 4.16 Basic data format

■Description

1st-Data	: BTN	Operating button data
2nd-Data	: OP	Option parameters
3rd-Data	: AOV	Analog coordinate overflow
4th-Data	: RES	Reserved
5th to 20th-Data	: AC	Analog coordinate axis regions determined by category

The data extends as far as the 20th-Data, but the last data No. may differ depending on the category.

BTN: Operating button data

bit	7	6	5	4	3	2	1	0
BTN	R	L	D	U	S	A	B	C
OFF	1	1	1	1	1	1	1	1
ON	0	0	0	0	0	0	0	0

Table 4.17 Operating button data

For details, see Section 3.2, "Operating buttons."

OP: Option parameters

bit	7	6	5	4	3	2	1	0
OP	OP_7	OP_6	OP_5	OP_4	OP_3	OP_2	OP_1	OP_0
Parameter	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	BATT	WIRE

Table 4.18 Option parameters

Reserved: This value is set to '0.'

BATT: For a wireless pointer or similar, this shows the state of the power source battery.

'0': No problem with battery capacity, or not a wireless pointer with internal power source.

'1': Battery capacity low, warning to stop operation.

At this point the coordinate data is not reliable.

WIRE: For a disconnectable pointer, this shows the connection state.

'0': Pointer is connected, or is not disconnectable.

'1': Pointer is disconnected.

At this point the coordinate data is not reliable.

AOV: coordinate data overflow

Bit	7	6	5	4	3	2	1	0
AOV	AOV_7	AOV_6	AOV_5	AOV_4	AOV_3	AOV_2	AOV_1	AOV_0
Analog coordinate axes	AC8	AC7	AC6	AC5	AC4	AC3	AC2	AC1
Overflow	1	1	1	1	1	1	1	1
Data valid	0	0	0	0	0	0	0	0

Table 4.19 Point data overflow

The bits of AOV indicate for the corresponding analog coordinate axes, whether the coordinate data assigned to an axis exceeds the maximum value for the accuracy (i.e. has overflowed); when overflow occurs the bit is 1.
At this point the coordinate value is indicated as the maximum (minimum) value.
When overflow has not occurred, the bit is 0.

RES: Reserved
Set to 0.

AC: Analog coordinate axis regions determined by category

For details, see the items for each category.
For analog axes which are not used, the origin coordinates of the specified coordinate system are set.

4.2 Notes on coordinate values with 10-bit accuracy

The coordinate values are split into the top two bits and the bottom eight bits.
The data is stored in little-endian format, as follows.

Bit	9	8	7	6	5	4	3	2	1	0
10 bit data	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

↓

Data format	Bit	7	6	5	4	3	2	1	0
n th-Data	Bottom 10 bits	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
(n+1) th-Data	Top 2 bits	0	0	0	0	0	0	D_9	D_8

n: odd

Table 4.20 Storage format of 10-bit accuracy coordinate values

5 COMMANDS

This chapter describes the commands specified in the "Maple Bus 1.0" Standard Specifications which are supported by

the Pointing function.

All of the setting examples assume that the Pointing Function is connected to port A.

5.1 Control commands

5.1.1 Get Condition

Issuing right : Host
Command code : 09h
Data size : 01h
Data field : Function type 4 Bytes
Expected return value : [Data Transfer]
Description : Requests the physical state of the function.
This is used to get the current displacement in the pointing function.

Data Address	Data	Example setting	Description
+0000h	Command code	09h	Specifies [Get Condition]
+0001h	Desitination AP	20h	Device
+0002h	Origin AP	00h	Port A
+0003h	Data size	01h	The data size is 4 bytes
+0004h	Function type	00h	Specifies the function type as “Pointing”
+0005h		00h	
+0006h		02h	
+0007h		00h	

Table 5.21 Get Condition Command Example

The data read with [Get Condition] is transferred with [Data Transfer].

This is the data format in the pointing function.

5.1.2 Data Transfer

Issuing right : Pointing Function
Command code : 08h
Data size : 06h(24byte)
Data : Function type 4 bytes
Data 20 bytes
Description : This command returns data in response to a request from the host.
The pointing data is returned in the data format.
The data size depends on the category.

