Innovati Joystick 3A Module 3-Dimensional Joystick and Joystick Button



Version: V1.1

Product Overview: Innovati's Joystick 3A module provides simple setup and position reading functions. Users can easily customize Joystick 3A module to fit their own needs. By connecting the Joystick 3A module to Innovati's BASIC Commander through cmdBUS, users can run various customized functions and achieve intuitive control of devices such as robots, robotic arms, etc. Joystick 3A provides two kinds of coordinate systems: rectangular and polar. Depending on their applications, users may switch to any one of the two coordinate systems. In addition to x- and y-axis movement control, Joystick 3A also has twisting control and buttons for complex applications. Please use "Joystick3A" as the module object name in program.

Applications:

- Robotic arm control: controls the rotation angle of the robotic arm with a polar coordinate output.
- Remote control of cars and aircraft with the use of wireless output module.
- > The operation of various testing equipment
- Motor control: controls motor acceleration with the use of a motor module and motor fixed-speed cruising with the use of buttons.
- Control of all Innovati's application packages

Product Features:

- Easy setup: To use the dedicated instructions for various applications, all that is needed is to connect Joystick 3A module to BASIC commander through cmdBUS.
- 3-dimensional movement: In addition to planar movements in the x- and y-axes, Joystick 3A can also twist left or right for control along a third axis.
- > Two return coordinate systems: Joystick 3A provides return values in two

coordinate systems: rectangular and polar. Users can choose either coordinate system at any time or use both simultaneously.

- 4-way and 8-way joystick positions: for fast and intuitive control of various basic control applications
- Large origin range: 0~10% customized origin range, preventing joystick bouncing.
- Return value limit: Users can set limits for all return coordinates and thus limit the joystick operational range.
- Customized resolution: The return coordinate can have at most 128 levels and the polar angle has at most 360 levels. Depending on actual applications, the two resolutions may be set individually.
- Programmable joystick button: There is a programmable button on the top of the stick. Customized commands, including activation time of keystroke repetition and repetition rates, can be set through joystick instructions.
- Easy Calibration: Joystick 3A module comes with a calibration button. The calibration of the joystick can be done at any time during the operation.

Connection: To access Joystick 3A through the BASIC Commander, set the ID switches to the desired number settings and connect cmdBUS to the proper pins on the BASIC Commander.



Product Specifications:

cmdBUS pins: Please use cmdBUS to connect the BASIC Commander. Please mind the pin assignment when connecting. These pins are, from left to right, Vin, GND, SDA, SCL, EVT, and SYN.

EVT light: The switched on green LED represents an event sent by the module.

SDA communication light: The switched on yellow LED represents the module sending data requests to the SBC. Module number switches: Sets the module ID in binary form. The leftmost switch No. 1 is the most significant bit of the module number. The top position is 1 while the bottom position 0. The module number shown in the figure is 31.

Calibration activation button: Pressing down and holding this button for 5 seconds activates the calibration mode. Please note that all instructions are ineffective during calibration. Upon completion of calibration, please press the button on the top of the stick to save the calibration results and exit the calibration mode. To exit without saving, please push again the calibration button. The module will exit the calibration mode immediately.

Calibration activation light: The flickering red LED represents the module in the calibration mode.

Figure 1: Pin assignment and switch description

111111



After entering the calibration mode (when the red LED starts to blink), please push the stick all the way up and move the stick against the edges for 2 turns to get the maximum and minimum displacements of the x- and y-axes. Next, twist the z-axis to the left limit and the right limit, respectively and stay at the limit for 2 seconds for the joystick to record the maximum and minimum z-axis displacement. Finally, move the stick back to the center and let it stand for 3 seconds for the joystick to record the center of the x-, y- and z-axes. Finally, push the button on the top of the stick to exit the calibration mode.

If the calibration mode is accidentally activated, exit the calibration mode by pressing the calibration button.

Incorrect calibration may cause incorrect return values.



