

# Universal Serial Bus (USB)

## Device Class Definition for Physical Interface Devices (PID)

9/8/99

Version 1.0

Please send comments via electronic mail to:  
*hidcomments@usb.org*

## ***Intellectual Property Disclaimer***

THIS SPECIFICATION IS PROVIDED “AS IS” WITH NO WARRANTIES WHATSOEVER INCLUDING ANY WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION, OR SAMPLE.

TO THE MAXIMUM EXTENT OF USB IMPLEMENTERS FORUM’S RIGHTS, USB IMPLEMENTERS FORUM HEREBY GRANTS A LICENSE UNDER COPYRIGHT TO REPRODUCE THIS SPECIFICATION FOR INTERNAL USE ONLY (E.G., ONLY WITHIN THE COMPANY OR ORGANIZATION THAT PROPERLY DOWNLOADED OR OTHERWISE OBTAINED THE SPECIFICATION FROM USB IMPLEMENTERS FORUM, OR FOR AN INDIVIDUAL, ONLY FOR USE BY THAT INDIVIDUAL). THIS SPECIFICATION MAY NOT BE REPUBLISHED EXTERNALLY OR OTHERWISE TO THE PUBLIC.

IT IS CONTEMPLATED THAT MANY IMPLEMENTATIONS OF THIS SPECIFICATION (E.G., IN A PRODUCT) DO NOT REQUIRE A LICENSE TO USE THIS SPECIFICATION UNDER COPYRIGHT. FOR CLARITY, HOWEVER, TO THE MAXIMUM EXTENT OF USB IMPLEMENTERS FORUM’S RIGHTS, USB IMPLEMENTERS FORUM HEREBY GRANTS A LICENSE UNDER COPYRIGHT TO USE THIS SPECIFICATION AS REASONABLY NECESSARY TO IMPLEMENT THIS SPECIFICATION (E.G., IN A PRODUCT).

NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS IS GRANTED OR INTENDED HEREBY.

USB IMPLEMENTERS FORUM AND THE AUTHORS OF THIS SPECIFICATION DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF PROPRIETARY RIGHTS, RELATING TO IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. AUTHORS OF THIS SPECIFICATION ALSO DO NOT WARRANT OR REPRESENT THAT SUCH IMPLEMENTATION(S) WILL NOT INFRINGE SUCH RIGHTS.

All product names are trademarks, registered trademarks, or service marks of their respective owners.

## ***Contributors***

<b>Company</b>	<b>Contact</b>
Microsoft Corporation	Manolito Adon
Immersion Corporation	Adam Braun
Microsoft Corporation	Trudy Culbreth
Microsoft Corporation	Gary Gordon
Intel Corporation	Steve McGowan - editor
Microsoft Corporation	Ervin Peretz
Microsoft Corporation	Om Sharma

***Revision History***

<b>Revision</b>	<b>Issue Date</b>	<b>Comments</b>
1.0	September 8, 1999	1.0 Release

## Table of Contents

<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1 SCOPE .....	1
1.2 PURPOSE .....	1
1.3 RELATED DOCUMENTS .....	1
1.4 DEFINITIONS .....	1
<b>2. FUNCTIONAL OVERVIEW .....</b>	<b>2</b>
<b>3. CONFIGURATIONS .....</b>	<b>3</b>
3.1 CONFIGURATION AND INTERFACE STRUCTURE .....	3
3.1.1 Configuration .....	3
3.1.2 Interface Definition .....	3
<b>4. CONVENTIONS .....</b>	<b>4</b>
4.1 ACTUATORS .....	4
4.2 UNITS .....	4
<b>5. PHYSICAL INPUT DEVICE PAGE (0X0F) .....</b>	<b>6</b>
5.1 EFFECT BLOCK PARAMETERS .....	8
5.1.1 Effect Types .....	8
5.2 ENVELOPE BLOCK PARAMETERS .....	11
5.3 CONDITION BLOCK PARAMETERS .....	12
5.4 CUSTOM FORCES .....	13
5.4.1 Sample Definition .....	13
5.4.2 Data Downloading .....	14
5.4.3 Custom Force Block Parameters .....	14
5.5 PERIODIC BLOCK PARAMETERS .....	15
5.6 CONSTANT FORCE BLOCK PARAMETERS .....	15
5.7 RAMP FORCE BLOCK PARAMETERS .....	15
5.8 VENDOR DEFINED PARAMETER BLOCKS .....	16
5.9 EFFECT OPERATIONS .....	16
5.10 DEVICE GAIN .....	16
5.11 PARAMETER BLOCK MANAGEMENT .....	17
5.11.1 Driver Managed .....	17
5.11.2 Device Managed .....	21
5.12 PID STATE .....	22
5.13 PID DEVICE CONTROL .....	23
<b>6. PID EXAMPLE .....</b>	<b>24</b>
<b>7. EXAMPLE REPORT SUMMARY .....</b>	<b>34</b>
7.1 INPUT REPORTS .....	34
7.2 OUTPUT REPORTS .....	34
7.3 FEATURE REPORTS .....	34
7.4 OTHER REPORTS .....	34
<b>8. EXAMPLE REPORTS .....</b>	<b>35</b>
<b>9. EXAMPLE OF CREATING A DEVICE MANAGED EFFECT .....</b>	<b>41</b>
<b>10. PARAMETER BLOCKS .....</b>	<b>44</b>

11. INDEX .....45

## **Table of Tables**

Table 1: Actuator Usages .....	4
Table 2: Physical Input Device Page.....	6
Table 3: Effect Types .....	11

## **Table of Figures**

Figure 1: Report Allocation in the Pool .....	18
Figure 2: Custom Effect Allocation in the Pool .....	19
Figure 3: Example PID Data Input Report.....	35
Figure 4: Example PID Set Effect Output Report .....	35
Figure 5: Example PID Set Envelope Output Report.....	36
Figure 6: Example PID Set Condition Output Report .....	36
Figure 7: Example PID Set Periodic Output Report.....	37
Figure 8: Example PID Set Constant Force Output Report.....	37
Figure 9: Example PID Set Ramp Force Output Report .....	37
Figure 10: Example PID Set Custom Force Data Output Report.....	38
Figure 11: Example PID Download Force Sample Output Report .....	38
Figure 12: Example PID Set Custom Force Output Report .....	38
Figure 13: Example PID Effect Operation Output Report .....	39
Figure 14: Example PID Pool Feature Report.....	39
Figure 15: Example PID State Input Report .....	39
Figure 16: Example PID Device Control Output Report.....	40
Figure 17: Example PID Pool Move Output Report .....	40
Figure 18: Example PID Device Gain Feature Report .....	40
Figure 19: Parameter Blocks .....	44

# 1. Introduction

## 1.1 Scope

This Document describes the USB Physical Interface Device or PID Class. The PID Class is intended as an extension of the Human Interface Device (HID) Class for devices that require “real-time” physical feedback. The main focus for the definition is for the use of haptic devices and the implementation of force feedback systems. However there is no requirement that members of this class generate this type of effect. Examples of devices that are member of this class are force feedback joysticks, motion platforms, and force feedback exoskeletons.

## 1.2 Purpose

This document is intended to supplement to the USB specification and provide manufacturers of PID class devices with the information necessary to build USB compatible devices. Further, this specification specifies how the PID class driver should extract data from USB data streams. Since this device class is an extension of the HID class, the HID class specifications will be required to implement the device functionality.

## 1.3 Related Documents

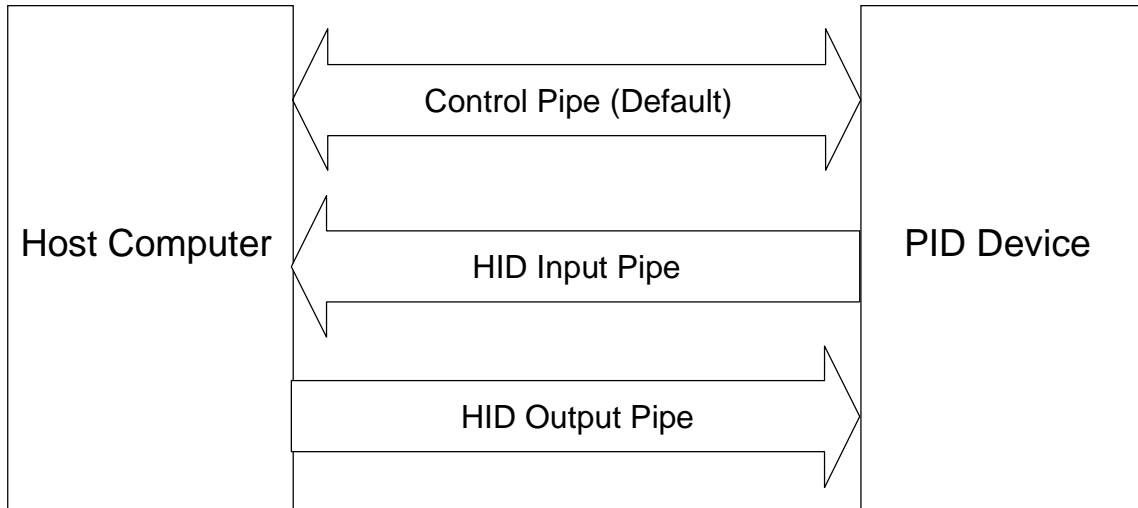
Universal Serial Bus Specification  
Device Class Definition for Human Interface Devices (HID)  
Universal Serial Bus HID Usage Tables

## 1.4 Definitions

Set <i>PB</i> Report	This term is used to refer to any of the Set-Report commands. Where Set-Reports represents one of the following report types: Set Effect Report, Set Custom Force Report, Set Periodic Report, Set Constant Force Report, Set Ramp Force Report, Set Vendor Report, Set Envelope Report, and Set Condition Report.
INFINITE	Referrers to the maximum value of a range. i.e. if in an 8 bit unsigned field the value of 255 would indicate INFINITE.
Normalized	When a usage is declared as a <i>Normalized</i> value, then the device assumes that a range of values between %0 and %100 can be applied by the device where the actual applied value is a percentage $p$ , where $p = (\text{Usage Value} / (\text{Logical Maximum} - \text{Logical Minimum})) * 100$ .

## 2. Functional Overview

The Functional connection model for the PID class is below.



The data transfer mechanism for the PID class is based on the HID class Report Descriptors. Communication with a PID device is identical to that of a HID device.

Report Type	Transfer for a GET_REPORT	Transfer for a SET_REPORT
Input	HID Interrupt In pipe	Control pipe
Output	Control pipe	HID Interrupt Out pipe
Feature	Control pipe	Control pipe



### 3. Configurations

#### 3.1 Configuration and Interface Structure

The PID class has a standard configuration and interface structure defined. The interface structure is designed to be an extension of the HID interface that will allow for the needed low latency output requirement. The first pipe is the default control pipe required for all devices. The second pipe in the interface is an interrupt IN pipe that will process reports as described by the HID specification. The third pipe is an interrupt OUT pipe. The intent of this pipe is to handle all output reports that are sent to the device. This will allow commands to be sent to the device with minimum latency.

##### 3.1.1 Configuration

##### 3.1.2 Interface Definition

Pipe	Type	Description	Required?
0	ctrl	USB control & class SET/GET commands	Y
a	interrupt in	HID data in from device	Y
b	interrupt out	PID command out to the device	N

## 4. Conventions

### 4.1 Actuators

An *Actuator* is a mechanism that can exert a force against an object. Typically the force applied by the actuator can be defined in some combination of 6 dimensions; X, Y, Z, Rx, Ry and Rz. By definition in the HID specification these dimensions are defined in a coordinate system that is referenced to the direction that the user is facing. For simplicity the usages X, Y, Z, Rx, Ry and Rz from the Generic Desktop page are used by PID class devices to define a NORMALIZED force values. i.e. if the Logical Minimum and Logical Maximum values for an axis are defined to be -100 and +100 respectively, then -100 represents maximum force in the negative direction, 0 no force, and +100 maximum force in the positive direction. If a device requires the assertion of precise absolute force then the Vectors Vx, Vy, and Vz can be used from the Generic Desktop usage table with Units declaring the actual range of applied force. Linear absolute forces result from setting the Units associated with the vector usages to SI Linear or English Linear. Absolute torque forces result from setting the Units associated with the vector usages to SI Rotation or English Rotation.

