

Mejora de procesos para el desarrollo de dispositivos protésicos de mano

Process Improvement for develop hand prosthetic devices

J. S. Cabrera y H. F. Jaramillo

Resumen

Entre los dispositivos protésicos al que mayor atención se le ha prestado es la prótesis de mano, en la que las arquitecturas de diferentes soluciones poseen una base común; sin embargo, los avances de cada una son poco portables debido al alto acoplamiento que suelen presentar sus módulos.

La distribución de labores en equipos de trabajo y la portabilidad de las soluciones puede facilitarse a través del uso de interfaces normalizadas para la interacción de los módulos, de manera que su implementación queda encapsulada y la entrega y recepción de señales es conocida *a-priori* por todos los integrantes del equipo de desarrollo.

El presente trabajo pretende difundir un conjunto de interfaces normalizadas para los diferentes módulos de un dispositivo protésico de mano, de forma que cada módulo se constituya en un componente reutilizable, independientemente de su implementación, y contribuya al mayor avance de los diferentes desarrollos en el área, a través de una integración sencilla, efectiva y bien documentada.

Palabras claves Reuso, Encapsulamiento, Interfaz normalizada, Prótesis de Mano.

Abstract

Among the prosthetic devices that more attention has been paid is the prosthetic hand, and most of the solutions have a common base architecture, but their modules are non-portable because of its high interdependencies (coupling).

The work distribution in teams, and portability of their solutions, can be easily made through the use of standardized interfaces for the interaction of the modules, so that its implementation is encapsulated, and the delivery and reception of signals is known *a-priori* by all members in development team.

This paper aims to let know a set of standardized interfaces for any module of a prosthetic hand device, so that each module becomes a reusable component, regardless of their implementation, and contribute to the further advancement of the different developments in the area through easy, effective and well documented integration.

Key words Reuse, Encapsulation, Normalized interface, Hand prosthetic device..

1 Introducción

Con la presente propuesta se busca entonces contribuir a la solución del siguiente problema:

¿Cómo integrar los componentes de un dispositivo protésico de manos, de tal forma que esa integración sea una tarea efectiva y debidamente documentada, con el fin de que soporte un desarrollo sostenible y permita su evolución?

Con el fin de dar respuesta a esta pregunta se genera la siguiente hipótesis: aplicar los conceptos enmarcados en la ingeniería de software a arquitecturas de sistemas electrónicos, especialmente en la definición de interfaces para los componentes que integran un sistema.

Un conjunto normalizado de interfaces para los componentes de un dispositivo protésico de mano permitiría un desarrollo sostenible y comprensible del mismo, de tal forma que si se hace necesario cambiar total o parcialmente la estrategia de control en alguno de los componentes se realice teniendo en cuenta las interfaces previamente establecidas, sin causar inconvenientes al funcionamiento del resto de componentes.

Dicho conjunto de interfaces facilitaría la integración y depuración de los componentes hardware/software que conforman el simulador de prótesis de mano robótica y permitiría llevar de una manera metódica un sistema de control de versiones, el cual registre los cambios realizados con miras a la puesta a punto, mantenimiento y evolución del sistema.

En [1] se establece que el marco de trabajo inicial para estructurar un sistema electrónico compuesto por hardware/software se conoce como diagrama arquitectónico de bloques. Los sistemas entre más complejos se descomponen en unidades que suministran algún conjunto relacionado de servicios, las nuevas tecnologías y la mayoría de las arquitecturas modernas requieren del uso de componentes para estructurar y distribuir la aplicación, así, una parte esencial del proceso de especificación es definir las interfaces de dichos componentes; una vez que se acuerden y definen las interfaces, los componentes se pueden desarrollar de forma independiente [2].

2 Construcción de la base de conocimiento.

Se tomará como referencia una adaptación del Modelo para la Investigación Documental propuesto en [3]. Específicamente el modelo propone la realización de las siguientes fases:

- Fase Preparatoria.
- Fase Descriptiva.
- Fase de Interpretación por Núcleos Temáticos.
- Fase de Construcción Teórica Global.

En la Fase Preparatoria se definió el alcance de la base de conocimiento con el fin de reunir de forma sistemática el conocimiento existente y disponible actualmente con el fin de generar el conjunto normalizado de interfaces para los componentes de un dispositivo protésico de mano.