Data Address	Data	Example setting	Description
+0000h	Command code	08h	Specifies [Data Transfer]
+0001h	Desitination AP	00h	Port A
+0002h	Origin AP	20h	Device
+0003h	Data size	06h	The data size is 24 bytes
+0004h	Function type	00h	Specifies the function type as “Pointing”
+0005h		00h	
+0006h		02h	
+0007h		00h	
+0008h	BTN	FFh	Data format
+0009h	OP	00h	
+000Ah	AOV	00h	
+000Bh	RES	00h	
+000Ch	AC		
+000Dh			
+000Eh			
+000Fh			
+0010h			
+0011h			
+0012h			
+0013h			
+0014h			
+0015h			
+0016h			
+0017h			
+0018h			
+0019h			
+001Ah			
+001Bh			

Table 5.22 Data Transfer Command Example

For the read format, see "4 data format."

5.1.3 Device Reply

Issuing right : Pointing Function
Command code : 07h
Data size : 00h
Data : none
Description : This command is returned to the host as the reply command if the command that the host sent was processed normally by the Pointing Function.
An example of this command is shown below.

Data Address	Data	Example setting	Description
+0000h	Command code	07h	Specifies [Device Reply]
+0001h	Desitination AP	00h	Port A
+0002h	Origin AP	20h	Device
+0003h	Data size	00h	The data size is 0 byte

Table 5.23 Device Reply Command Example

5.1.4 Device Request

Issuing right : Host
Command code : 01h
Data size : 00h
Data : none
Expected return value : [Device Status]
Description : This command requests [Device Status] from the desitination AP peripheral device.
After initialization, the Pointing Function does not respond to any other commands until it receives this command.
An example of this command is shown below.

Data Address	Data	Example setting	Description
+0000h	Command code	01h	Specifies [Device Request]
+0001h	Destination AP	20h	Device
+0002h	Origin AP	00h	Port A
+0003h	Data size	00h	The data size is 0 byte

Table 5.24 Device Request Command Example

5.1.5 All Status Request

Issuing right : Host
Command code : 02h
Data size : 00h
Data field : none
Expected return value : [Device All Status]
Description : This command requests all device statuses from the destination AP peripheral device. (both Fixed Device Status and Free Device Status)
An example of this command is shown below.

Data Address	Data	Example setting	Description
+0000h	Command code	02h	Specifies [All Status Request]
+0001h	Destination AP	20h	Device
+0002h	Origin AP	00h	Port A
+0003h	Data size	00h	The data size is 0 byte

Table 5.25 All Status Request Command Example

5.1.6 Device Reset

Issuing right : Host
Command code : 03h
Data size : 00h
Data field : none
Expected return value : [Device Reply]
Order of operation : (1) [Device Reply] returned.
(2) Initialization.
Description : This command enables the device specified by the destination AP to be initialized.
An example of this command is shown below.

Data Address	Data	Example setting	Description
+0000h	Command code	03h	Specifies [Device Reset]
+0001h	Destination AP	20h	Device
+0002h	Origin AP	00h	Port A
+0003h	Data size	00h	The data size is 0 byte

Table 5.26 Device Reset Command Example

5.1.7 Device Kill

Issuing right : Host
Command code : 04h
Data size : 00h
Data field : none
Expected return value : [Device Reply]
Order of operation : (1) [Device Reply] returned.
(2) Operation terminated.
Description : Operation by the peripheral specified by the destination AP is not recognized.
The function stands by in standby power consumption mode, and no commands can be received.
To start operation, a hard reset must be performed, or the power should be turned off and operation then started again.
An example of this command is shown below.