Figure 2: Joystick specifications (Unit: mm)





Figure 4: Joystick movements and the polar coordinate system

Symbol	Parameter	Test Conditions		Min	Tun	May	Unit
		VIN	Conditions	I VI III.	Typ.	IVIAX.	UIII
Iin	Operating Current	7.5	_		10.5		mA

Table 1: Rated operating current (at 25°C)

Absolute Maximum Ratings:

The lifetime of the stick is 500,000 twists.

Operating temperature: $-10^{\circ}C \sim 80^{\circ}C$

Storage temperature: $-10^{\circ}C \sim 80^{\circ}C$

Commands and Events:

The following tables list all the unique commands and events provided with the Servo Runner A Module. Note that essential words in the commands will be written in **bold** type and *italics* in bold type. The bold type word must be written exactly as shown, whereas the italic bold type words must be replaced with the user values. Note that the innoBASIC language is case-insensitive.

Command Format	Function of the Command		
Commands for joystick calibrati	on		
	Read the joystick x-axis calibration values. The three return		
CotColliburgtion V(Version Version	values are digitized voltage values. Xmin is the minimal x		
GetCalibrationX(<i>Xmin</i> , <i>Xcen</i> ,	coordinate, <i>Xcen</i> is the center x coordinate, and <i>Xmax</i> is the		
Amax)	maximal x coordinate. <i>Xmin, Xcen</i> and <i>Xmax</i> ranging from		
	0 to 65535.		
	Read the joystick y-axis calibration values. The three return		
C-4C-Phanetics V(Varian Varia	values are digitized voltage values. Ymin is the minimal y		
GetCalibration Y (Ymin, Ycen,	coordinate, Ycen is the center x coordinate, and Ymax is the		
Tmax)	maximal y coordinate. Ymin, Ycen and Ymax ranging from		
	0 to 65535.		
	Read the joystick z-axis calibration value. The three return		
	values are digitized voltage values. Zmin is the minimal z		
GetCalibrationZ(Zmin, Zcen,)	coordinate, Zcen is the center z coordinate, and Zmax is the		
	maximal z coordinate. Zmin, Zcen and Zmax ranging from		
	0 to 65535.		
	Set the x-axis joystick calibration values. Three arguments		
SatCalibrationV(Vmin Vaan	of word type are required: <i>Xmin</i> , the minimum joystick		
Vman)	calibration value, <i>Xcen</i> , the joystick center calibration		
Δπαλ)	value, and <i>Xmax</i> the maximum calibration value. When		
	manually calibrating the joystick, please pay attention to the		

	input order: please input the minimum value in <i>Xmin</i> and		
	the maximum value in <i>Xmax</i> . <i>Xmin, Xcen</i> and <i>Xmax</i>		
	ranging from 0 to 65535.		
	Set the y-axis joystick calibration values. Three arguments		
	of word type are required: <i>Ymin</i> , the minimum joystick		
	calibration value. <i>Ycen</i> the joystick center calibration		
SetCalibrationV(<i>Ymin Ycen</i>	value and <i>Ymar</i> the maximum calibration value. When		
Vmar)	manually calibrating the joystick please pay attention to the		
I mux)	input order: please input the minimum value in Vmin and		
	the maximum value in Vman Vmin Vacu and Vman		
	the maximum value in <i>Imax</i> . <i>Imin</i> , <i>Icen</i> and <i>Imax</i>		
	ranging from 0 to 65535.		
	Set the z-axis joystick calibration values. Three arguments		
	of word type are required: Zmin , the minimum joystick		
	calibration value, Zcen , the joystick center calibration		
SetCalibrationZ(Zmin, Zcen,	value, and <i>Zmax</i> the maximum calibration value. When		
Zmax)	manually calibrating the joystick, please pay attention to th		
	input order: please input the minimum value in Zmin and		
	the maximum value in Zmax. Zmin, Zcen and Zmax		
	ranging from 0 to 65535.		
	Activate the joystick calibration mode. After executing this		
	function, the joystick enters the calibration mode. At this		
	moment, please push the stick all the way up and move the		
	stick against the edges for 2 turns to get the maximum and		
	minimum displacements of the x- and y-axes. Next, twist		
	the z-axis to the left limit and the right limit, respectively		
StartCalibration()	and stay at the limit for 2 seconds for the joystick to record		
	the maximum and minimum z-axis displacement. Finally,		
	move the stick back to the center and let it stand for 3		
	seconds for the joystick to record the center of the x-, y- and		
	z-axes. Finally, push the button on the top of the stick to exit		
	the calibration mode.		
Commands for reading the joyst	ick coordinates		
	Read the current x and y coordinates of the joystick in the		
	rectangular coordinate system. X is the x coordinate, and Y		
$\operatorname{Get}\mathbf{X}\mathbf{Y}(X, Y)$	the v coordinate. Both have a default range from -127~127.		
	SetXYRes () changes the x and y ranges.		
	Read the current z coordinate of the joystick. \mathbf{Z} is the z		
GetZ(Z)	coordinate, and has a default range from -127 ~ 127.		