In some cases the application of force does not have an easily identifiable set of dimensions that can be applied. An example is the force applied by a pressure suit in a flight simulator. In this document this type of “dimensionless” force will be referred to as a *Normal* force. For simplicity the PID:Normal usage is used to define a Normalized force that can vary between 0 and the maximum force that a device can assert. If the assertion of precise absolute force is required by a device then the vector Vno can be used from the Generic Desktop usage table with Units declaring the actual range of applied force.

**Table 1: Actuator Usages**

Actuator Type	Normalized Force Usages	Absolute Force Usages	Orientation
Dimensionless	PID: Normal	Vno	Perpendicular to the surface of an object.
Translation	Generic Desktop: X	Vx, Uint = linear	Force in linear X direction
Translation	Generic Desktop: Y	Vy, Uint = linear	Force in linear Y direction
Translation	Generic Desktop: Z	Vz, Uint = linear	Force in linear Z direction
Rotation	Generic Desktop: Rx	Vx, Uint = rotation	Torque about X
Rotation	Generic Desktop: Ry	Vy, Uint = rotation	Torque about Y
Rotation	Generic Desktop: Rz	Vz, Uint = rotation	Torque about Z

### 4.2 Units

Usages state specifically whether *Units* are required to define a time base or other reference. A *Units* requirement implies the definition of Units, Physical Maximum, Physical Minimum and Unit Exponent items.

Remember that Units are **global** items that persist from main item to main item. *Units* can be disabled by setting Units to “None”, and Physical Maximum, Physical Minimum and Unit Exponent equal to 0. Applications will typically ignore *Units* when it encounters a Usage that they are not appropriate for, but don’t count on it. Always make sure that the Units are correctly declared for every Usage.

*Normalized* Usages use Logical Minimum and Logical Maximum to define their range of values. These values are automatically scaled by an application to fit the range of values required for the target operation and do not require *Units*.

To declare a vector ( $V_x$ ,  $V_y$  or  $V_z$ ) as a linear force, define the *System* field of the Unit item associated with the vector usage to be either *SI Linear* or *English Linear*. To declare vectors as absolute torques, define the *System* field of the Unit item to be either *SI Rotation* or *English Rotation*.

## 5. Physical Input Device Page (0x0F)

Force feedback devices use PID usages to describe their output and Usages from other Usage Pages to describe their input.

A Physical Input Device can use one of two methods to generate sensations to the user: Effects or Custom Effects. A device may support either or both methods. Effects generate force feedback based on the parameters stored in a variety of parameter blocks (structures) that are maintained in the device. The host loads the parameter blocks, then issues a Start Effect Command over an Interrupt Out pipe. Custom Effects are similar to playing an audio file, force information for each axis can be downloaded and “played” locally or sent to the device over an Isochronous Output pipe.

Effects are comprised of parameter blocks. There are 6 types of parameter blocks defined: Effect, Envelope, Condition, Periodic, Constant Force and Ramp Force. All Effects require an Effect Parameter Block. Depending on the effect one or more additional parameter blocks will be required. Parameter blocks are accessed by an handle. The size of a parameter block for a particular device is a function of the size of the associated report.

A default allocation of each type of parameter block is provided by the device, however the host driver can modify the resource allocation to best meet its needs.

**Table 2: Physical Input Device Page**

Usage ID	Usage Name	Usage ID	Usage Name
00	Undefined	72	Period
01	<b>Physical Interface Device</b>	73	<b>Set Constant Force Report</b>
02-1F	Reserved	74	<b>Set Ramp Force Report</b>
20	Normal	75	Ramp Start
21	<b>Set Effect Report</b>	76	Ramp End
22	Effect Block Index	77	<b>Effect Operation Report</b>
23	Parameter Block Offset	78	<b>Effect Operation</b>
24	ROM Flag	79	Op Effect Start
25	<b>Effect Type</b>	7A	Op Effect Start Solo
26	ET Constant Force	7B	Op Effect Stop
27	ET Ramp	7C	Loop Count
28	ET Custom Force Data	7D	<b>Device Gain Report</b>
29-2F	Reserved	7E	Device Gain
30	ET Square	7F	<b>PID Pool Report</b>
31	ET Sine	80	RAM Pool Size
32	ET Triangle	81	ROM Pool Size
33	ET Sawtooth Up	82	ROM Effect Block Count
34	ET Sawtooth Down	83	Simultaneous Effects Max
35-3F	Reserved	84	Pool Alignment
40	ET Spring	85	<b>PID Pool Move Report</b>
41	ET Damper	86	Move Source
42	ET Inertia	87	Move Destination

## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

Usage ID	Usage Name	Usage ID	Usage Name
43	ET Friction	88	Move Length
44-4F	Reserved	89	<b>PID Block Load Report</b>
50	Duration	8A	Reserved
51	Sample Period	8B	<b>Block Load Status</b>
52	Gain	8C	Block Load Success
53	Trigger Button	8D	Block Load Full
54	Trigger Repeat Interval	8E	Block Load Error
55	<b>Axes Enable</b>	8F	Block Handle
56	Direction Enable	90	<b>PID Block Free Report</b>
57	<b>Direction</b>	91	<b>Type Specific Block Handle</b>
58	<b>Type Specific Block Offset</b>	92	<b>PID State Report</b>
59	<b>Block Type</b>	93	Reserved
5A	<b>Set Envelope Report</b>	94	Effect Playing
5B	Attack Level	95	<b>PID Device Control Report</b>
5C	Attack Time	96	<b>PID Device Control</b>
5D	Fade Level	97	DC Enable Actuators
5E	Fade Time	98	DC Disable Actuators
5F	<b>Set Condition Report</b>	99	DC Stop All Effects
60	CP Offset	9A	DC Device Reset
61	Positive Coefficient	9B	DC Device Pause
62	Negative Coefficient	9C	DC Device Continue
63	Positive Saturation	9D	Reserved
64	Negative Saturation	9E	Reserved
65	Dead Band	9F	Device Paused
66	<b>Download Force Sample</b>	A0	Actuators Enabled
67	Isoch Custom Force Enable	A1-A3	Reserved
68	<b>Custom Force Data Report</b>	A4	Safety Switch
69	Custom Force Data	A5	Actuator Override Switch
6A	Custom Force Vendor Defined Data	A6	Actuator Power
6B	<b>Set Custom Force Report</b>	A7	Start Delay
6C	Custom Force Data Offset	A8	<b>Parameter Block Size</b>
6D	Sample Count	A9	Device Managed Pool
6E	<b>Set Periodic Report</b>	AA	Shared Parameter Blocks
6F	Offset	AB	<b>Create New Effect Report</b>
70	Magnitude	AC	RAM Pool Available
71	Phase	AB-FFFF	Reserved

The following controls support force feedback devices.

### Physical Interface Device

Normal

CA - A collection of PID usages.

DV – A force applied perpendicular to the surface of an object.

## 5.1 Effect Block Parameters

The minimal Effect parameter block must contain (Effect) Parameter Block Index, Effect Type, Duration, Sample Period, Gain, Trigger Button, Trigger Repeat Interval, Axis Direction, and Type Specific Block Handle values.

<b>Set Effect Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the effect parameters of a PID.
Effect Block Index	DV - Effect Block Index varies from 1 to the maximum number of effects stored in the device. See Parameter Block Management for more information.
Parameter Block Offset	DV – For a device that employs a Driver Managed memory model, this usage represents the byte offset into the Parameter block pool where the Parameter Block is stored.  For a device that employs a Device Managed memory model, this usage defines an index of the target axis. The order of axis declaration in the Axes Enable collection defines the index. In the example Report Descriptor in section 6, two axes are declared, X and Y. The X axis is declared first so an offset of 0 is applied when referencing the X axis, and an offset of 1 when referencing the Y axis. See section 9 for more information.
ROM Flag	DF – This flag used to distinguish between RAM and ROM based Effect Block Indices and Offsets.

### 5.1.1 Effect Types

Individual effects are identified by the contents of the Effect Type (ET) field. The declared ET usages enumerate the effects that a device supports. See Table 3 for required Type Specific Block Offsets.

<b>Effect Type</b>	NArY – This collection identifies the Type of Effect this command defines. See Table 3 for list of applicable Effect Type selectors. Vendor defined effect type selectors can also be included in this collection.
ET Constant Force	Sel – The current effect parameters define a Constant waveform.
ET Ramp	Sel - The current effect parameters define a Ramp waveform.
ET Custom Force Data	Sel - The current effect parameters define a downloaded Custom Force waveform.
ET Square	Sel - The current effect parameters define a Square waveform.
ET Sine	Sel - The current effect parameters define a Sine waveform.
ET Triangle	Sel - The current effect parameters define a Triangle waveform.
ET Sawtooth Up	Sel - The current effect parameters define a Sawtooth Up waveform.
ET Sawtooth Down	Sel - The current effect parameters define a Sawtooth Up waveform.
ET Spring	Sel - The current effect parameters creates a compliant restoring force that causes the device to return to a specified point in the device's range of motion.

ET Damper	Sel - The current effect parameters define a Damper effect.
ET Inertia	Sel - The current effect parameters define a Inertia effect
ET Friction	Sel - The current effect parameters define a Friction effect.
Duration	DV - The total duration of the effect. To sustain an effect until explicitly stopped with the <b>Stop</b> method, set <i>Duration</i> to INFINITE (Null). If an envelope has been applied to the effect, then the attack will be applied, followed by an infinite sustain. Units are used to define the time base.
Sample Period	<p>DV - The period at which the device should play back the effect. A value of zero indicates that the default playback sample rate should be used. Units are used to define the time base.</p> <p>If the device is not capable of playing back the effect at the specified rate, it will choose the supported rate that is closest to the requested value.</p> <p>Setting a custom Sample Period can be used for special effects. For example, playing a sine wave at an artificially large sample period results in a rougher texture.</p>
Gain	DV - The gain to be applied to the effect. The gain is a normalized scaling factor that is applied to all magnitudes and envelopes of an effect.
Trigger Button	DV - The identifier or offset of the button to be used to trigger playback of this effect. A Null Trigger Button value indicates that this Effect is not tied to a button.
Trigger Repeat Interval	DV - The auto-repeat interval, for playback of effects triggered by holding down the trigger button. The interval is the time between the end of the playing effect and start of the next effect. If this effect is a one-time effect (no auto-repeat is desired), this value should be set to INFINITE (Null). Units are used to define the time base.
<b>Axes Enable</b>	US – The <i>Axes Enable</i> collection contains joint collections. Each joint collection contains axes or vectors from the Generic Desktop page. This usage changes the type of these Generic Desktop usages to Dynamic Flags (DF) where each usage identifies whether the respective axis or vector is enabled for this effect.
Direction Enable	DF – If the <i>Direction Enable</i> flag is set then the Axes Enable flags are ignored, only one Condition Parameter Block is defined and the Direction is applied to the Condition Parameter Block as a polar direction.
<b>Direction</b>	<p>CL – The <i>Direction</i> collection contains joint collections. Each joint collection contains axes or vectors from the Generic Desktop page. Each axis usage is treated as a Dynamic Variable (DV). If the values are in Cartesian coordinates then axes usages X, Y or Z or vector usages Vx, Vy or Vz will be declared in this collection as normalized values.</p> <p>If the values are in polar coordinates then axes usages Rx, Ry or</p>

Rz or vector usages Vbrx, Vbry or Vbrz will be declared in this collection as normalized values.

Start Delay

DV - The start delay interval, for the playback of effect. The interval is the time between an Op Effect Start or Op Effect Start Solo command and initiation of the effect. If no delay is desired this value should be set to 0. Units are used to define the time base.

#### **5.1.1.1 Type Specific Block Offsets**

The number of Type Specific Block Handles declared in an Effect report is determined by the following rules:

- 1) Custom Downloaded Effects do not use Envelope Blocks.
- 2) Effects that use Condition Blocks do not use Envelope Blocks.
- 3) Effects that use Condition Blocks require 1 Condition Block per axis.
- 4) Effects that do not use Condition Blocks use 1 Parameter Block and an optional Envelope Block.
- 5) Rules 2 through 4 apply to each declared joint.

The number of Type Specific Block Handles equals the number of joints times the number axes or 2, which ever is greater. The minimum of 2 will support an Envelope and one additional parameter block.