Para la construcción de la Base Inicial de Conocimiento se tuvo como referencia el siguiente conjunto inicial de preguntas:

- ¿Cuáles son las arquitecturas más relevantes el desarrollo prótesis de mano?
- ¿Cómo se realiza la integración entre los componentes de dichas arquitecturas?
- ¿Cuáles son las herramientas tecnológicas más relevantes utilizadas en el desarrollo de dispositivos protésicos de mano?

A partir de estas preguntas se seleccionaron una serie de unidades de análisis que aportan elementos para dar respuesta al problema expuesto, las cuales se agruparon en los siguientes cinco núcleos temáticos:

- Arquitecturas para dispositivos protéticos de mano.
- Adquisición, caracterización y reconocimiento de intención de movimiento, a partir de las señales mioeléctricas.
- Sistema de control para prótesis de mano robótica.
- Realimentación del estado del dispositivo protético al usuario.
- Propuestas para la definición de interfaces de componentes hardware/software.

En la Fase Descriptiva, por cada una de las unidades de análisis se elaboró una ficha descriptora, las cuales se pueden consultar en [4]; sin embargo, debido a la gran cantidad de información que esto implica, para el propósito de este artículo se decide presentar la Fase de Interpretación por Núcleos Temáticos, la cual se describe a continuación. Por último, a partir de la Fase de Construcción Teórica Global se conforma el conjunto normalizado de interfaces el cual se expondrá más adelante, específicamente en el Mejora para el proceso de desarrollo de prótesis de mano.

2.1 Arquitecturas para dispositivos protéticos de mano

En aplicaciones como [5], [6], [7], [8], [9] y [10], como diseño del sistema se elabora un diagrama arquitectónico de bloques. De igual manera, para la generación del conjunto normalizado de interfaces se hace necesaria la construcción de un diagrama de este tipo, el cual es una aproximación inicial para el diseño arquitectónico del sistema.

Particularmente en [11] se propone un diagrama en bloques estándar denominado: Esquema de control formal multifuncional de una mano artificial (ver Figura 2.1). A continuación se describe abreviadamente los módulos que conforman el diagrama propuesto.

2.1.1 TCM - Módulo de control de alto nivel (Top-Level Controlling Module)

Este módulo adquiere la señal de entrada desde el usuario y genera los comandos de control o intenciones de movimiento que serán las señales de entrada para el módulo de control de bajo nivel.

2.1.2 LCM - Módulo de control de bajo nivel (Low-Level Controlling Module)

Este módulo recibe como entradas la señal generada por el TCM y las señales provenientes del Módulo de Procesamiento Sensorial (SPM), para controlar los movimientos de la mano y la fuerza de agarre.

2.1.3 SPM - Módulo de Procesamiento Sensorial (Sensory Processing Module)

Este módulo adquiere la señal proveniente de los sensores localizados en la mano; convierte y entrega la información al módulo LCM cerrando el ciclo de control de la prótesis de mano.

2.1.4 SFM - Módulo de Retroalimentación Sensorial (Sensory Feedback Module)

Este módulo adquiere la información generada por el SPM y la presenta al usuario utilizando una codificación apropiada.

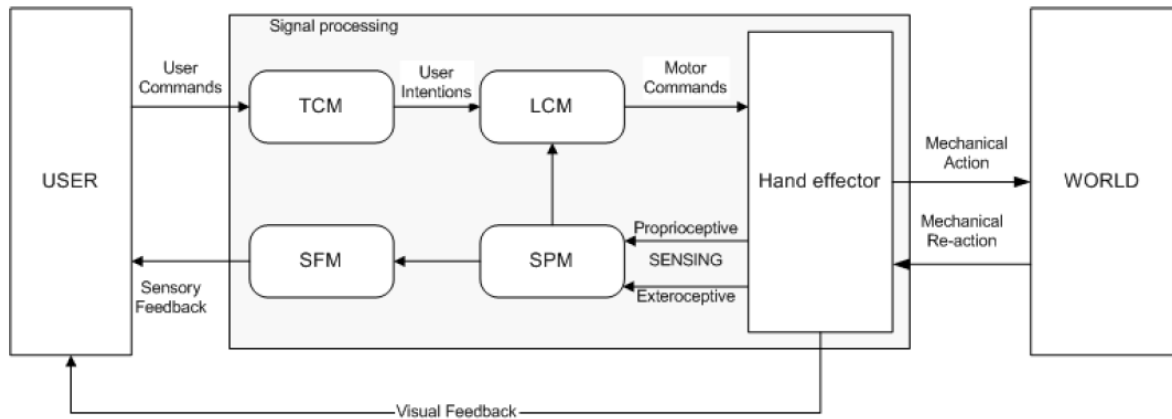


Figura 2.1. Arquitectura de control multifuncional de una mano artificial, Ph.D. Zecca.