	SetKnobRes() changes the z range.			
	Read the integer polar coordinate of the joystick. The return			
	value <i>Radius</i> is the radial coordinate, and <i>Angle</i> the angular			
	coordinate with the positive x-axis being zero degrees.			
GetPolarBinaryRadian(<i>Radius</i> ,	Angle increases along the counterclockwise direction. By			
Angle)	default, <i>Radius</i> ranges from 0 to 127 and SetRadiusRes ()			
	changes its range. By default, <i>Angle</i> ranges from 0 to 127			
	and SetRadianRes () changes its range			
	Read the floating-point polar coordinate of the joystick. The			
C-4D-LD(D	first return value, <i>Radius</i> , is the floating-point radial			
GetPolarKadian(<i>Kadius</i> ,	coordinate, ranging from 0 to 1. The second return value			
Raaian)	<i>Radian</i> is the floating-point angular coordinate in unit of			
	radian, ranging from 0~6.37.			
	Read the 4-way joystick position. The return value <i>Dir</i> is a			
	directional value, ranging from 0~4. They represent:			
	$0 \rightarrow$ The joystick is in the center			
Dir = Get4WayStatus()	1 \rightarrow The joystick is on the right.			
	2 \rightarrow The joystick is at the bottom			
	$3 \rightarrow$ The joystick is on the left.			
	4 \rightarrow The joystick is on the top.			
	Read the 8-way joystick position. The return value <i>Dir</i> is a			
	directional value, ranging from 0~8. They represent:			
	$0 \rightarrow$ The joystick is in the center			
	1 \rightarrow The joystick is on the right.			
	2 \rightarrow The joystick is on the bottom right.			
Dir = Get8WayStatus()	3 \rightarrow The joystick is at the bottom.			
	4 \rightarrow The joystick is on the bottom left.			
	5 \rightarrow The joystick is on the left.			
	6 \rightarrow The joystick is on the top left.			
	7 \rightarrow The joystick is on the top.			
	8 \rightarrow The joystick is on the top right.			
Commands for reading joystick	range settings			
	Read the origin dead zones of the x-axis and the y-axis. The			
CatStick Dec dZame(DZa, DZa)	return values, DZx and DZy , are integers, ranging from 0 to			
GetSuckDeauZone(DZx, DZy)	10 in unit of %. The system treats as the origin those return			
	rectangular coordinates that fall within these ranges.			
CatKnahDaadZana(DZ-)	Read the origin dead zone of the z-axis. The return value,			
GermonDeauZone(DZZ)	DZz, is one integer, ranging from 0 to 10 in unit of %. The			