Number of Type Specific Block Offset = joints \* (axes or 2, which ever is greater)

The following table lists the type of parameter block that a Type Specific Block Offsets point to for each Effect Type. The table assumes that the device has declared a single joint with two axes, X and Y. If the device had 3 axes then a third Type Specific Block Offset would have been declared. The third Offset would only be used for those Effect Types that require a Condition Block for the axis.



**Table 3: Effect Types**

Effect Type	Type Specific Block Offset 1	Type Specific Block Offset 2
ET Constant Force	Magnitude	Envelope
ET Ramp	Ramp	Envelope
ET Custom Force Data	Download Custom Force	NA
ET Square	Periodic	Envelope
ET Sine	Periodic	Envelope
ET Triangle	Periodic	Envelope
ET Sawtooth Up	Periodic	Envelope
ET Sawtooth Down	Periodic	Envelope
ET Spring	Condition (X Axis)	Condition (Y Axis)
ET Damper	Condition (X Axis)	Condition (Y Axis)
ET Inertia	Condition (X Axis)	Condition (Y Axis)
ET Friction	Condition (X Axis)	Condition (Y Axis)

**Type Specific Block Offset** CL – This collection Ordinal usages to distinguish the individual Type Specific Block Offset fields. Depending on the effect type one or more fields will be valid. See Effect Types for a discussion of the required Offsets.

The Parameter Blocks pointed to by these offsets must remain valid throughout the life of the effect.

**Block Type** NArY – This collection contains an array item that lists the parameter Block Types supported by the device. The Selectors in this Named Array are the various PID report usages: Set Effect Report, Set Envelope Report, Set Condition Report, Set Periodic Report, Set Constant Report and Set Ramp Force Report. These overloaded usages identify the respective parameter block report.

## 5.2 Envelope Block Parameters

The Envelope Block describes the envelope to be used by an effect. Note that not all effect types use envelopes. The minimal Envelope parameter block must contain (Envelope) Parameter Block Index, Attack Level, Attack Time, Fade Level and Fade Time values.

**Set Envelope Report** CL – This usage applies the logical collection that identifies the report associated with setting the envelope parameters of a PID. This collection must contain the *Attack Level*, *Attack Time*, *Fade Level* and *Fade Time* usages.

Attack Level DV - Normalized amplitude for the start of the envelope, from the baseline.

Attack Time DV - The transition time to reach the sustain level. Units are used to define the time base.

Fade Level DV - Normalized amplitude to end the envelope, from baseline.

Fade Time DV - The fade time to reach the fade level. Units are used to define the time base.

### 5.3 Condition Block Parameters

The minimal Condition parameter block must contain (Condition) Parameter Block Index, Center Point Offset, Positive Coefficient, Negative Coefficient, Positive Saturation, Negative Saturation and Dead Band values.

Different types of conditions will interpret the parameters differently, but the basic idea is that force resulting from a condition is equal to  $A(q - q_0)$  where A is a scaling coefficient, q is some metric, and  $q_0$  is the neutral value for that metric.

The simplified formula given above must be adjusted if a nonzero dead band is provided. If the metric is less than CP Offset - Dead Band, then the resulting force is given by the following formula:

$$\text{force} = \text{Negative Coefficient} * (q - (\text{CP Offset} - \text{Dead Band}))$$

Similarly, if the metric is greater than CP Offset + Dead Band, then the resulting force is given by the following formula:

$$\text{force} = \text{Positive Coefficient} * (q - (\text{CP Offset} + \text{Dead Band}))$$

A spring condition uses axis position as the metric.

A damper condition uses axis velocity as the metric.

An inertia condition uses axis acceleration as the metric.

If the number of Condition report blocks is equal to the number of axes for the effect, then the first report block applies to the first axis, the second applies to the second axis, and so on. For example, a two-axis spring condition with CP Offset set to zero in both Condition report blocks would have the same effect as the joystick self-centering spring. When a condition is defined for each axis in this way, the effect must not be rotated.

If there is a single Condition report block for an effect with more than one axis, then the direction along which the parameters of the Condition report block are in effect is determined by the direction parameters passed in the Direction field of the Effect report block. For example, a friction condition rotated 45 degrees (in polar coordinates) would resist joystick motion in the northeast-southwest direction but would have no effect on joystick motion in the northwest-southeast direction.

<b>Set Condition Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the effect condition parameters. This collection must contain the <i>CP Offset</i> , <i>Positive Coefficient</i> , <i>Negative Coefficient</i> , <i>Positive Saturation</i> , <i>Negative Saturation</i> and <i>Dead Band</i> usages.
CP Offset	DV – The Normalized Center Point Offset. Offset from axis 0 position.
Positive Coefficient	DV - The Normalized coefficient constant on the positive side of the neutral position.
Negative Coefficient	DV - The Normalized coefficient constant on the negative side of the neutral position.
Positive Saturation	DV - The Normalized maximum positive force output.

Negative Saturation	DV - The Normalized maximum negative force output.
Dead Band	DV - The region around <i>CP Offset</i> where the condition is not active. In other words, the condition is not active between <i>Offset - Dead Band</i> and <i>Offset + Dead Band</i> . This value is Normalized.

## 5.4 Custom Forces

Custom Forces are analogous to audio sound files, they are “played” by the device. The analog continues in that, stereo audio files consist of samples of interleaved right and left channel data. Custom Forces consist of samples of interleaved axis data. Downloaded Custom Forces are stored in the device and then played back in real-time by the device from local memory.

An Effect block is created of type *ET Downloaded Custom Effect*. This effect is similar to a Periodic effect however the waveform, rather than being predefined by ET Sine, ET Sawtooth, etc., is supplied by the host. The waveform is downloaded into the devices Pool. To provide this functionality the Download Custom Force Parameter Block must be declared.

To provide this functionality three reports must be declared:

- 1) *Download Force Sample* to define the format of the downloaded data.
- 2) *Custom Force Data Report* to define to the report that moves the data to the device.
- 3) *Set Custom Force Report* to define the parameter block that is pointed to by the Effect block.

A *Download Force Sample* collection is declared to identify the format of the effect sample that the device is capable of playing from memory. I.e. Two 8-bit samples of X and Y data, three 16-bit samples of X, Y and Z data, etc. The Download Force Sample is defined with it’s own Report ID. This report is never actually transferred to the device, it is simply used to declare the format of the downloaded samples.

The *Custom Force Data Report* is used to load the samples into the device. This report consists of 3 parts: *Parameter Block Offset*, *Generic Desktop: Byte Count* and the *Custom Force Data* itself. The *Parameter Block Offset* identifies the byte offset into the pool to start loading the *Custom Force Data*. The *Byte Count* identifies how many bytes of the *Custom Force Data* buffer are to be copied onto the pool.

The *Set Custom Force Report* defines the parameter block that is pointed to by the Type Specific Block Offset in the Effect parameter block. The Set Custom Force parameter block identifies the start offset in the pool and the size of the effect in samples of the data that has been loaded by the Custom Force Data Reports.

### 5.4.1 Sample Definition

A Sample declares the format of Custom Force data that the device is capable of handling. A sample consists of one or more axes. The size of the sample is a function of the number of axes and the size of their respective bit fields.

A *Force Sample* usage is a collection, which contains joint collections. Each joint collection contains axes from the Generic Desktop page or a *Normal* usage. These usages identify the number and types of axes in a sample. Each axis usage is treated as an Dynamic Variable (DV).

If the values are in Cartesian coordinates then linear axis (X, Y, Z) Axis or vector (Vx, Vy, Vz) usages will be declared in this collection.

If the values are in polar coordinates then a rotational axis (Rx, Ry, Rz) or vector (Vbrx, Vbry, Vbrz) usages will be declared in this collection.

**Download Force Sample**

CL – The *Download Force Sample* collection contains joint collections. This usage identifies the format of the force data sample that the host must store in the Pool for later playback by the device.

### 5.4.2 Data Downloading

Downloading of custom effect data is performed using the *Custom Force Data Report*. This report consists of 3 parts: *Parameter Block Offset*, *Generic Desktop: Byte Count* and the *Custom Force Data* itself.

A Downloaded Custom Force parameter block consumes *Byte Count* bytes in the pool. This allows a Downloaded Custom Force parameter blocks to be loaded sequentially in the Pool, creating a custom force data buffer larger than the Report Count defined for the *Custom Force Data* field.

**Custom Force Data Report**

CL – This usage applies the logical collection that identifies the report associated with loading Custom Force data. This collection must contain *Parameter Block Offset*, *Generic Desktop: Byte Count* and *Custom Force Data usages*.

Custom Force Data

DV – The *Custom Force Data* usage is attached to a Buffered Bytes item that contains the data to be transferred. Units may be used to define the force value. If no units are declared then assume a normalized value.

Custom Force Vendor Defined Data

DV – The *Custom Force Vendor Defined Data* usage is attached to a Buffered Bytes item that contains the data to be transferred. The format of the data is vendor specific so a *Download Force Sample* is not required.

Samples are packed. A sample might not be an integral number of bytes in length, or the size of the Custom Force Data field may not be an integral number of samples in length, therefore a sample may not end on a byte boundary in the Custom Force Data buffer. If a custom force data buffer required in the device is larger than the Custom Force Data field, then sequential Custom Force Data buffers will have to be loaded into the Pool, and a sample may span Custom Force Data buffers.

For example, assume the following:

- 1) The Download Force Sample is defined as 3 axes (X, Y, Z) of 8 bit data, resulting in a 24 bit sample.
- 2) The Custom Force Data Field is 256 bytes in size.
- 3) The driver wants to load 100 samples (300 bytes).

The driver would send two *Download Custom Data Reports*, one with the *Byte Count* = 256, *Data Offset* = X and another with the *Byte Count* = 44, *Data Offset* = X+256. The first transfer would contain 85.33 samples and the second 14.66. This would form a contiguous 300-byte data buffer in the device's memory.

### 5.4.3 Custom Force Block Parameters

The minimal Downloaded Custom Force parameter block must contain (Downloaded Custom Force) Parameter Block Offset, Downloaded Custom Force Block Offset and Sample Count values.

**Set Custom Force Report**

CL – This usage applies the logical collection that identifies the report associated with setting the downloaded custom force effect parameters.

Custom Force Data Offset	DV – Offset into the Pool where the Downloaded Custom Force data starts.
Sample Count	DV – The number of Downloaded Custom Force samples in a single period of the effect.

### 5.5 Periodic Block Parameters

The minimal Periodic parameter block must contain (Periodic) Parameter Block Index, Magnitude, Offset, Phase and Period values. Once started a periodic effect will loop every *Period* until a Pause or Stop command is received.

<b>Set Periodic Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the periodic effect parameters. This collection must contain the <i>Magnitude</i> , <i>Offset</i> , <i>Phase</i> and <i>Period</i> usages.
Offset	DV - Normalized baseline offset. The range of forces generated by the effect will be Offset - Magnitude to Offset + Magnitude. The value of the Offset member is also the baseline for any envelope that is applied to the effect.
Magnitude	DV - The Normalized magnitude of the effect. If an envelope is applied to this effect, then the value represents the magnitude of the envelope. If no envelope is applied, then the value represents the amplitude of the entire effect.
Phase	DV - Determines the position in the wave that playback begins. This is a Normalized value between 0 and 360 degrees. The angular increment is defined by the resolution of the field.  A device driver may not provide support for all values of Phase. In this case the value will be rounded off to the nearest supported value.
Period	DV - The period of the effect. Units are used to define the time base.

### 5.6 Constant Force Block Parameters

The minimal Ramp Force parameter block must contain (Constant Force) Parameter Block Index and Magnitude values.

<b>Set Constant Force Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the Constant Force effect parameters. This collection must contain the <i>Magnitude</i> usages.
----------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 5.7 Ramp Force Block Parameters

The minimal Ramp Force parameter block must contain (Ramp Force) Parameter Block Index, Ramp Start and Ramp End values.

Note: The Duration for a ramp force effect cannot be INFINITE.

<b>Set Ramp Force Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the Ramp Force effect parameters.
------------------------------	-------------------------------------------------------------------------------------------------------------------------------------

	This collection must contain the <i>Ramp Start</i> and <i>Ramp End</i> usages.
Ramp Start	DV - The Normalized magnitude at the start of the effect.
Ramp End	DV - The Normalized magnitude at the end of the effect.