2.2 Adquisición, caracterización y reconocimiento de intención de movimiento a partir de las señales mioeléctricas.

2.2.1 Adquisición de señal ME

La adquisición y procesamiento de la señal mioeléctrica hace parte del módulo de control de alto nivel. Otros autores han denominado a este módulo: cabecera analógica y sistema de adquisición de datos [5], captación de la señal, rectificación y filtrado [6], acondicionamiento y procesamiento [12], módulo de adquisición y pre-procesamiento [10], entre otros. Al igual que [11] y [13], los anteriores autores coinciden en que las señales Electromiográficas Superficiales o EMGS son una fuente de información muy apropiada para el control de dispositivos prostéticos.

La electromiografía según [11] es la disciplina que trata la detección, el análisis y el uso de señales eléctricas que emiten las contracciones musculares. Esta señal es referida como señal electromiográfica o señal EMG, el cual es un término más apropiado en el pasado que en el presente. En el pasado la única manera de capturar la señal para un estudio subsecuente era obteniendo una representación gráfica. Hoy en día es posible almacenar la señal en una cinta magnética, en un disco compacto o en dispositivos electrónicos. Un número creciente de profesionales en el área ha adoptado el término de señal mioeléctrica o señal ME, aunque el término EMG es aún muy usado sobre todo en el ámbito clínico.

El problema paradójico con las señales ME es que son unas de las señales electro-fisiológicas más fáciles de medir, pero también unas de las más difíciles de interpretar. Incluso con un osciloscopio y dos objetos metálicos es posible registrar señales ME. Según [14] "Para su detrimento, la electromiografía es tan fácil de usar y consecuentemente tan fácil de abusar".

La electromiografía puede clasificarse en dos tipos según los electrodos: EMG de aguja, en Inglés Needle EMG y de superficie o EMGS, los cuales son el principal transductor para la conformación del sensor

En [12] el término sensor en lugar de electrodo se utiliza para subrayar el hecho de que cada señal ME necesita de la medición de un conjunto de al menos dos electrodos que a menudo se integra junto al pre-amplificador en el mismo encapsulado. Según [15] un electrodo es la superficie metálica del sensor, la cual realiza el contacto eléctrico con la piel también conocido como sensor pasivo, por ejemplo, los electrodos desechables fabricados por [16]. Por otro lado, un sensor activo es la unidad que contiene los electrodos y la circuitería electrónica para el amplificador diferencial de las señales ME y reduce la contribución de varias fuentes de ruido. A este tipo de sensores también se los conoce como sensores activos.

A través del proyecto SENIAM [17], el cual tiene como objetivo integrar las investigaciones básicas y aplicadas sobre señales electromiográficas de superficie, se ha establecido un acuerdo sobre la configuración de la superficie de detección, las dimensiones de un electrodo, entre otras. Entre las recomendaciones se encuentran:

- Disposición de los electrodos en configuración diferenciales.
- Las superficies de detección consisten de dos barras paralelas, cada una de 1.0 cm de largo y entre 1 y 2 mm de ancho, con 1.0 cm de separación.
- El ancho de banda de 20 - 500 Hz con un roll-off de al menos 12dB por octava.
- Tasa de rechazo de modo común CMRR >80dB.
- Ruido < 2 μ Voltios en la banda de 20 - 400Hz.
- Impedancia de entrada >100 Mega Ohms.
- Localizar los electrodos en el centro del vientre muscular, entre la unión miotendinosa y la zona de la inervación más cercana, con las superficies de detección orientadas paralela a la longitud de las fibras musculares.
- Utilizar el valor RMS de la señal para medir la amplitud de la señal EMG provocadas voluntariamente.

La introducción del procesamiento digital en el análisis de señales ME sugiere su digitalización mediante un proceso de muestreo. De acuerdo con [12], alrededor del 95% de la potencia proveniente de armónicos se localiza por encima de los 400Hz, y la mayoría del porcentaje restante es generado por los electrodos y ruido proveniente de equipos electrónicos cercanos a la muestra.