	system treats as the origin those return z coordinates that			
	fall within this range.			
	Read the origin dead zone of the radial axis. The return			
	value, <i>DZr</i> , is one integer, ranging from 0 to 10 in unit of			
GetRadiusDeadZone(DZr)	%. The system treats as the origin those return radial polar			
	coordinates that fall within this range.			
	Read the saturation limits of the x- and y- axes. The return			
	values, <i>SATx</i> and <i>SATy</i> , are integers, ranging from 60 to			
	100 in unit of %. The system returns the maximum or			
GetXYSaturation(SATx, SATy)	minimum values when the current rectangular coordinates,			
	positive or negative, exceed these limits. The system only			
	calculates division values with the full scale being the			
	saturation limit.			
	Read the saturation limit of the z-axis. The return value,			
	<i>SATz</i> , is one integer, ranging from 60 to 100 in unit of %.			
	The system returns the maximum or minimum values when			
GetKnobSaturation(SATz)	the current z coordinate, positive or negative, exceeds this			
	limit. The system only calculates division values with the			
	full scale being the saturation limit.			
	Read the saturation limit of the radial axis. The return value,			
	<i>SATr</i> , is one integer, ranging from 60 to 100 in unit of %.			
	The system returns the maximum or minimum values when			
GetRadiusSaturation(SAIr)	the current radial polar coordinate, positive or negative,			
	exceeds this limit. The system only calculates division			
	values with the full scale being the saturation limit.			
	Read the resolution settings of the x- and y-axes. The return			
	values, RESx and RESy , are integers, ranging from 0 to			
	128. Within the distinguishable ranges, the system quantizes			
C-4VVD(DEC., DEC.)	the rectangular coordinates by the scale of the resolution			
GetA Y Res(RESX, RESy)	settings. Because the index scale includes zero, a resolution			
	of 128 means the positive scale is divided into 128 divisions			
	from 0 to 127. Likewise, the negative scale is also divided			
	into 128 divisions, from 0 to -127.			
	Read the resolution setting of the z-axis. The return value,			
	<i>RESz</i> is one integer, ranging from 0 to 128. Within the			
GetKnobRes(RESz)	distinguishable ranges, the system quantizes the z			
	coordinate by the scale of the resolution setting. Because the			
	index scale includes zero, a resolution of 128 means the			

	positive scale is divided into 128 divisions from 0 to 127.		
	Likewise, the negatives scale is also divided into 128		
	divisions, from 0 to -127.		
	Read the resolution setting of the radial axis. The return		
	value, <i>RESr</i> is one integer, ranging from 0 to 128. Within		
	the distinguishable ranges, the system quantizes the radial		
GetRadiusRes(RESr)	polar coordinate by the scale of the resolution setting.		
	Because the index scale includes zero, a resolution of 128		
	means the positive scale is divided into 128 divisions from 0		
	to 127.		
	Read the resolution setting of the polar angle. The return		
	value, RESa is one integer, ranging from 0 to 360. The		
	system quantizes the polar angle by the scale of the		
GetRadianRes(RESa)	resolution setting. Because the index scale includes zero, a		
	resolution of 360 means the scale is divided into 360		
	divisions from 0 to 359.		
Commands for customizing joys	tick ranges		
RestoreSettings()	Restore all joystick settings to the default values.		
<u>0</u>	Set the origin dead zones of the x-axis and the y-axis. The		
	two arguments the x-axis and the y-axis settings.		
	respectively. Both DZx and DZy range from 0 to 10 in unit		
SetStickDeadZone(DZx, DZy)	of %. The system treats as the origin those rectangular		
	coordinates of joystick positions that fall within these		
	ranges		
	Set the origin dead zone of the z-axis. The argument DZ_{z} is		
	one integer ranging from 0 to 10 in unit of % The system		
SetKnobDeadZone(DZz)	treats as the origin those z coordinates of joystick positions		
	that fall within this range		
	Set the origin dead zone of the radial axis in the polar		
	coordinate system. The argument DZr is one integer.		
SetRadiusDeadZone(DZr)	ranging from 0 to 10 in unit of %. The system treats as the		
	origin those radial coordinates of joystick positions that fall		
	within this range		
	Set the saturation limits of the x_{-} and y_{-} axes in the		
	rectangular coordinate systems. The arguments SAT and		
SetXVSaturation(SATr SATu)	SAT _v are integers, ranging from 60 to 100 in unit of $\%$		
Seas i Saturation (SALA, SALY)	After executing the function the system returns the		
	maximum or minimum values when the current rectangular		
	maximum or minimum values when the current rectaligutar		