### 5.8 Vendor Defined Parameter Blocks

In a Vendor Defined Parameter Block a vendor can use:

- 1) Usages defined in the PID spec.; Offset, Magnitude, etc.
- 2) Vendor define usages from their own usage page.

To be consistent with this document, vendor defined parameter blocks should be defined using a similar format.

- 1) Using a unique Report ID for each vendor defined report
- 2) Wrapping the report in a Set xxx Report logical collection, where the Set xxx Report is the vendor defined name on their usage page.
- 3) Include a Parameter Block Offset or a Block Handle.

### 5.9 Effect Operations

This report is used to control the operation of effects.

<b>Effect Operation Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the effect operational parameters. This collection must contain <i>Effect Block Index</i> , <i>Effect Operation</i> and <i>Loop Count</i> usages.
<b>Effect Operation</b>	NArY - Operation to perform on the effect identified by the Effect ID. This named array that contains Operation (Op) selectors.
Op Effect Start	Sel - Start the effect identified by the Effect Handle.
Op Effect Start Solo	Sel - Start the effect identified by the Effect Handle and stop all other effects.
Op Effect Stop	Sel - Stop the effect identified by the Effect Handle.
Loop Count	DV – The number of times the device will repeat the operation. i.e. a Stop command is not required. A INFINITE value implies “loop for ever”.

### 5.10 Device Gain

This report is used to control the gain of the overall device.

<b>Device Gain Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the device gain parameters. This collection can contain a <i>Device Gain</i> usage.
Device Gain	DV - The device gain to be applied to all effects on the device. The device gain is a normalized scaling factor applied to all magnitudes and envelopes of an effect, scaling the final output of the device.

## 5.11 Parameter Block Management

Parameter Blocks are stored in the PID class device. A vendor can choose whether to let the system driver manage the parameter block pool or manage it themselves.

When the driver manages the Parameter Block Pool, the device simply provides a block of memory (the Pool) where the driver loads Parameter Blocks. All decisions as to where to load Parameter Blocks, are made by the driver. The driver maintains a list of all Parameter Blocks that are currently stored in the pool and will perform any garbage collection that is required.

If a device manages its own pool then the driver simply passes Parameter Blocks to the device and receives either a negative acknowledgement if there was no room for the block or a positive acknowledgement and a handle for the block if there was room. The handle is used for all future references to the Parameter Block.

### 5.11.1 Driver Managed

A driver reads the PID Pool Report to identify the size of the RAM and ROM spaces on the device, and the size of the individual parameter blocks (Reports) to be stored there. The RAM Pool size identifies the amount of read/write storage available for parameter blocks. The pool is logically divided into two spaces: the Effect Parameter Block array and the general Pool. The driver manages the space allocation in the Pool and all parameter block types except Effect Parameter Blocks are referenced by their byte offset from the beginning of the Pool.

For efficiency, Effect Parameter Blocks are handled slightly differently. An array of Effect Parameter Blocks always starts at the first location of the Pool. Effect Parameter Blocks are accessed by their index in this array. If a driver determines it wants to store 10 Effects in the device at one time and the Effect Parameter Block is 12 bytes long then the first 120 bytes of the Pool will be occupied by the Effect Parameter Block array. The remainder of the pool will be allocated to the other types of parameter blocks. This approach allows a device with a large Pool (1 MByte) to have a compact Operation field in the PID Operation Output Report.

ROM based effects are also supported. ROM based parameter blocks are accessed the same way that RAM based effects are. The only difference is that they are read-only. A driver can read in the ROM parameter blocks to determine the effects that are stored in the device.

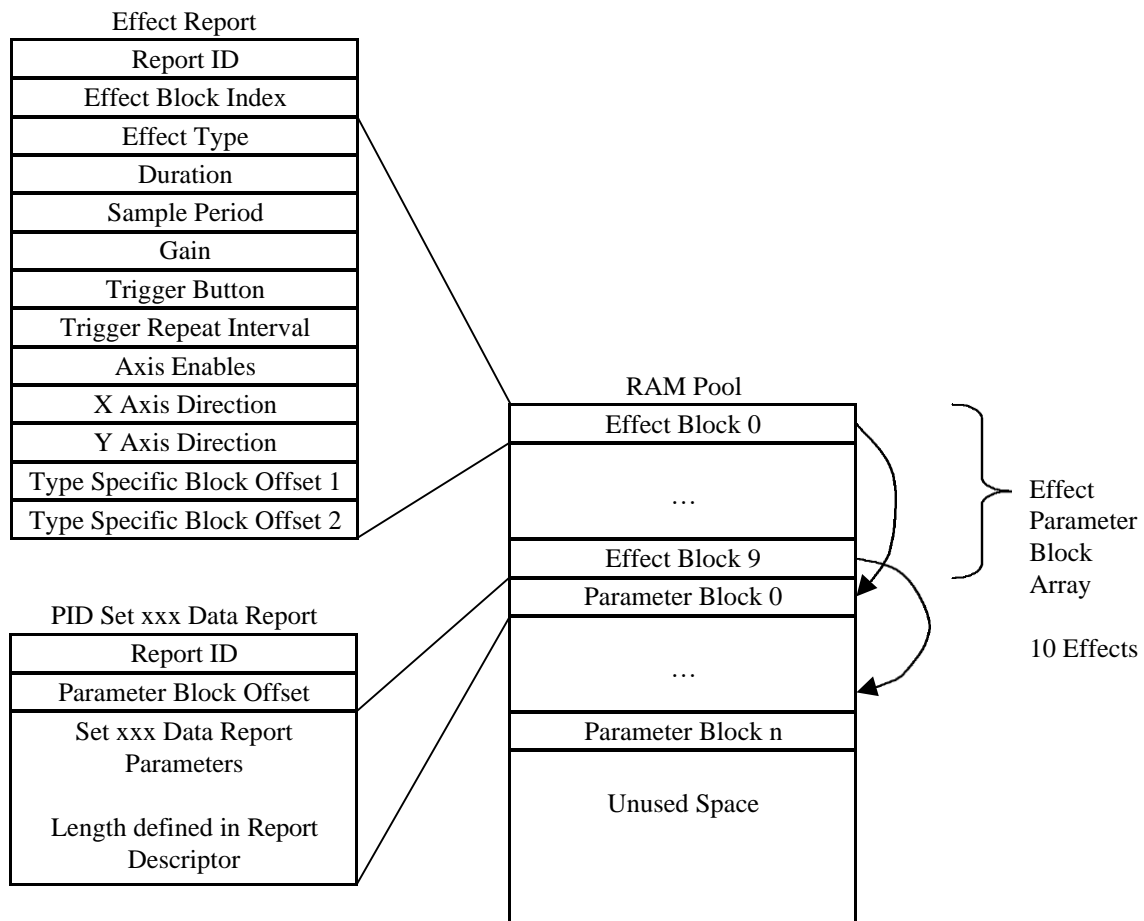
Because pool allocation for parameter blocks is managed by the driver, the driver will know which types of blocks it has loaded and where. The device uses the Effect Type field in the Effect Parameter Block to identify the type(s) of Parameter blocks that the Type Specific Block Offsets point to.

Figure 1 assumes the RAM Pool is initially empty. The Effect Parameter Block Array always starts at the beginning of memory. The Effect Block Index field determines the location in the Effect Parameter Block Array where the Effect Parameter Block will be loaded. In this example space is allocated for 10 Effects.

Note that the space allocated for a single Effect Parameter Block in the RAM Pool is the size of the Effect Report **minus** the size of the Report ID, Effect Block Index and ROM Flag fields if there is one. The same is true for all other types of Parameter Blocks however they can be placed anywhere in the available Pool space. The Report ID identifies the size and type of the Parameter Block while the Effect Block Index and ROM Flag fields identify the offset of the Parameter Block in the Pool. This information is stripped from the Parameter Block before it is stored in memory to save space.

Figure 1 shows the Parameter Blocks immediately following the Effect Parameter Block Area. The arrows on the right show how the Type Specific Block Offset of a Effect Parameter Block will point to any Parameter Blocks associated with that Effects Parameter Block.

**Figure 1: Report Allocation in the Pool**



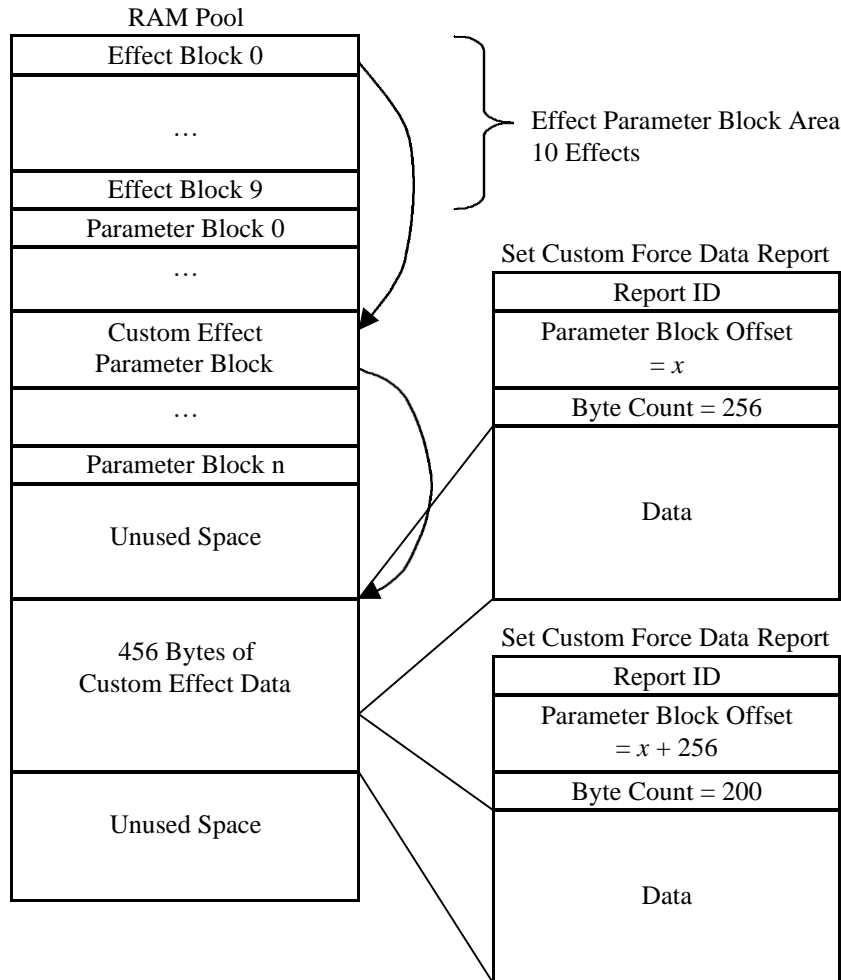


Custom Effect Data is stored in memory as a large contiguous block that is formed by concatenating the Data portion of Set Custom Force Data Reports. In this case the Byte Count is also stripped be for storing the data. Figure 2 shows two Set Custom Force Data Reports, the first containing 256 bytes of data and the second containing 200 bytes. The Data portions of these reports are concatenated starting at the Parameter Block Offset in the Pool. This will generate a 456 byte Custom Force Data Block in the Pool.

The arrows in the center represent how the Type Specific Block Offset of a Effect Parameter Block will point to a Custom Effect Parameter Block which in turn points to the Custom Effect Data. This all assumes that the Type of the Effect is Custom Effect.

Other than the Effect Parameter Blocks the placement of Parameter Blocks and Custom Effect Data in the Pool is determined strictly by the driver. The driver may group blocks of like size or employ other techniques to minimize fragmentation of the Pool.

**Figure 2: Custom Effect Allocation in the Pool**



The following usages are only used by devices that let the driver manage their Parameter Block pool.