La frecuencia de corte es usualmente escogida cercana a los 500Hz; por lo tanto, de acuerdo con el teorema de muestreo, el cual sugiere que la frecuencia debe ser al menos dos veces más grande que la frecuencia más alta de la señal, la frecuencia de muestreo debe ser al menos 1000 muestras/segundo. En [13] para el análisis de las señales se recomienda segmentar los registros en segmentos de longitud de 256 muestras, es decir, 256 ms de grabación y aplicar las técnicas de extracción de características a cada segmento.

Los requerimientos de digitalización de la señal ME dependen de tamaño de las amplitudes esperadas. La señal ME capturada a través de electrodos de plata/cloruro de plata varía entre 0 y 6mV [7]. Por lo tanto la detección de detalles en bajas amplitudes está limitada por el nivel de ruido del sistema amplificador. Los amplificadores modernos tienen un nivel de ruido del orden de unos pocos milivoltios, por lo tanto una digitalización al rededor de $0.5mV/bit$ es suficientemente precisa [12]. Un convertor de 16 bits tiene $2^{16} = 65536$ posibles niveles; esto significa que sin una etapa de adaptación de ganancia, el rango de medida con un convertor de este tipo es +16mV, el cual es suficiente para la mayoría de aplicaciones.

2.2.2 Caracterización de la señal ME

A partir de la digitalización de la señal ME se viabiliza el uso del procesamiento digital de señales con el fin de filtrar y extraer información de tal forma que se logre reconocer patrones e identificar la intención de movimiento que va a ejecutar el paciente. Este proceso se reconoce en investigaciones como [6], [7], [9], [13] y [18].

No es posible ir al detalle respecto al procesamiento de la señal ME, dado que las posibilidades son virtualmente ilimitadas. Sin embargo, de acuerdo con [19], a continuación se nombran las características más relevantes tanto en el dominio temporal como de la frecuencia.

Las características basadas en estadísticos temporales tienen como particularidad que se calculan rápidamente en cada segmento de tiempo. Entre las características se encuentran:

- Rectificación.
- Promedio o media de la señal rectificada.
- Integración.
- Valor raíz cuadrática media o *rms*.
- Valor medio absoluto de la señal.
- Valor medio absoluto de pendientes.
- Longitud de la forma de onda.
- Coeficientes modelo paramétrico auto regresivo, AR.

Por otro lado, la representación tiempo-frecuencia de la señal provee información de la distribución de su energía en los dos dominios lográndose una descripción más completa del fenómeno físico [13]. Sin embargo, una transformación de la señal implica un costo computacional adicional y un espacio de características de mayor dimensión. Entre las características están:

- Mediana, Media y Moda del espectro de frecuencia.
- Ancho de banda del espectro de potencia.
- Coeficientes Transformada Wavelet y Transformada de Paquetes Wavelet.
- Coeficientes Ceptrales.

Para el análisis del espectro de frecuencia es particularmente útil la Transformada de Fourier y sus derivaciones como la Transformada de Fourier de Tiempo Corto (STFT) [20] y la Transformada de Gabor [21].

Una forma simple de medir la frecuencia de la señal es a través del valor de cruces por cero y el número de inflexiones o cambios de signo. Para ver una descripción detallada de cada característica, referirse a [13], [14] y [22].

De la caracterización de la señales ME, la idea es conformar un vector descriptor utilizando las características anteriormente mencionadas y así reconocer un patrón por cada intención de movimiento.

A partir del vector descriptor de la señal ME, el cual puede estar compuesto por todas o algunas de las características previamente mencionadas, es de interés eliminar información repetitiva y que no aporta para el reconocimiento de la intención de movimiento. En [13] a este proceso se le conoce como reducción de dimensionalidad del vector descriptor. Entre las técnicas más relevantes para este fin se encuentran:

- El análisis de componente principal o Transformada de Karhuemn Loeve.
- Descomposición de valores singulares.
- Algoritmos Genéticos.

2.2.3 Reconocimiento de intención de movimiento

El sistema de reconocimiento de patrones tiene como finalidad reconocer las intenciones de movimiento del paciente. Para una prótesis de mano los tipos de agarres serían los movimientos por ejecutar. Sin embargo, de acuerdo con [23] y [24], el agarre no es un movimiento único: existe un proceso el cual se divide en tres fases. La fase previa al agarre, la cual prepara a la mano de acuerdo con el objeto que va a sujetar; la fase del agarre, la cual empieza en el momento que la mano toca y sostiene el objeto de una manera estable; y por último, la fase de manipulación, la

cual se caracteriza por movimientos que resultan como consecuencia de mover el objeto con relación al ambiente.