	coordinates, positive or negative, exceed these limits. The		
	system only calculates division values with the full scale		
	being the saturation limit.		
	Set the saturation limit of the z-axis. The argument, $SATz$, is		
	one integer, ranging from 60 to 100 in unit of %. After		
	executing the function, the system returns the maximum or		
SetKnobSaturation(SATz)	minimum values when the current z coordinate, positive or		
	negative, exceeds this limit. The system only calculates		
	division values with the full scale being the saturation limit.		
	Set the saturation limit of the radial axis in the polar		
	coordinate system The argument SATr is one integer		
	ranging from 60 to 100 in unit of % After executing the		
Sot D adius Saturation $(SATr)$	function the system returns the maximum or minimum		
SetKaulusSaturation(SATT)	rule to it. The system returns the maximum of minimum		
	values when the current radial polar coordinate, positive or		
	negative, exceeds this limit. The system only calculates		
	division values with the full scale being the saturation limit.		
	Set the resolutions of the x- and y-axes. The arguments,		
	<i>RESx</i> and <i>RESy</i> , are integers, ranging from 0 to 128.		
	Within the distinguishable ranges, the system quantizes the		
	rectangular coordinates by the scale of the resolution		
SotVVDos(PESr PESy)	settings. Because the index scale includes zero, a resolution		
SetA I Res(RESA, RESy)	of 128 means the positive scale is divided into 128 divisions		
	from 0 to 127. Likewise, the negative scale is also divided		
	into 128 divisions, from 0 to -127. Please not that, although		
	the resolution can be 0 or 1, the return x and y coordinates		
	are both zero with 0 or 1 resolution settings.		
	Set the resolution setting of the z-axis. The argument, <i>RESz</i>		
	is one integer, ranging from 0 to 128. Within the		
	distinguishable ranges, the system quantizes the z		
	coordinate by the scale of the resolution setting. Because the		
	index scale includes zero, a resolution of 128 means the		
SetKnobRes(RESz)	positive scale is divided into 128 divisions from 0 to 127.		
	Likewise, the negative scale is also divided into 128		
	divisions from 0 to -127		
	Please not that although the resolution can be 0 or 1 the		
	return z coordinate is zero with 0 or 1 resolution settings		
	Set the resolution setting of the radial axis. The argument		
SetRadiusRes(RESr)	DEST is one integer renging from 0 to 129 Within the		
	NEST , is one integer, ranging from 0 to 128. within the		