<b>PID Pool Report</b>	CL – This usage applies the logical collection that identifies the report associated with setting the overall PID control parameters.
RAM Pool Size	DV - The size of the PID RAM parameter pool in bytes.
ROM Pool Size	SV - The size of the PID ROM parameter pool in bytes.
ROM Effect Block Count	SV - The number of ROM Effect Parameter blocks in the ROM parameter pool.
Simultaneous Effects Max	SV – The maximum number of simultaneous effects supported by the device. A device declares this usage to indicate the total number of effects that it can execute at one time.
Pool Alignment	SV – The block start address alignment boundary in bytes. This usage can be defined by a device so that parameter blocks start on 16-bit (Pool Alignment = 2) or 32-bit boundaries (Pool Alignment = 4) etc. If undefined then Pool Alignment = 1 is assumed.
<b>Parameter Block Size</b>	CL – This usage only applies to a Driver Managed pool. It is a logical collection that identifies the size that host should allocate for each fixed size Parameter Block in the Pool. That is, any Report collection that contains an <i>Parameter Block Index</i> usage declaration and does not declare a <i>Generic Desktop:Byte Count</i> .
	<p>This collection changes the meaning of the Usages that it contains. The possible included usages are: the <i>Set Effect Report</i>, <i>Set Envelope Report</i>, <i>Set Condition Report</i>, <i>Set Custom Force Report</i>, <i>Set Periodic Report</i>, <i>Set Constant Force Report</i> and <i>Set Ramp Force Report</i> Logical Collection (LC) usages. These LC usages each define specific reports that require space in the Pool. When found in a <i>Parameter Block Size</i> collection the type of these usages change from a Logical Collection (LC) to a Static Variable (SV), which define the size, in bytes, of the respective report.</p> <p>Note that the <i>Custom Force Data Report</i> normally defines a variable sized parameter block that can be loaded in the Pool. A <i>Generic Desktop: Byte Count</i> usage is used to define the size of the parameter block dynamically.</p>
Device Managed Pool	SF - This usage will be set to 1 for devices that manage their own memory pool, and 0 for those that support driver managed pool.
Shared Parameter Blocks	SF - This usage will be 1 for devices that support a single parameter block to be shared between multiple effects. Value will be zero for devices which have one parameter block set for each effect block.

When the Pool gets fragmented the driver must perform garbage collection to compact the valid Parameter Blocks and to free the unused ones. The PID Pool Move Report is issued to the device to copy data from one place to another in the Pool. This also allows the device to move any private data that it may have associated with a Parameter Block.

<b>PID Pool Move Report</b>	CL – This usage applies the logical collection that identifies the report associated with moving Parameter Blocks in the Pool for
-----------------------------	-----------------------------------------------------------------------------------------------------------------------------------

garbage collection. The host must update any effect specific references. Moving a parameter block of an effect that is playing will result in undefined behavior.

Move Source	DV – The source offset of the Parameter Block.
Move Destination	DV – The destination offset of the Parameter Block.
Move Length	DV – The size of the Parameter Block to move in bytes.

Note: The *Data Offset* fields must be declared large enough to access the full extent of the RAM and ROM Pools.

### 5.11.2 Device Managed

In the following text, Type Specific Block Handle and Effect Block Index will be used interchangeably.

When the device manages Parameter Blocks, Pool addresses are not visible to the driver. Instead the host must request a handle for an effect by sending a **Create New Effect Report** with the requested effect type to the device. Upon receiving this report the device allocates memory for the effect and allocates an *Effect Block Index* (or handle) for the effect. If an *ET Custom Force Data* effect is created then this report must also contain the size of the custom report to be created. A *Generic Desptop:Byte Count* usage is used to define the size of the custom force data report in bytes. For the effects other than *ET Custom Force Data*, the device will ignore the *Generic Desptop:Byte Count* field in a *Create New Effect Report*.

After sending the *Create New Effect Report*, the host must then send a HID Get\_Report request. The device will respond with a *PID Block Load Report* containing the newly assigned *Effect Block Index*. All subsequent references to the effect block by the host will use this index.

After the host has received and verified the contents of the *PID Block Load Report*, it will then send a *Set Effect Report* containing the *Effect Block Index* that was returned by the *PID Block Load Report* along with the other parameters required to define the effect.

The *PID Block Load Report* will indicate in the *RAM Pool Available* field, the amount of memory remaining in the device so the host can anticipate whether the device has enough memory pool remaining for additional effects or for custom effect data. An example of report descriptor for *Create New Effect Report* and *PID Block Load Report* is given in Section 9.

The *Set Effect Report* uses *Type Specific Block Offsets* to identify additional parameter blocks that relate to the effect. Table 3 lists the parameter blocks pointed to by the *Type Specific Block Offsets* as function of effect type. In most cases, each *Type Specific Block Offsets* points to a different type of parameter block so there is no ambiguity as to which parameter block a "Set xxx Report" command refers to. However, the Spring, Damper, Inertia and Friction effects each require the definition of two condition parameter blocks. In this case the Parameter Block Offset becomes an index in to the *Type Specific Block Offsets* (rather than a pointer as in the driver managed case). The order of axis declaration in the Axes Enable collection defines the index used for the Parameter Block Offset. See the example in section 9 for more detailed information.

The following usage's are only used devices that manage their own memory pool.

<b>Create New Effect Report</b>	CL - this usage applies the logical collection that identifies the report associated with creating a new effect for the device that manages its own memory.
---------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>PID Block Load Report</b>	CL - This usage applies the logical collection that identifies the report associated with a device's response to a Create New Effect Report.
<b>Block Load Status</b>	NArY - Identifies the completion status of a Set PB Report request. This named array contains Block Load selectors.
Block Load Success	Sel - The Set Effect Report command successfully completed.
Block Load Full	Sel - The Set Effect Report command failed to complete because there was no room in the device's memory pool.
Block Load Error	Sel - The Set Effect Report command failed to complete because an error occurred in the device.
RAM Pool Available	DV - The number of bytes remaining the in the devices memory pool.
<b>PID Block Free Report</b>	CL – This usage applies the logical collection that identifies the report associated with freeing a Parameter Block in a device. This report will contain the <i>Block Handle</i> to free.
<b>Type Specific Block Handle</b>	CL - This collection uses Ordinal usages to distinguish the individual Type Specific Block Handle fields. Depending on the effect type one or more fields will be valid. See Effect Types for a discussion of the required Handles.  The Parameter Blocks referenced by these handles must remain valid throughout the life of the effect.

### 5.12 PID State

<b>PID State Report</b>	CL – This usage applies the logical collection that identifies the report associated with identifying the PID state.
<b>PID Effect State</b>	NArY –The state of the effect identified by the Effect Handle. This is a named array that contains Effect State (PES) selectors.
Effect Playing	DF – his usage indicates the current play state for effect on the device. If this is set, then the effect is currently playing on the device. If this is cleared, then the effect is not currently playing (it is stopped).
Device Paused	DF – Indicates that all effects on the device are paused. All effects on the device are paused at the current time step.
Actuators Enabled	DF – Indicates that the device's actuators are enabled.
Safety Switch	DF – A control indicating the state of the Safety switch on the device. If the <i>Safety Flag</i> is set and the device is usable. (Read-only)
Actuator Override Switch	DF – This is an actuator override switch available to the user. The user enables the device's actuators if this flag is set.
Actuator Power	OOO - When read back this usage indicates the current actuator power status. (Read-only)

### 5.13 PID Device Control

<b>PID Device Control Report</b>	CL – This usage applies to the logical collection that identifies the report associated with executing device control commands. These are the commands that have effect over the entire device and require no parameters.
<b>PID Device Control</b>	NAry – This usage controls the overall state of the device. This is named array that contains Device Control (DC) selectors.
DC Enable Actuators	Sel – Enable all device actuators.
DC Disable Actuators	Sel – Disable all the device actuators.
DC Stop All Effects	Sel - Issues a stop on every running effect.
DC Reset	Sel – Clears any device paused condition, enables all actuators and clears all effects from memory.
DC Pause	Sel – The all effects on the device are paused at the current time step.
DC Continue	Sel – The all effects that running when the device was paused are restarted from their last time step.

## 6. PID Example

This is an example of a joystick with force feedback capabilities. This device does support Effects, . isochronous and downloaded Custom Force operations. The RAM Pool Size is 256 bytes and the ROM Pool size is 32 K bytes.

In this example the Generic Desktop Pointer is considered a joint and items related to that joint are tied together in Generic: Desktop Pointer physical collections. You will see this demonstrated in the input report axes, Axes Enable fields and the Custom Effect Force axes.

This device supports all Effect Types except Friction.

There are 2 Input Reports: PID Data (stick and button state) and PID State (effect state). 10 output reports: Set Effect, Set Envelope, Set Condition, Set Periodic, Set Constant Force, Set Ramp Force, Set Custom Force, Download Force Sample, Set Custom Force and Effect Operation. And 3 Feature reports: Pool, State and Device Gain.

```

USAGE_PAGE (Generic Desktop)          05 01
LOGICAL_MINIMUM (0)                   15 00
USAGE (Joystick)                       09 04
COLLECTION (Application)               A1 01

```

;**Start Joystick input definition**

```

REPORT_ID (1)                          85 01
USAGE_PAGE (Simulation Controls)        05 02
USAGE (Throttle)                        09 BB
LOGICAL_MINIMUM (-127)                  15 81
LOGICAL_MAXIMUM (127)                   25 7F
REPORT_SIZE (8)                         75 08
REPORT_COUNT (1)                        95 01
INPUT (Data,Var,Abs)                   81 02
USAGE_PAGE (Generic Desktop)           05 01

```

;**Define the axes**

```

USAGE (Pointer)                         09 01
COLLECTION (Physical)                   A1 00
  USAGE (X)                              09 30
  USAGE (Y)                              09 31
  REPORT_COUNT (2)                       95 02
  INPUT (Data,Var,Abs)                   81 02
END_COLLECTION                           C0

```

;**Define the hat switch**

```

USAGE (Hat switch)                      09 39
LOGICAL_MINIMUM (0)                     15 00
LOGICAL_MAXIMUM (3)                     25 03
PHYSICAL_MINIMUM (0)                    35 00
PHYSICAL_MAXIMUM (270)                  46 0E 01
UNIT (Eng Rot:Angular Pos)              66 14 00
REPORT_SIZE (4)                         75 04
REPORT_COUNT (1)                        95 01
INPUT (Data,Var,Abs)                   81 02
PHYSICAL_MAXIMUM (0)                    45 00
UNIT (None)                             66 00 00

```

**;Define the buttons**

```

USAGE_PAGE (Button)                05 09
USAGE_MINIMUM (Button 1)           19 01
USAGE_MAXIMUM (Button 4)           29 04
LOGICAL_MAXIMUM (1)                25 01
REPORT_COUNT (4)                   95 04
REPORT_SIZE (1)                    75 01
PHYSICAL_MINIMUM (0)               35 00
INPUT (Data,Var,Abs)               81 02

```

**;End Joystick Input definition**

**;Start Force Feedback command definitions**

```

USAGE_PAGE (Physical Interface)     05 0F

```

**;Effect Report Definition**

```

USAGE (Set Effect Report)           09 21
COLLECTION (Logical)                A1 02
  USAGE (Effect Block Index)        09 22
  LOGICAL_MAXIMUM (127)             25 7F
  REPORT_SIZE (7)                   75 07
  REPORT_COUNT (1)                  95 01
  OUTPUT (Data,Var,Abs)             91 02

  USAGE (ROM Flag)                  09 24
  LOGICAL_MAXIMUM (1)               25 01
  REPORT_SIZE (1)                   75 01
  OUTPUT (Data,Var,Abs)             91 02

```

**;Define the available effect types. Effect Type is a named array that will  
; accept any of the ET usages listed.**

```

USAGE (Effect Type)                 09 25
COLLECTION (Logical)                A1 02
  USAGE (ET Constant Force)         09 26
  USAGE (ET Ramp)                   09 27
  USAGE (ET Square)                 09 30
  USAGE (ET Sine)                   09 31
  USAGE (ET Triangle)               09 32
  USAGE (ET Sawtooth Up)            09 33
  USAGE (ET Sawtooth Down)          09 34
  USAGE (ET Spring)                 09 40
  USAGE (ET Damper)                 09 41
  USAGE (ET Inertia)                09 42
  LOGICAL_MINIMUM (1)               15 01
  LOGICAL_MAXIMUM (10)              25 0A
  REPORT_SIZE (8)                   75 08
  OUTPUT (Data,Ary,Abs)             91 00
END_COLLECTION                       C0

  USAGE (Duration)                  09 50
  USAGE (Trigger Repeat Interval)    09 54
  LOGICAL_MINIMUM (0)               15 00
  LOGICAL_MAXIMUM (10000)           26 10 27
  PHYSICAL_MAXIMUM (10000)          46 10 27
  REPORT_SIZE (16)                  75 10
  UNIT (Eng Lin:Time)                66 03 10
  UNIT_EXPONENT (-3)                 55 0D