Data Address	Data	Example setting	Description
+0000h	Command code	04h	Specifies [Device Kill]
+0001h	Destination AP	20h	Device
+0002h	Origin AP	00h	Port A
+0003h	Data size	00h	The data size is 0 byte

Table 5.27 Device Kill Command Example

5.1.8 Device All Status

Issuing right : Peripheral device
Command code : 06h
Data size : 1Ch + n/4
Data field : Fixed Device Status: 112 Bytes
Device ID : 16 Bytes
Destination code : 1 Byte
Product name : 31 Bytes
License : 60 Bytes
Standby current consumption : 2 Bytes
Maximum current consumption : 2 Bytes
Free Device Status: n Byte
Description : This command returns both the Fixed Device Status and Free Device Status in response to the [All Status Request] command from the host.

5.1.9 Device Status

Issuing right : Peripheral device
Command code : 05h
Data size : 1Ch
Data field : Device ID : 16 Bytes
Destination code : 1 Byte
Product name : 31 Bytes
License : 60 Bytes
Standby current consumption : 2 Bytes
Maximum current consumption : 2 Bytes
Description : This command returns Fixed Device Status data in response to [Device Request] from the host.
An example of this command is shown below..

Data Address	Data	Example setting	Description
+0000h	Command code	05h	Specifies [Device Status]
+0001h	Desitination AP	00h	Port A
+0002h	Origin AP	20h	Device
+0003h	Data size	1Ch	The data size is 112 bytes
+0004h to +0013h	Device ID		Specifies the device ID
+0014h	Destination code		Specifies the destination code
+0015h to +0033h	Product name		Specifies the ptoduct name
+0034h to +006Fh	License		Specifies the license
+0070h	Standby current consumption		Specifies the standby current consumption
+0071h			
+0072h	Maximum current consumption		Specifies the maximum current consumption
+0073h			

Table 5.28 Device Status Command Example

5.2 Error commands

5.2.1 Function Type Unknown

Issuing right : Peripheral device
Command code : FEh
Data size : 00h
Data field : none
Description : This error command is returned when the function type that was received does not exist forthe peripheral device.
Possible causes : (1) Mistaken specification of function type.
(2) Data is written incorrectly.
(3) Data of device ID is jumbled.
(4) Data became jumbled during communication.
Remedies : (1) Specify command correctly.
(2) Write data correctly.
(3) Resend [Device Request] to obtain device ID.
(4) Send again (maximum of 3 times; subsequent attempts are processed as Time Out).

5.2.2 Command Unknown

Issuing right : Pointing Function
Command code : FDh
Data size : 00h
Data field : none
Description : This command is returned when the pointing function does not support the command sent.
Possible causes : (1) Mistaken specification of function type.
(2) Data is written incorrectly.
(3) Data of device ID is jumbled.
(4) Data became jumbled during communication.
Remedies : (1) Specify command correctly.
(2) Write data correctly.
(3) Resend [Device Request] to obtain device ID.
(4) Send again (maximum of 3 times; subsequent attempts are processed as Time Out).

5.2.3 Transmit Again

Issuing right	: Host, Pointing Function
Command code	: FCh
Data size	: 00h
Data field	: none
Description	: This command is used to request that data be transmitted again when the data contained some kind of error.
Possible causes	: (1) Parity error was generated. (2) Data overflowed. (3) Data became jumbled during communication. (4) Incorrect setting. (5) Others
Remedies	: Send again (maximum of 3 times; subsequent attempts are processed as Time out).
Notes	: For the pointing function, data is always updated by an access from the host. Therefore, it is not possible to return exactly the same data (point coordinate values) as the previous time from the pointing function.

6 PROTOCOL FLOW

The following diagram provides a general overview of the basic communications protocol between the host and the Pointing Function.