distinguishable ranges, the system quantizes the z coordinate by the scale of the resolution setting. Because the index scale includes zero, a resolution of 128 means the scale is divided into 128 divisions from 0 to 127. Please not that, although the resolution can be 0 or 1, the return radial coordinate is zero with 0 or 1 resolution settings.SetRadianRes(RESa)Set the resolution setting of the polar angle. The argument, RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is pressed. 1: The button is not pressed. 2: The button is disabled: 0: The button is disabled: 0: The button is not pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is not pressed. 2: The button is not pressed. 3: The button is not pressed. 4: The button is not pressed. 4: The button is not pressed. 4: The button is				
coordinate by the scale of the resolution setting. Because the index scale includes zero, a resolution of 128 means the scale is divided into 128 divisions from 0 to 127. Please not that, although the resolution can be 0 or 1, the return radial coordinate is zero with 0 or 1 resolution settings.SetRadianRes(RESa)Set the resolution setting of the polar angle. The argument, RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition of the joystick button. The argument, Reg, is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the ipoystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the ipoystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released. <th></th> <th>distinguishable ranges, the system quantizes the z</th>		distinguishable ranges, the system quantizes the z		
SetRadianRes(RESa)index scale includes zero, a resolution of 128 means the scale is divided into 128 divisions from 0 to 127. Please not that, although the resolution can be 0 or 1, the return radial coordinate is zero with 0 or 1 resolution settings.Set the resolution setting of the polar angle. The argument, RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is pressed. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition or the joystick button. The argument, Rep, is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button i		coordinate by the scale of the resolution setting. Because the		
scale is divided into 128 divisions from 0 to 127. Please not that, although the resolution can be 0 or 1, the return radial coordinate is zero with 0 or 1 resolution settings.SetRadianRes(RESa)Set the resolution setting of the polar angle. The argument, RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution setting.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is		index scale includes zero, a resolution of 128 means the		
Iterationthat, although the resolution can be 0 or 1, the return radial coordinate is zero with 0 or 1 resolution settings.Set the resolution setting of the polar angle. The argument, <i>RESa</i> , is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution as be 0 or 1, the return polar angle is zero with 0 or 1 resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution as be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in <i>Sta</i> . The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(<i>Rep</i>)Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. <th></th> <th>scale is divided into 128 divisions from 0 to 127. Please not</th>		scale is divided into 128 divisions from 0 to 127. Please not		
coordinate is zero with 0 or 1 resolution settings.Set the resolution setting of the polar angle. The argument, <i>RESa</i> , is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in <i>Sta</i> . The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is not pressed. 1: The button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(<i>Rep</i>)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Nolly one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Men the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Nolly one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Nen the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the aut		that, although the resolution can be 0 or 1, the return radial		
SetRadianRes(RESa)Set the resolution setting of the polar angle. The argument, RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joysti		coordinate is zero with 0 or 1 resolution settings.		
SetRadianRes(RESa)RESa, is one integer, ranging from 0 to 360. The system quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is pressed. 1: The button is pressed. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the		Set the resolution setting of the polar angle. The argument,		
SetRadianRes(RESa)quantizes the polar angle by the scale of the resolution setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		RESa , is one integer, ranging from 0 to 360. The system		
SetRadianRes(RESa)setting. Because the index scale includes zero, a resolution of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is not pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		quantizes the polar angle by the scale of the resolution		
SetKadiankes(KE.Sa)of 360 means the scale is divided into 360 divisions from 0 to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is pressed. 1: The button is not pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		setting. Because the index scale includes zero, a resolution		
to 359. Please not that, although the resolution can be 0 or 1, the return polar angle is zero with 0 or 1 resolution settings.Commands for button operationsRead the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed.	SetRadianRes(<i>RESa</i>)	of 360 means the scale is divided into 360 divisions from (
I, the return polar angle is zero with 0 or 1 resolution settings. Commands for button operations Read the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed. ButtonRepeatFunc(Rep) Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held.		to 359. Please not that, although the resolution can be 0 or		
settings. Commands for button operations Read the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. Sta = GetButtonStatus() 0: The button is pressed. Sta = GetButtonStatus() 0: The button is pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 0: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed. Enable and disables the automatic keystroke repetition of the joystick button. The argument, Rep, is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held.		1, the return polar angle is zero with 0 or 1 resolution		
Commands for button operations Read the button status and saves it in <i>Sta</i> . The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed. ButtonRepeatFunc(Rep) Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. ButtonRepeatFunc(Rep)		settings.		
Read the button status and saves it in Sta. The return values are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat 	Commands for button operation	S		
Sta = GetButtonStatus()are described below: When the keystroke repetition is disabled: 0: The button is pressed. 1: The button is not pressed. 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		Read the button status and saves it in <i>Sta</i> . The return values		
Sta = GetButtonStatus()When the keystroke repetition is disabled: 0: The button is not pressed. 1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		are described below:		
Sta = GetButtonStatus()0: The button is pressed. 1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed.		When the keystroke repetition is disabled:		
Sta = GetButtonStatus()1: The button is not pressed. When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		0: The button is pressed.		
Sta = GetButtonStatus()When the keystroke repetition is enabled: 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.ButtonRepeatFunc(Rep)Enable and disables the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		1: The button is not pressed.		
Sta = GetButtonStatus() 0: No new button event is detected. 1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed. Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		When the keystroke repetition is enabled:		
1: The button is just pressed down or the button is pressed and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonRepeatFunc(<i>Rep</i>)ButtonRepeatFunc(<i>Rep</i>)pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.	<i>Sta</i> = GetButtonStatus()	0: No new button event is detected.		
and held for a time equal to or longer than repeat time. The value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.Enable and disables the automatic keystroke repetition of the joystick button. The argument, Rep, is one integer with input values of 0 and 1 only.0: Disable the automatic keystroke repetition. Only one ButtonRepeatFunc(Rep)Button RepeatFunc(Rep)(pressed) when the joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		1: The button is just pressed down or the button is pressed		
value will be set as 1 repeatedly after an interval of repeat rate. This value remains 1 until GetButtonStatus is executed.Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only.0: Disable the automatic keystroke repetition. Only one ButtonRepeatFunc(<i>Rep</i>)ButtonRepeatFunc(<i>Rep</i>)(pressed) when the joystick button is pressed. The return joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		and held for a time equal to or longer than repeat time. The		
rate. This value remains 1 until GetButtonStatus is executed.Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		value will be set as 1 repeatedly after an interval of repeat		
executed.Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only.0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		rate. This value remains 1 until GetButtonStatus is		
Enable and disables the automatic keystroke repetition of the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only. 0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		executed.		
the joystick button. The argument, <i>Rep</i> , is one integer with input values of 0 and 1 only.0: Disable the automatic keystroke repetition. Only one ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		Enable and disables the automatic keystroke repetition of		
input values of 0 and 1 only.0: Disable the automatic keystroke repetition. Only oneButtonEvent is activated when the joystick button ispressed. The return joystick button status remains 1(pressed) when the joystick button is pressed and held.1: Enable the automatic keystroke repetition. When thejoystick button is pushed down, ButtonEvents are activatedrepeatedly according to repeat time and repeat rate until thebutton is released.		the joystick button. The argument, <i>Rep</i> , is one integer with		
0: Disable the automatic keystroke repetition. Only oneButtonEvent is activated when the joystick button ispressed. The return joystick button status remains 1(pressed) when the joystick button is pressed and held.1: Enable the automatic keystroke repetition. When thejoystick button is pushed down, ButtonEvents are activatedrepeatedly according to repeat time and repeat rate until thebutton is released.		input values of 0 and 1 only.		
ButtonRepeatFunc(Rep)ButtonEvent is activated when the joystick button is pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		0: Disable the automatic keystroke repetition. Only one		
ButtonRepeatFunc(Rep)pressed. The return joystick button status remains 1 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		ButtonEvent is activated when the joystick button is		
 (pressed) when the joystick button is pressed and held. 1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released. 	ButtonRepeatFunc(<i>Rep</i>)	pressed. The return joystick button status remains 1		
1: Enable the automatic keystroke repetition. When the joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		(pressed) when the joystick button is pressed and held.		
joystick button is pushed down, ButtonEvents are activated repeatedly according to repeat time and repeat rate until the button is released.		1: Enable the automatic keystroke repetition. When the		
repeatedly according to repeat time and repeat rate until the button is released.		joystick button is pushed down, ButtonEvents are activated		
button is released.		repeatedly according to repeat time and repeat rate until the		
		button is released.		