En [23] se presentan algunas de las clasificaciones para los agarres de una mano humana; entre ellas están: Schlesinger, Napier y Cutovsky. Sin embargo, la clasificación de Schelsinger se propuso desde la perspectiva del diseño de una mano protésica basado en los parámetros de la forma del objeto, superficie de la mano y forma de la mano; particularmente, los tipos de agarre propuestos son: cilíndrico, pinza, gancho, palmar, esférico y lateral, los cuales se presentan en la Figura 2.2. Las clasificaciones de Napier y Cutovsky se concentran en diferenciar los tipos de agarre de acuerdo con la precisión y la fuerza, y no con la intención de movimiento.

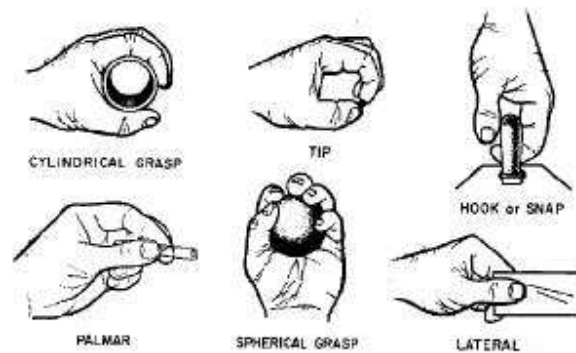


Figura 2.2 Clasificación de agarres de mano

En el proceso de reconocimiento de intención de movimiento se usan los resultados de la extracción de características como un conjunto de entrenamiento ya sea para una red neuronal o como una base de identificación estadística para el reconocimiento de patrones. Es decir, en este proceso se pretende identificar la huella digital de cada uno de los movimientos generados por el paciente, a partir del conjunto de características temporales o en frecuencia. En [22] se considera la técnica de reconocimiento de patrones como una disciplina de la inteligencia artificial.

Existen una variedad de herramientas para el reconocimiento o clasificación de patrones de las cuales se pueden destacar:

- Clasificador Bayesiano.
- Análisis Discriminante Lineal.
- Redes Neuronales Artificiales.
- Lógica Difusa.
- Sistemas Neuro-Difusos.

El clasificador Bayesiano se reconoce en estudios como [25] [26] y [27], el cual se basa en el teorema de Bayes, utiliza el concepto de probabilidad *a-posteriori*, probabilidad *a-priori* y probabilidad condicional.

En [28] se presenta el Análisis Discriminante Lineal como un método estadístico que discrimina la separabilidad de clases en forma lineal, lo cual no se ajusta de manera eficiente a los patrones EMG por su dinámica estocástica. Sin embargo, ha sido utilizado por Englehart y Hudgins en aplicaciones con señales ME [18].

Las redes neuronales artificiales junto a la lógica difusa se encuentran entre las herramientas más utilizadas. Su aplicación puede reconocerse en [5], [29] y [30].

2.3 Sistema de control para prótesis de mano robótica

En [8] se afirma que la principal limitación en el uso de una prótesis artificial es el control del sistema. Dicho control debe ser intuitivo y que implique un escaso esfuerzo mental con el fin de hacer de la prótesis una herramienta práctica y que pueda ser usada por largos periodos.

Dado el alto grado de complejidad de la mano humana, tanto ingenieros como investigadores han simplificado el diseño del dispositivo prostético. Como se evidencia en [5], [31], [32], [33], [34], [35], el mecanismo de la prótesis consiste en una palma estática central o metacarpo de la que surgen entre dos y cinco dedos. Dicha palma está unida al antebrazo por una unión llamada muñeca o carpo. Aunque el pulgar no tiene falange media y por eso no es correcto considerarlo como dedo, siempre se encuentra presente en el diseño de los dispositivos prostéticos.

A pesar que existen muchas sensaciones que se han perdido en un proceso de amputación, de acuerdo con [36] y [37] las sensaciones de contacto, fuerza y deslizamiento son las más útiles para el agarre. De acuerdo con [38], la finalidad del módulo de control radica en realizar dos acciones: ubicar los dedos en las posiciones deseadas según el tipo de agarre (consignas de posición) y posteriormente entrar en contacto con el objeto controlando la fuerza (consignas de fuerza).

