ServoMaster: Design Contract

Table of contents

1. I'd like the ideal servo controller to...

Feature	FT639	Quad	Advanced	1-Wire	Pololu	Parallax
Report the hardware status.	See Hardware Status Reporting					+
Support capabilities discovery. Of course, on the hardware level the servo can't really tell much about itself, but the abstraction should support it.	+					
Implication: a database of servos in existence would not take much space, but will be very useful.						
Complication: the servo speed and torque changes as the voltage changes, so coming up with a uniform description for this						

dependency is going to be tricky.						
It should be possible to control the parameters for each servo separately (in other words, the servo abstraction should be configurable).		+	+	+	+	+
It should be possible to preset the minimum and maximum angle for the servo. And not only 90° vs. 180°, but arbitrary angle.	+					
Implied: the pulse length range. The limits may be defined by the servo construction and/or mechanical layout of the apparatus in which the servo is used. This is going to get really		+	+	?	+	+

technical, but the ability to specify the minimum and maximum pulse length just has to be present.						
Implied: range should be included into capabilities discovery. Implied: the capabilities for each particular servo are different.		+	+	?	+	+
Even though the actual range may be different from 90° or 180°, the number of steps to cover the actual range should be the same.			+	?	+	+
Exactly as the capabilities of each physical instance of the servo are different,						

the capabilities of each physical instance of the controller may be different. To allow proper operation, there should be a mechanism to override default (advertised) capabilities of the servo and the controller. This can be achieved by providing "configurabilit	
An entity able to instantiate the controller and configure it ("controller factory") will be useful. Implied: generalized servo controller and servo definition format. Begs for XML configuration file.	

The servo interface shouldn't really accept the values from 0 to 255 for the servo position, but rather 0.0 to 1.0 (extreme left to extreme right). The reason is, different controllers have different value ranges - some operate from 0 to 255, some from 0 to 254, there is at least one that accepts positions from 0 to 4000, some use the high bit as the direction indicator and the rest as position. Implied: precision is different. Implied: controller precision should be included	+

into capabilities discovery.	
Provide the coordinate transformation between the physical servo position and input value. Useful for compensating the rotational servo movement to linear controlled body movement, as well as applications where the natural input value range differs from 0...1 (for example, steering wheels and rudders). This abstraction shouldn't be really a part of the servo abstraction, but rather a module you can plug the servo into.	+
Inverting the coordinate polarity to reverse the	+

servo is not exactly a life saver, but can be useful and is really simple. It can be a part of the coordinate transformatio module as well.	
Provide the means to incorporate the closed loop feedback signal and mapping between the feedback and control.	
The ideal servo controller doesn't necessarily have to be integrated - there are cases when the servos are located in remote locations. The surface mount design can possibly fit on the back of the servo.	
Implication: it's not that	+

the abstraction of the **servo controller** that is important, it is the abstraction of the **servo**.						
However, the abstraction of the servo controller is important as well: even though the actual servo drivers may be physically and logically separated, they are accessed in the same uniform manner, and share common properties.	+					
From the practical standpoint: how do I care if the controller is one piece or distributed, as long as the cost is comparable?				+		
The ideal controller		+	+	+	+/-	+/-

should not be resource hungry. Serial and parallel devices are evil, they want a dedicated port to control them, why? Good candidates: USB, 1-Wire.						
Hardware controlled transition is a desirable feature.			+		+	+
Controlled transition: the transition pattern can be specified, for example, just a simple linear transition in a given time, or a constant and/or limited acceleration transition (good with moving heavy controlled bodies), or approximatior	See Transitions & Transformations					

transition (good when the exactness of approach to the specified position is required), or any sort of special effect ("drunken master").	
Transition controller isn't really stackable, so it is enough to "attach" one to the servo abstraction.	+
Transition controller may interfere with the coordination transformers, so there should be a possibility to attach it at any level of the coordination transformer stack.	+

2. Hardware Status Reporting

At the moment of writing, the only controllers that support hardware status reporting are the Parallax serial and USB controllers. However, there are some additional limited special cases

- they are discussed below.

2.1. Removable Devices

One case where a hardware device departure (that may be considered a failure) and/or arrival is really easy to determine is when the device communicates with the computer using a hotpluggable protocol such as USB or IEEE 1394, a.k.a. FireWire, or 1-Wire®.

Consequently, all the devices that are using one of these protocols are supported as removable. ServoMaster API users will get arrival and departure notifications should they choose to listen to them.

2.2. Disconnected Mode

In some cases services provided by ServoMaster API are required to run mission-critical, long-lived applications. Since disconnect related hardware failures may be transient, it makes perfect sense not to fail the API object completely, but switch to *disconnected mode*, and return back to normal mode as soon as connectivity is restored.

It is possible to programmatically control whether the device is allowed to be disconnected.