	Set the activation interval of automatic keystroke repetition.		
	This value ranges from 0 to 255, in unit of 10ms. After		
	setting, the system returns a ButtonEvent when the button		
	is pushed and held longer than this setting and starts		
SotPopostTime(Time)	repeatedly returning ButtonEvents according to repeat rate.		
SetKepeat Time(<i>Time</i>)	Please note that ButtonRepeatFunc () has to be executed		
	first to enable the automatic keystroke repetition. Besides,		
	zero reat time does not disable the automatic keystroke		
	repetition. Instead, the system immediately starts return		
	ButtonEvents repeatedly according to repeat rate.		
Cat Danast Time(Time)	Read the activation interval of automatic keystroke		
GetRepeat Time(Time)	repetition. This return value ranges from 0 to 255.		
	Set the repetition interval of automatic keystroke repetition.		
	This value <i>Rate</i> ranges from 0 to 255, in unit of 10ms. After		
	setting, the system returns ButtonEvents repeatedly every		
	interval of repeat rate when the button is pushed and held		
SetRepeatRate(Rate)	longer than repeat time. Please note that		
	ButtonRepeatFunc() has to be executed first to enable the		
	automatic keystroke repetition. With zero repeat rate, the		
	system does not repeatedly return ButtonEvents after		
	repeat time.		
CotPopostPato(Pata)	Read the repetition interval of automatic keystroke		
OctReptatRate(Rule)	repetition. This return value <i>Rate</i> ranges from 0 to 255.		
Commands for enabling and dis	abling events		
EnableStickEvent()	Enable StickEvent		
DisableStickEvent()	Disable StickEvent		
EnableKnobEvent()	Enable KnobEvent		
DisableKnobEvent()	Disable KnobEvent		
Enable4WayEvent()	Enable Chang4WayEvent		
Disable4WayEvent()	Disable Chang4WayEvent		
Enable8WayEvent()	Enable Chang8WayEvent		
Disable8WayEvent()	Disable Chang8WayEvent		
EnableButtonEvent()	Enable ButtonEvent		
DisableButtonEvent()	Disable ButtonEvent		
EnableCalEndEvent()	Enable CalEndEvent		
DisableCalEndEvent()	Disable CalEndEvent		
SetEventRefreshRate(<i>Rate</i>)	Set the event refresh rate. This value <i>Rate</i> ranges from 0 to		