```

## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

```

REPORT_COUNT (2)                95 02
OUTPUT (Data,Var,Abs)          91 02

UNIT_EXPONENT (-6)              55 0A
USAGE (Sample Period)          09 51
REPORT_COUNT (1)                95 01
OUTPUT (Data,Var,Abs)          91 02
PHYSICAL_MAXIMUM (0)           45 00
UNIT_EXPONENT (0)              55 00
UNIT (None)                     65 00

USAGE (Gain)                    09 52
USAGE (Trigger Button)          09 53
LOGICAL_MAXIMUM (127)           25 7F
REPORT_SIZE (8)                 75 08
REPORT_COUNT (2)                95 02
OUTPUT (Data,Var,Abs)          91 02

USAGE (Axes Enable) ; Tie these axes to the stick 09 55
COLLECTION (Logical)            A1 02
  USAGE_PAGE (Generic Desktop)  05 01
  USAGE (Pointer)                09 01
  COLLECTION (Physical)          A1 00
    USAGE (X)                    09 30
    USAGE (Y)                    09 31
    LOGICAL_MAXIMUM (1)          25 01
    REPORT_SIZE (1)              75 01
    REPORT_COUNT (2)             95 02
    OUTPUT (Data,Var,Abs)        91 02
  END_COLLECTION                 C0
END_COLLECTION                   C0

REPORT_COUNT (6) ; 6-bit pad    95 06
OUTPUT (Cnst,Var,Abs)          91 03

USAGE_PAGE (Physical Interface) 05 0F
USAGE (Direction)              09 57
COLLECTION (Logical)            A1 02
  USAGE_PAGE (Generic Desktop)  05 01
  USAGE (Pointer) ; Tie these axes to the stick 09 01
  COLLECTION (Physical)          A1 00
    USAGE (X)                    09 30
    USAGE (Y)                    09 31
    LOGICAL_MINIMUM (0)          15 00
    LOGICAL_MAXIMUM (255)        26 FF 00
    PHYSICAL_MAXIMUM (360)       46 68 01
    UNIT (Eng Rot:Angular Pos)   66 14 00
    REPORT_SIZE (8)              75 08
    REPORT_COUNT (2)             95 02
    OUTPUT (Data,Var,Abs)        91 02
    UNIT (None)                  65 00
    PHYSICAL_MAXIMUM (0)         45 00
  END_COLLECTION                 C0
END_COLLECTION                   C0

USAGE_PAGE (Physical Interface) 05 0F
USAGE (Type Specific Block Offset) 09 58
COLLECTION (Logical)            A1 02

```



## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

```
    USAGE (Ordinals:Instance 1)                0B 01 00 0A 00
    USAGE (Ordinals:Instance 2)                0B 02 00 0A 00
    LOGICAL_MAXIMUM (32765) ; 32K RAM or ROM max. 26 FD 7F
    REPORT_SIZE (16)                           75 10
    REPORT_COUNT (2)                           95 02
    OUTPUT (Data,Var,Abs)                     91 02
    END_COLLECTION                             C0
END_COLLECTION                               C0
```

### ;Envelope Report Definition

```
    USAGE (Set Envelope Report)                09 5A
    COLLECTION (Logical)                      A1 02
    REPORT_ID (2)                             85 02
    USAGE (Parameter Block Offset)            09 23
    LOGICAL_MAXIMUM (32765) ; 32K RAM or ROM max 26 FD 7F
    REPORT_SIZE (15)                          75 0F
    REPORT_COUNT (1)                          95 01
    OUTPUT (Data,Var,Abs)                    91 02

    USAGE (ROM Flag)                          09 24
    LOGICAL_MAXIMUM (1)                       25 01
    REPORT_SIZE (1)                           75 01
    OUTPUT (Data,Var,Abs)                    91 02

    USAGE (Attack Level)                      09 5B
    USAGE (Fade Level)                        09 5D
    LOGICAL_MAXIMUM (255)                     26 FF 00
    REPORT_SIZE (8)                           75 08
    REPORT_COUNT (2)                          95 02
    OUTPUT (Data,Var,Abs)                    91 02

    USAGE (Attack Time)                       09 5C
    USAGE (Fade Time)                         09 5E
    LOGICAL_MAXIMUM (10000)                   26 10 27
    PHYSICAL_MAXIMUM (10000)                  46 10 27
    UNIT (Eng Lin:Time)                       66 03 10
    UNIT_EXPONENT (-3)                        55 0D
    REPORT_SIZE (16)                          75 10
    OUTPUT (Data,Var,Abs)                    91 02
    PHYSICAL_MAXIMUM (0)                      45 00
    UNIT (None)                               65 00
    UNIT_EXPONENT (0)                         55 00
    END_COLLECTION                             C0
```

### ;Condition Report Definition

```
    USAGE (Set Condition Report)              09 5F
    COLLECTION (Logical)                      A1 02
    REPORT_ID (3)                             85 03
    USAGE (Parameter Block Offset)            09 23
    LOGICAL_MAXIMUM (32765)                   26 FD 7F
    REPORT_SIZE (15)                          75 0F
    REPORT_COUNT (1)                          95 01
    OUTPUT (Data,Var,Abs)                    91 02

    USAGE (ROM Flag)                          09 24
    LOGICAL_MAXIMUM (1)                       25 01
    REPORT_SIZE (1)                           75 01
    OUTPUT (Data,Var,Abs)                    91 02
```

## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

```
USAGE (CP Offset) 09 60
USAGE (Positive Coefficient) 09 61
USAGE (Negative Coefficient) 09 62
USAGE (Positive Saturation) 09 63
USAGE (Negative Saturation) 09 64
USAGE (Dead Band) 09 65
LOGICAL_MAXIMUM (255) 26 FF 00
REPORT_SIZE (8) 75 08
REPORT_COUNT (6) 95 06
OUTPUT (Data,Var,Abs) 91 02
END_COLLECTION C0
```

### ;Periodic Report Definition

```
USAGE (Set Periodic Report) 09 6E
COLLECTION (Logical) A1 02
  REPORT_ID (4) 85 04
  USAGE (Parameter Block Offset) 09 23
  LOGICAL_MAXIMUM (32765) 26 FD 7F
  REPORT_SIZE (15) 75 0F
  REPORT_COUNT (1) 95 01
  OUTPUT (Data,Var,Abs) 91 02

  USAGE (ROM Flag) 09 24
  LOGICAL_MAXIMUM (1) 25 01
  REPORT_SIZE (1) 75 01
  OUTPUT (Data,Var,Abs) 91 02

  USAGE (Magnitude) 09 70
  USAGE (Offset) 09 6F
  USAGE (Phase) 09 71
  LOGICAL_MAXIMUM (255) 26 FF 00
  REPORT_SIZE (8) 75 08
  REPORT_COUNT (3) 95 03
  OUTPUT (Data,Var,Abs) 91 02

  USAGE (Period) 09 72
  LOGICAL_MAXIMUM (10000) 26 10 27
  PHYSICAL_MAXIMUM (10000) 46 10 27
  UNIT (Eng Lin:Time) 66 03 10
  UNIT_EXPONENT (-3) 55 0D
  REPORT_SIZE (16) 75 10
  REPORT_COUNT (1) 95 01
  OUTPUT (Data,Var,Abs) 91 02
  PHYSICAL_MAXIMUM (0) 45 00
  UNIT (None) 65 00
  UNIT_EXPONENT (0) 55 00
END_COLLECTION C0
```

### ;Constant Force Report Definition

```
USAGE (Set Constant Force Report) 09 73
COLLECTION (Logical) A1 02
  REPORT_ID (5) 85 05
  USAGE (Parameter Block Offset) 09 23
  LOGICAL_MAXIMUM (32765) 26 FD 7F
  REPORT_SIZE (15) 75 0F
  REPORT_COUNT (1) 95 01
  OUTPUT (Data,Var,Abs) 91 02
```

```

USAGE (ROM Flag) 09 24
LOGICAL_MAXIMUM (1) 25 01
REPORT_SIZE (1) 75 01
OUTPUT (Data,Var,Abs) 91 02

USAGE (Magnitude) 09 70
LOGICAL_MAXIMUM (255) 26 FF 00
REPORT_SIZE (8) 75 08
OUTPUT (Data,Var,Abs) 91 02
END_COLLECTION C0

```

**;Ramp Force Report Definition**

```

USAGE (Set Ramp Force Report) 09 74
COLLECTION (Logical) A1 02
  REPORT_ID (6) 85 06
  USAGE (Parameter Block Offset) 09 23
  LOGICAL_MAXIMUM (32765) 26 FD 7F
  REPORT_SIZE (15) 75 0F
  REPORT_COUNT (1) 95 01
  OUTPUT (Data,Var,Abs) 91 02

USAGE (ROM Flag) 09 24
LOGICAL_MAXIMUM (1) 25 01
REPORT_SIZE (1) 75 01
OUTPUT (Data,Var,Abs) 91 02

USAGE (Ramp Start) 09 75
USAGE (Ramp End) 09 76
LOGICAL_MAXIMUM (255) 26 FF 00
REPORT_SIZE (8) 75 08
REPORT_COUNT (2) 95 02
OUTPUT (Data,Var,Abs) 91 02
END_COLLECTION C0

```

**;Custom Force Data Report Definition**

**; Downloads are always into RAM space so the ROM usage is not declared.**

```

USAGE (Custom Force Data Report) 09 68
COLLECTION (Logical) A1 02
  REPORT_ID (7) 85 07
  USAGE (Parameter Block Offset) 09 23
  LOGICAL_MAXIMUM (32765) 26 FD 7F
  REPORT_SIZE (15) 75 0F
  REPORT_COUNT (1) 95 01
  OUTPUT (Data,Var,Abs) 91 02

USAGE (Generic Desktop:Byte Count) 0B 3B 00 01 00
LOGICAL_MAXIMUM (256) 26 00 01
REPORT_SIZE (9) 75 09
OUTPUT (Data,Var,Abs) 91 02

USAGE (Custom Force Data) 09 69
LOGICAL_MAXIMUM (255) 26 FF 00
REPORT_SIZE (8) 75 08
REPORT_COUNT (256) 96 00 01
OUTPUT (Data,Var,Abs,Buf) 92 02 01
END_COLLECTION C0

```

**;Download Force Sample Definition**

```

USAGE (Download Force Sample)                09 66
COLLECTION (Logical)                          A1 02
  REPORT_ID (8)                               85 08
  USAGE_PAGE (Generic Desktop)                05 01
  USAGE (Pointer)                             09 01
  COLLECTION (Logical)                       A1 02
    USAGE (X)                                 09 30
    USAGE (Y)                                 09 31
    LOGICAL_MINIMUM (-127)                    15 81
    LOGICAL_MAXIMUM (127)                     25 7F
    REPORT_SIZE (8)                           75 08
    REPORT_COUNT (2)                           95 02
    OUTPUT (Data,Var,Abs)                     91 02
  END_COLLECTION                             C0
END_COLLECTION                                C0
USAGE_PAGE (Physical Interface)              05 0F

```

**;Define the Custom Force parameter block**

**; Custom Effects are always RAM based**

**; so ROM flags are not declared.**

```

USAGE (Set Custom Force Report)                09 6B
COLLECTION (Logical)                          A1 02
  REPORT_ID (9)                               85 09
  ; Parameter block offset in pool
  ; Custom Force data offset in pool
  USAGE (Parameter Block Offset)              09 23
  USAGE (Custom Force Data Offset)            09 6C
  USAGE (Sample Count)                        09 6D
  LOGICAL_MINIMUM (0)                         15 00
  LOGICAL_MAXIMUM (32765) ; 32K of RAM or ROM max. 26 FD 7F
  REPORT_COUNT (3)                             95 03
  REPORT_SIZE (16)                             75 10
  OUTPUT (Data,Var,Abs)                       91 02
END_COLLECTION                                C0