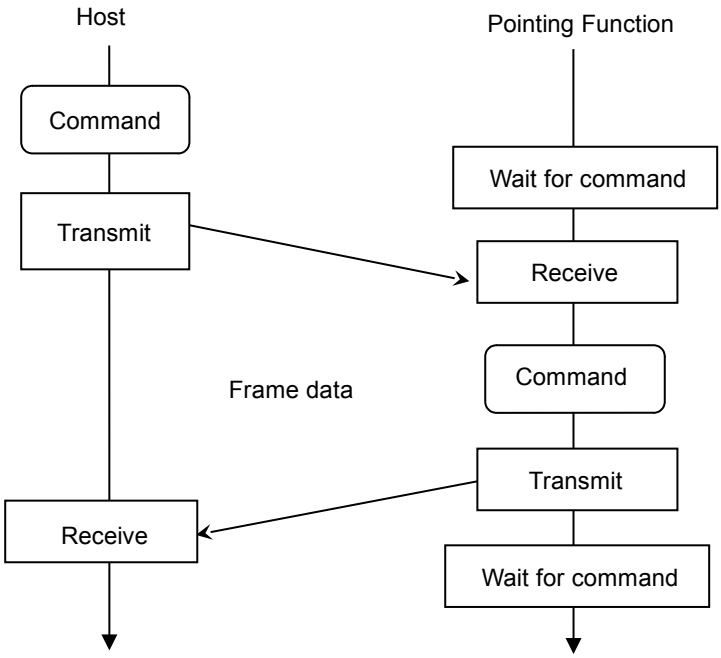

Fig. 6.29 Overview of Communications Protocol

6.1 Processing flow for data reading

The processing flow for reading the point coordinate values (read format) is shown in the following figure.

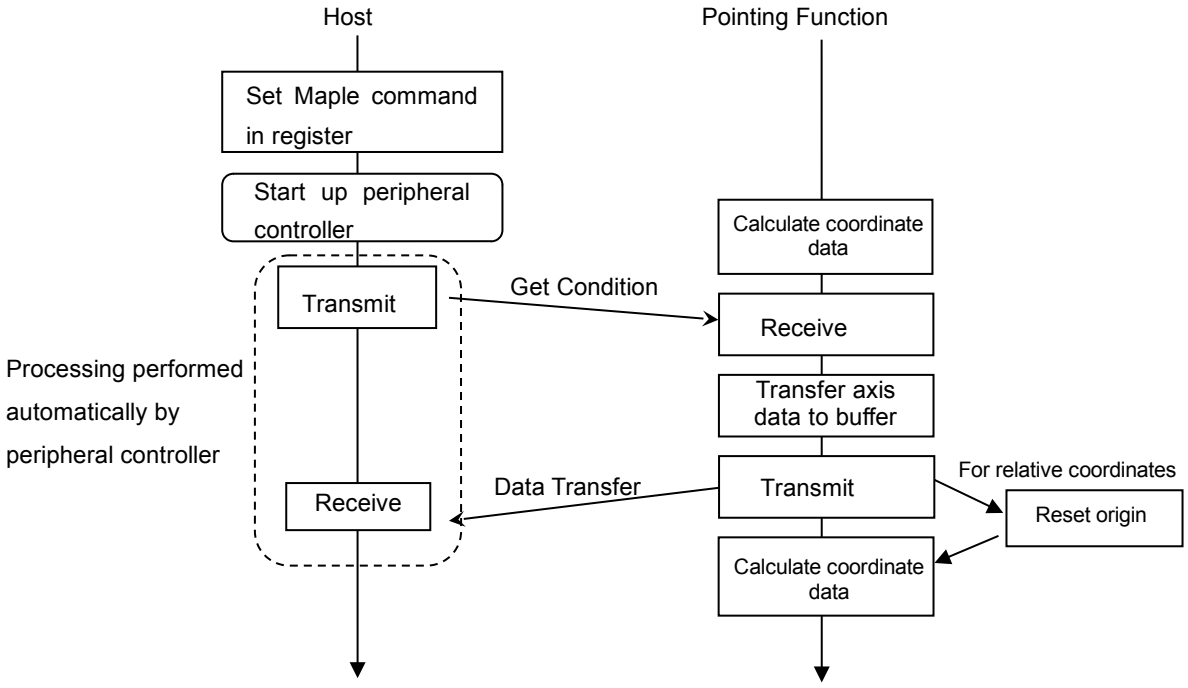

Fig. 6.30 Processing flow for reading coordinate values

7 AFTERWORD

Untl the official version (Rev. 1.0) is distributed, contents will be modified to a small or large extent.