	255 in unit of 10ms. After setting, two successive events	
	will be activated in the intervals set by this function. Please	
	note that zero refresh rates are equivalent to the refresh rate	
	of 1.	
CatEventDefreebDate(Date)	Read the event refresh rate. This return value <i>Rate</i> ranges	
GetEventKelfesnKate(Kate)	from 0 to 255.	

Event	Activating Conditions		
StickEvent	This event is activated when joystick movement is detected.		
KnobEvent	This event is activated when joystick twist is detected.		
Change/WayEyent	This event is activated when the 4-way value of the joystick		
Change4 way Event	changes.		
Change Way Event	This event is activated when the 8-way value of the joystick		
ChangeowayEvent	changes.		
	When the automatic keystroke repetition is disabled:		
	This event is activated when the joystick button is pushed.		
ButtonEvent	When the automatic keystroke repetition is enabled:		
	This event is activated when the joystick button is pressed after		
	an internal of Repeat Time and, every interval of Repeat Rate.		
CalEndEvent	This event is activated when the joystick calibration completes.		

Exaple Program:

(Detects joystick movements and twists, and returns its x, y and z coordinates when movements or twists are detected.)

```
Peripheral myJoy As Joystick3A @ 0
                                              Set the module to be operated as 0.
Sub Main()
     myJoy.SetStickDeadZone(2, 2)
                                        "
                                              Set the x- and y-axis origin dead zones.
                                        "
     myJoy.SetKnobDeadZone(2)
                                              Set the z-axis origin dead zone.
                                        "
     myJoy.SetXYSaturation(80, 80)
                                              Set the x- and y-axis saturation limits.
                                        "
     myJoy.SetKnobSaturation(80)
                                              Set the z-axis saturation limit.
                                        "
     myJoy.SetXYRes(128, 128)
                                              Set the x- and y-axis resolutions.
     myJoy.SetKnobRes(128)
                                        "
                                              Set the z-axis resolution.
     myJoy.SetEventRefreshRate(1)
                                        "
                                              Set the event refresh rate.
                                        "
     myJoy.EnableStickEvent()
                                              Enable StickEvent.
                                        "
     myJoy.EnableKnobEvent()
                                              Enable KnobEvent.
     Do
```

Loop

End Sub

```
Event myJoy.StickEvent()
```

Dim sX, sY As Short

```
myJoy.GetXY(sX, sY)
Debug CSRXY(1, 1), "X: ", %DEC4R sX, ", Y: ", %DEC4R sY, CR
```

End Event

```
Event myJoy.KnobEvent()
```

```
Dim sZ As Short
```

```
myJoy.GetZ(sZ)
Debug CSRXY(17, 1), ", Z:", %DEC4R sZ, CR
```

```
End Event
```

Appendix

- 1. Known problem:
 - At version 1.0, command "RestoreSettings" will only change the values in RAM. After turn off the power and turn on again, the settings will become the last set values.
- 2. Table for the module numbers and the switches:

0	8	16	24
1	9	17	25
2	10	18	26
3	11	19	27
4	12	20	28
5	13	21	29
6	14	22	30
7	15	23	31