```

**;Effect Operation Report Definition**

```

USAGE (Effect Operation Report)                09 77
COLLECTION (Logical)                          A1 02
  REPORT_ID (10)                              85 0A
  USAGE (Effect Block Index)                  09 22
  LOGICAL_MAXIMUM (127)                       25 7F
  REPORT_SIZE (7)                             75 07
  REPORT_COUNT (1)                             95 01
  OUTPUT (Data,Var,Abs)                       91 02

  USAGE (ROM Flag)                             09 24
  LOGICAL_MAXIMUM (1)                         25 01
  REPORT_SIZE (1)                              75 01
  OUTPUT (Data,Var,Abs)                       91 02

  USAGE (Effect Operation)                    09 78
  COLLECTION (Logical)                        A1 02
    USAGE (Op Effect Start)                    09 79
    USAGE (Op Effect Start Solo)               09 7A
    USAGE (Op Effect Stop)                     09 7B
    LOGICAL_MINIMUM (1)                        15 01
    LOGICAL_MAXIMUM (3)                        25 03

```

## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

```

    REPORT_SIZE (8)                75 08
    OUTPUT (Data,ArY,Abs)         91 00
END_COLLECTION                    C0

    USAGE (Loop Count)           09 7C
    LOGICAL_MINIMUM (0)          15 00
    LOGICAL_MAXIMUM (255)        26 FF 00
    OUTPUT (Data,Var,Abs)        91 02
END_COLLECTION                    C0

```

### ;PID Pool Report Definition

```

    USAGE (PID Pool Report)       09 7F
    COLLECTION (Logical)         A1 02
    REPORT_ID (1)                85 01
    USAGE (RAM Pool Size)         09 80
    USAGE (ROM Pool Size)         09 81
    USAGE (ROM Effect Block Count) 09 82
    LOGICAL_MAXIMUM (32765)      26 FD 7F
    REPORT_COUNT (3)             95 03
    REPORT_SIZE (16)             75 10
    FEATURE (Data,Var,Abs)       B1 02

    USAGE (Parameter Block Size) 09 A8
    COLLECTION (Logical)         A1 02
    USAGE (Set Effect Report)     09 21
    USAGE (Set Envelope Report)   09 5A
    USAGE (Set Condition Report)  09 5F
    USAGE (Set Periodic Report)   09 6E
    USAGE (Set Constant Force Report) 09 73
    USAGE (Set Ramp Force Report) 09 74
    USAGE (Set Custom Force Report) 09 6B
    LOGICAL_MAXIMUM (255)        26 FF 00
    REPORT_SIZE (8)              75 08
    REPORT_COUNT (7)             95 07
    FEATURE (Data,Var,Abs)       B1 02
END_COLLECTION                    C0

    LOGICAL_MAXIMUM (1)          25 01
    REPORT_SIZE (7)              75 07
    REPORT_COUNT (1)             95 01
    FEATURE (Cnst,Var,Abs) ; 7-bit pad B1 03
    USAGE (Isoch Custom Force Enable) 09 67
    REPORT_SIZE (1)              75 01
    FEATURE (Data,Var,Abs)       B1 02
END_COLLECTION                    C0

```

### ;PID State Report Definition

```

    USAGE (PID State Report)     09 92
    COLLECTION (Logical)         A1 02
    REPORT_ID (2)                85 02
    USAGE (Effect Block Index)    09 22
    LOGICAL_MAXIMUM (127)        25 7F
    REPORT_SIZE (7)              75 07
    INPUT (Data,Var,Abs)        81 02

    USAGE (ROM Flag)             09 24
    LOGICAL_MAXIMUM (1)          25 01
    REPORT_SIZE (1)              75 01

```

## Device Class Definition for Physical Interface Devices (PID) Version 1.0

---

```
REPORT_COUNT (1)          95 01
INPUT (Data,Var,Abs)     81 02

USAGE (Effect Playing)   09 94
USAGE (Actuators Enabled) 09 A0
USAGE (Safety Switch)    09 A4
USAGE (Actuator Power)   09 A6
REPORT_SIZE (1)          75 01
REPORT_COUNT (4)         95 04
INPUT (Data,Var,Abs)     81 02

INPUT (Cnst,Var,Abs) ; 4-bit pad 81 03
END_COLLECTION           C0
```

### ;PID Device Control Report Definition

```
USAGE (PID Device Control Report) 09 95
COLLECTION (Logical)              A1 02
REPORT_ID (11)                    85 0B
  USAGE (PID Device Control)       09 96
  COLLECTION (LOGICAL)             A1 02
    USAGE (DC Enable Actuators)    09 97
    USAGE (DC Disable Actuators)   09 98
    USAGE (DC Stop All Effects)    09 99
    USAGE (DC Reset)               09 9A
    USAGE (DC Pause)               09 9B
    USAGE (DC Continue)            09 9C
  LOGICAL_MINIMUM (1)              15 01
  LOGICAL_MAXIMUM (6)              25 06
  REPORT_SIZE (1)                  75 01
  REPORT_COUNT (4)                 95 08
  OUTPUT (Data,Arg,Abs)            91 02
END_COLLECTION                     C0
END_COLLECTION                     C0
```

### ;PID Pool Move Report Definition

```
USAGE (PID Pool Move Report)      09 85
COLLECTION (Logical)              A1 02
REPORT_ID (12)                    85 0C
  USAGE (Move Source)              09 86
  USAGE (Move Destination)         09 87
  USAGE (Move Length)              09 88
  LOGICAL_MAXIMUM (32767)          26 FF 7F
  REPORT_SIZE (16)                 75 10
  REPORT_COUNT (3)                 95 03
  OUTPUT (Data,Var,Abs,Buf)        92 02 01
END_COLLECTION                     C0
```

### ;Device Gain Report Definition

```
USAGE (Device Gain Report)        09 7D
COLLECTION (Logical)              A1 02
REPORT_ID (2)                     85 02
  USAGE (Device Gain)              09 7E
  LOGICAL_MAXIMUM (255)            26 FF 00
  REPORT_SIZE (8)                  75 08
  REPORT_COUNT (1)                 95 01
  FEATURE (Data,Var,Abs)           B1 02
END_COLLECTION                     C0
```

END\_COLLECTION

C0

## 7. Example Report Summary

### 7.1 *Input Reports*

- 1 Data (standard joystick data)
- 2 State

### 7.2 *Output Reports*

- 1 Set Effect
- 2 Set Envelope
- 3 Set Condition
- 4 Set Periodic
- 5 Set Constant
- 6 Set Ramp Force
- 7 Set Custom Force Data
- 8 Download Force Sample
- 9 Set Custom Force
- 10 Effect Operation Report
- 11 Device Control
- 12 Pool Move

### 7.3 *Feature Reports*

- 1 Pool
- 2 Device Gain

### 7.4 *Other Reports*

- Custom Force Streaming Output



## 8. Example Reports

**Figure 3: Example PID Data Input Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 1							
1	Throttle							
2	X-axis							
3	Y-axis							
4	Button 4	Button 3	Button 2	Button 1	Hat Switch			

**Figure 4: Example PID Set Effect Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 1							
1	ROM Flag	Effect Block Index						
2	Effect Type							
3	Duration							
5	Sample Period							
6	Gain							
7	Trigger Button							
8	Trigger Repeat Interval							
9	Pad						Y Axis Enable	X Axis Enable
10	X Axis Direction							
11	Y Axis Direction							
12 - 13	Type Specific Block Offset 1							
14 - 15	Type Specific Block Offset 2							

**Figure 5: Example PID Set Envelope Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 2							
1	Parameter Block Offset (bits 7 – 0)							
2	ROM Flag	Parameter Block Offset (bits 14 – 8)						
3	Attack Level							
4	Attack Time							
5	Fade Level							
6	Fade Time							

**Figure 6: Example PID Set Condition Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 3							
1	Parameter Block Offset (bits 7 – 0)							
2	ROM Flag	Parameter Block Offset (bits 14 – 8)						
3	CP Offset							
4	Positive Coefficient							
5	Negative Coefficient							
6	Positive Saturation							
7	Negative Saturation							
8	Dead Band							

**Figure 7: Example PID Set Periodic Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 4							
1	Parameter Block Offset (bits 7 – 0)							
2	ROM Flag	Parameter Block Offset (bits 14 – 8)						
3	Magnitude							
4	Offset							
5	Phase							
6	Period							

**Figure 8: Example PID Set Constant Force Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 5							
1	Parameter Block Offset (bits 7 – 0)							
2	ROM Flag	Parameter Block Offset (bits 14 – 8)						
3	Magnitude							

**Figure 9: Example PID Set Ramp Force Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 6							
1	Parameter Block Offset (bits 7 – 0)							
2	ROM Flag	Parameter Block Offset (bits 14 – 8)						
3	Ramp Start							
4	Ramp End							

**Figure 10: Example PID Set Custom Force Data Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID =7							
1	Parameter Block Offset (bits 7 – 0)							
2	Byte Count (bit 0)	Parameter Block Offset (bits 14 – 8)						
3	Byte Count (bits 8-1)							
4-259	Custom Force Data (256 bytes)							

**Figure 11: Example PID Download Force Sample Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID =8							
1	X Axis							
2	Y Axis							

Note: This report is never actually transferred.

**Example PID Custom Force Streaming Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	X Axis							
1	Y Axis							

**Figure 12: Example PID Set Custom Force Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 9							
1-2	Parameter Block Offset							
3-4	Custom Force Data Offset							
5-6	Sample Count							

**Figure 13: Example PID Effect Operation Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 10							
1	ROM Flag	Effect Block Index						
2	Effect Operation							
3	Loop Count							

**Figure 14: Example PID Pool Feature Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 1							
1	RAM Pool Size							
3	ROM Pool Size							
5	ROM Effect Block Count							
7	Parameter Block Size (Set Effect Report)							
8	Parameter Block Size (Set Envelope Report)							
9	Parameter Block Size (Set Condition Report)							
10	Parameter Block Size (Set Periodic Report)							
11	Parameter Block Size (Set Constant Force Report)							
12	Parameter Block Size (Set Ramp force Report)							
13	Parameter Block Size (Set Custom Force Report)							
14	Custom Force Enable	Pad						

**Figure 15: Example PID State Input Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 2							
1	ROM Flag	Effect Block Index						
2	Pad				Actuator Power	Safety Switch	Enable Actuators	Effect Playing

**Figure 16: Example PID Device Control Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 11							
1	PID Device Control							

**Figure 17: Example PID Pool Move Output Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 12							
1	Move Source							
3	Move Destination							
5	Move Length							

**Figure 18: Example PID Device Gain Feature Report**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Report ID = 2							
1	Device Gain							

## 9. Example of Creating a Device Managed Effect

The report descriptor definitions for Create New Effect Report and PID Block Load Report are given below.

### Create New Effect Report

```
USAGE (Create New Effect Report)
COLLECTION (Logical)
  REPORT_ID (1)
  USAGE (Effect Type)
  COLLECTION (Logical)
    USAGE (ET Constant Force)
    USAGE (ET Ramp)
    USAGE (ET Square)
    USAGE (ET Sine)
    USAGE (ET Triangle)
    USAGE (ET Sawtooth Up)
    USAGE (ET Sawtooth Down)
    USAGE (ET Spring)
    USAGE (ET Damper)
    USAGE (ET Inertia)
    USAGE (ET Friction)
    USAGE (ET Custom Force Data)
  LOGICAL_MAXIMUM (12)
  LOGICAL_MINIMUM (1)
  REPORT_SIZE (8)
  REPORT_COUNT (1)
  FEATURE (Data, Ary, Abs)
END_COLLECTION
USAGE_PAGE (Generic Desktop)
USAGE (Byte Count)
LOGICAL_MINIMUM (0)
LOGICAL_MAXIMUM (255)
REPORT_SIZE (8)
REPORT_COUNT (1)
FEATURE (Data, Var, Abs)
END_COLLECTION
```

### PID Block Load Report

```
USAGE (PID Block Load Report)
COLLECTION (Logical)
  REPORT_ID (2)
  USAGE (Effect Block Index)
  LOGICAL_MAXIMUM (10)
  LOGICAL_MINIMUM (1)
  REPORT_SIZE (8)
  REPORT_COUNT (1)
  FEATURE (Data, Var, Abs)
  USAGE (Block Load Status)
  COLLECTION (Logical)
    USAGE (Block Load Success)
    USAGE (Block Load Full)
    USAGE (Block Load Error)
  LOGICAL_MAXIMUM (3)
```

```

LOGICAL_MINIMUM (1)
REPORT_SIZE (8)
REPORT_COUNT (1)
FEATURE (Data, Ary, Abs)
END_COLLECTION
USAGE (RAM Pool Available)
LOGICAL_MINIMUM (0)
LOGICAL_MAXIMUM (65535)
REPORT_SIZE (16)
REPORT_COUNT (1)
FEATURE (Data, Var, Abs)
END_COLLECTION
    
```

The procedure of creating a new Spring Effect.

1. Host sends a Set Report request to device. The format of the Set Report is shown in the below.

Setup:

<i>Offset</i>	<i>Field</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
0	bmRequestType	1	00100001b	From host to device
1	bRequest	1	0x09	Set_Report
2	wValue	2	0x0103	Report ID (1) and Report Type (feature)
4	wIndex	2	0x0000	Interface
6	wLength	2	0x0300	Number of bytes to transfer in the data phase (3 bytes)

Out:

	7	6	5	4	3	2	1	0
0	Report ID = 1 Create New Effect Report							
1	Effect Type							
2	Byte Count Note: This field is only valid when creating an ET Custom Force Data effect.							

2. Once the device receives this Set Report request and validates that it is capable of performing the requested operation, the device allocates the memory for the requested effect. The size of the allocated memory is decided by device.
3. Host then sends a Get Report request to device for the PID Block Load Report. The format of the Get Report is shown below:

<i>Offset</i>	<i>Field</i>	<i>Size</i>	<i>Value</i>	<i>Description</i>
0	bmRequestType	1	10100001b	From device to host
1	bRequest	1	0x01	Get_Report
2	wValue	2	0x0203	Report ID (2) and Report Type (feature)
4	wIndex	2	0x0000	Interface
6	wLength	2	0x0500	Number of bytes to transfer in the data phase (5 bytes)

4. Device responds the Get Report request with the PID Block Load report.



	7	6	5	4	3	2	1	0
0	Report ID = 2, PID Block Load Report							
1	Effect Block Index (1 ~ 255)							
2	Block Load Status 1: Block load success 2: Block load full (out of memory) 3: Block load error							
3~4	RAM Pool Available							

- If there is not enough memory in the device or the device can not create the effect for some reason, the device will return a 0 Effect Block Index and the appropriate Block Load Status to indicate an error in the PID Block Load Report request.
- If the device can successfully create the effect, then the host will send a Set Effect Report to the device to download the effect parameters for that effect with the ID previously received from device in the Effect Index field.. The Set Effect Report is an Output report sent to device.

	7	6	5	4	3	2	1	0
0	Report ID = Set Effect Report							
1	Effect Block Index (previously received from device in PID Block Load Report)							
...	.....							

- If the created effect was an ET Spring then the host would also send the two Condition Parameter Blocks to the device using Set Condition Reports.

0	Report ID = Set Condition Report							
1	Effect Block Index (previously received from device in PID Block Load Report)							
2	Parameter Block Offset <input checked="" type="checkbox"/> 0 for X axis, 1 for Y axis, Order of Axes Enable declaration							
..	..							

Note: For a device that employs a Device Managed memory model, the usage Parameter Block Offset field of the Set Condition Report is overloaded to indicate which axis the Set Condition report is targeted at. The order of axis declaration in the Axes Enable collection defines the index used for the Parameter Block Offset. In this example Report Descriptor two axes are declared, X and Y. The X axis is the first declared so a offset of 0 is applied, the Y axis gets an offset of 1.

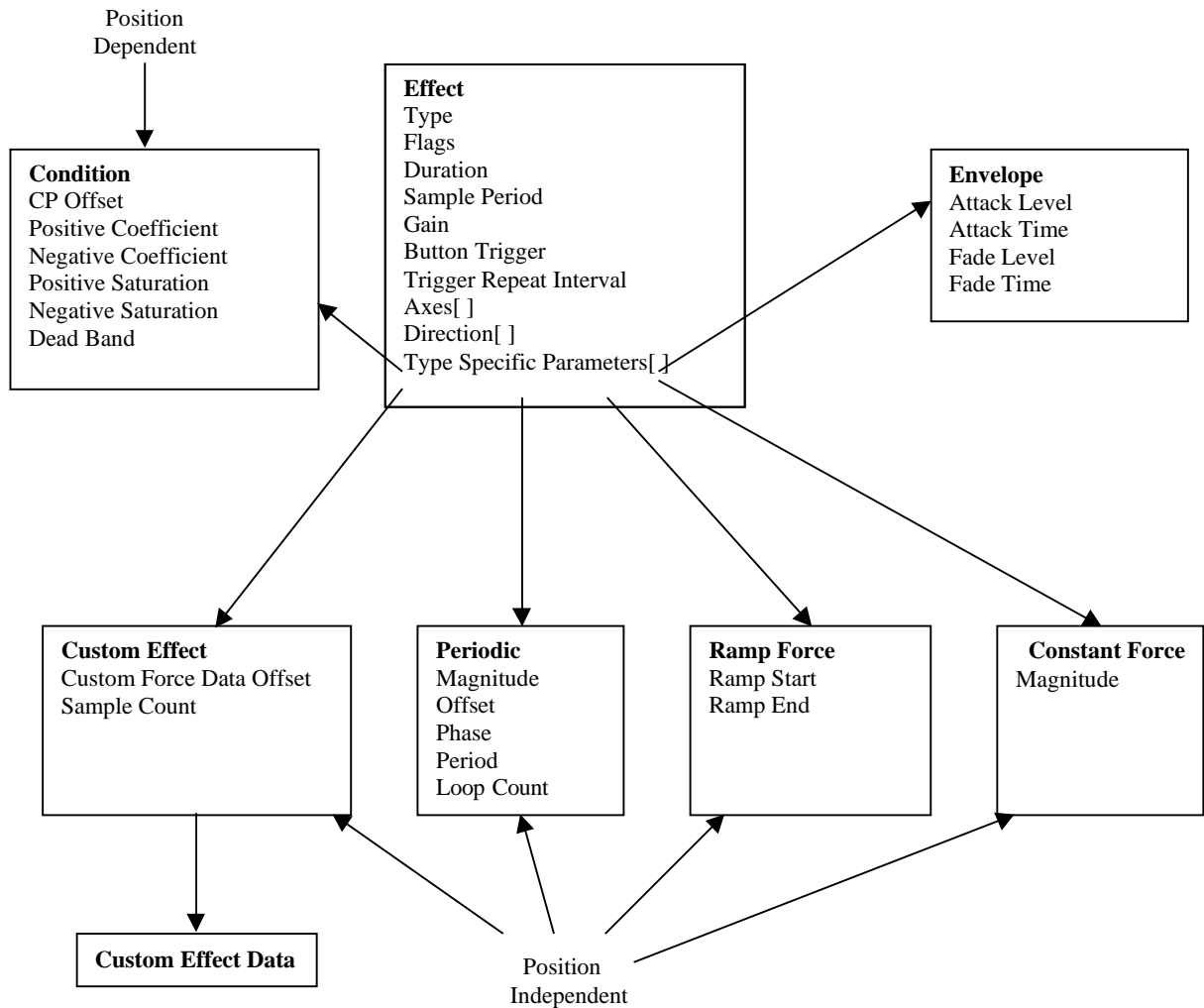
- Once the effect has been created and all the associated parameters have been successfully downloaded, the host now can start the effect.

## 10. Parameter Blocks

Depending on the Effect Type the Type Specific Parameters of the Effect Parameter Block will point to or provide a Handle for other Parameter Block types.

Position Dependence and Position Independence is related to the position of the actuator not the position in the Pool.

Figure 19: Parameter Blocks



## 11. Index

- Actuator Override Switch, 22
- Actuator Power, 22
- actuators, 4
- Actuators Enabled, 22
- Attack Level, 11
- Attack Time, 11
- Axes Enable**, 9
- Block Load Error, 22
- Block Load Full, 22
- Block Load Status**, 22
- Block Load Success, 22
- Block Type**, 11
- Condition, 12
- Condition parameter block, 12
- Constant Force, 15
- Constant Force parameter block, 15
- Contributing companies, ii
- coordinate scaling, 5
- CP Offset, 12
- Create New Effect Report**, 21, 41
- Custom, 13
- Custom Effect allocation, 19
- Custom Force, 14
- Custom Force Data, 14
- Custom Force Data Offset, 15
- Custom Force Data Report**, 14
- Custom Force parameter block, 14
- Custom Force Vendor Defined Data, 14
- Custom Forces, 13
- DC Continue, 23
- DC Disable Actuators, 23
- DC Enable Actuators, 23
- DC Pause, 23
- DC Reset, 23
- DC Stop All Effects, 23
- Dead Band, 13
- Device Control, 23
- Device Gain, 16
- Device Gain Report**, 16
- Device Managed Pool, 20
- Device Paused, 22
- Direction**, 9
- Direction Enable, 9
- Disclaimer, intellectual property, ii
- Download Force Sample**, 14
- Downloading of custom effect data, 14
- Duration, 9
- Effect Block Index, 8
- Effect Operation**, 16
- Effect Operation Report**, 16
- Effect Operations, 16
- Effect Type**, 8
- Envelope, 11
- ES Playing, 22
- ET Constant Force, 8
- ET Custom Force Data, 8
- ET Damper, 9
- ET Friction, 9
- ET Inertia, 9
- ET Ramp, 8
- ET Sawtooth Down, 8
- ET Sawtooth Up, 8
- ET Sine, 8
- ET Spring, 9
- ET Square, 8
- ET Triangle, 8
- Example, 24
- Fade Level, 11
- Fade Time, 12
- force, 4
- force vectors, 4
- Gain, 9
- INFINITE, 1
- Intellectual property disclaimer, ii
- interrupt OUT, 3
- License, software, ii
- Loop Count, 16
- Magnitude, 15
- Move Destination, 21
- Move Length, 21
- Move Source, 21
- Negative Coefficient, 12
- Negative Saturation, 13
- Normal, 8
- Normal force, 4
- Normalized, 1
- Normalized Usages, 5
- Offset, 15
- Op Effect Start, 16
- Op Effect Start Solo, 16
- Op Effect Stop, 16
- Parameter block management, 17
  - device managed, 21
  - driver managed, 17
- Parameter Block Offset, 8
- Parameter Block Size**, 20
- Period, 15
- Periodic, 15
- Phase, 15
- Physical Interface Device**, 8

PID (Physical Interface Device)  
  revision history, ii  
**PID Block Free Report**, 22  
**PID Block Load Report**, 21  
**PID Device Control**, 23  
**PID Device Control Report**, 22  
**PID Effect State**, 22  
**PID Pool Move Report**, 20  
PID Pool Report, 17, 20  
**PID State Report**, 22  
pool, 17  
Pool Alignment, 20  
pool allocation, 17  
Pool fragmentation, 20  
Positive Coefficient, 12  
Positive Saturation, 12  
RAM Pool Available, 22  
RAM Pool Size, 20  
Ramp End, 16  
Ramp Force, 15  
Ramp Force parameter block, 15  
Ramp Start, 16  
Report  
  Custom Force Data, 14  
  Device Gain, 16  
  Effect Operation, 16  
  PID Block Free, 22  
  PID Device Control, 22  
  PID Pool, 20  
  PID Pool Move, 20  
  PID State, 22  
  Set Condition, 12  
  Set Constant Force, 15  
  Set Custom Force, 14  
  Set Effect, 8  
  Set Envelope, 11  
  Set Periodic, 15  
  Set Ramp Force, 15  
ROM Effect Block Count, 20  
ROM Flag, 8  
ROM Pool Size, 20  
Safety Switch, 22  
Sample Count, 15  
Sample definition, 13  
Sample Period, 9  
Samples, 13  
scaling, 5  
**Set Condition Report**, 12  
**Set Constant Force Report**, 15  
**Set Custom Force Report**, 14  
**Set Effect Report**, 8  
**Set Envelope Report**, 11  
Set *PB* Report, 1  
**Set Periodic Report**, 15  
**Set Ramp Force Report**, 15  
Shared Parameter Blocks, 20  
Simultaneous Effects Max, 20  
Software license, ii  
Start Delay, 10  
Trigger Button, 9  
Trigger Repeat Interval, 9  
**Type Specific Block Handle**, 22  
Type Specific Block Handles, 10  
Type Specific Block Offset, 10, 11  
Units, 4  
Usage Page, 6  
vector, 13  
vectors, 4  
Vendor Defined Parameter Block, 16