Para construir un sistema de control realimentado de fuerza o posición se utilizan ya sea las herramientas del control clásico o moderno como del control inteligente. Sin embargo, no es relevante el tipo de herramienta utilizada para definir las interfaces del módulo de control. Es decir, sin importar la estrategia de control las variables de entrada y salida del módulo deben ser las mismas.

Nótese que tanto el control de posición como el de fuerza generan una orden de par a cada uno de los actuadores de las articulaciones para garantizar el seguimiento de las consignas. La acción simultánea de los dos controles puede generar conflicto dado que dos tipos diferentes de acción se están ordenando a los mismos actuadores. Por lo tanto, es necesario crear una estrategia de sincronización con el fin de que las acciones correctivas generadas por uno de los de los controles no interfieran con las señales del otro.

Por otro lado, es de igual importancia la disposición de sensores en el dispositivo prostético. En aplicaciones de agarre, tales como las realizadas por [39], [40] y [41] se conforma una matriz de sensores táctiles que cubren la superficie de la mano robótica. En [36] se determinan algunos puntos para la ubicación de sensores de fuerza y deslizamiento sobre la mano PROTECI basados en las observaciones del agarre de la mano humana.

En la Figura 2.3 se presenta un esquema habitual de un sistema de control realimentado, en donde el dispositivo prostético se encuentra representado por la planta o proceso, el sistema de realimentación sensorial por el módulo de medición y, por último, la estrategia de control representada por el módulo controlador.

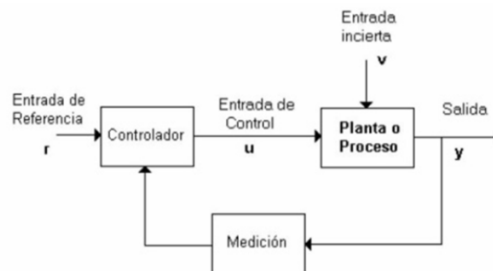


Figura 2.3 Esquema básico de control realimentado

2.4 Realimentación del estado del dispositivo protésico al usuario.

El principal interés de este módulo es proporcionarle al usuario algunas sensaciones con el fin de conocer el estado del dispositivo protésico, especialmente en la percepción del agarre y la generación de señales ME.

Aunque este módulo sea considerado de baja acogida entre los pacientes debido a que las técnicas hasta el momento utilizadas no se asemejen a una sensación natural, es de gran importancia no perderlo de vista para futuros avances en el área.

Las sensaciones se transmiten al usuario a través de señales visuales, estímulos táctiles, sonidos, temperatura y especialmente a través de la Estimulación Eléctrica Funcional (EEF), la cual es aplicada ya sea en el músculo o directamente en los hilos nerviosos presentes en el miembro amputado.

Como se muestra en [42], la EEF es utilizada como una herramienta clínica para rehabilitación en pacientes con parálisis en sus extremidades. Sin embargo, en trabajos como [10] y [43] se la utiliza para realimentar al paciente con la información sensorial implantada en el dispositivo protésico.

2.5 Propuestas para la definición de interfaces de componentes hardware/software.

Actualmente, los sistemas electrónicos se conforman por componentes hardware y/o software tales como los sistemas basados en procesadores digitales, en los cuales se ejecuta un conjunto de instrucciones. Dado que se pretende definir un conjunto de interfaces para los componentes hardware y software de un dispositivo protésico de mano, a continuación se expone la forma como regularmente se especifica una interfaz para dichos componentes.

2.5.1 Interfaz Software

De acuerdo con [44] un módulo o componente software es un archivo de código fuente que contiene funciones y las variables para soportar dichas funciones. En [45], se especifica que la interfaz de un componente software es la parte de un programa que sirve para comunicarse con otro programa, con una parte del hardware o con el usuario. En un sentido más amplio, puede verse la interfaz como el conjunto de funciones que permiten la intercomunicación del programa con cualquier otro programa o entre partes del propio programa, o incluso con componentes hardware. A continuación se especifica la interfaz de una función software, en donde se especifica las variables o argumentos de entrada y salida de la función.

TipoRetorno **nombreFuncion**(*tipo* Argumento1, *tipo* Argumento2, ... *tipo* ArgumentoN)

En el *Firmware Development Standard* (Estándar de Desarrollo Firmware) propuesto en [46], para la documentación de las interfaces software se sugiere los siguientes aspectos: el nombre de la función, los valores de entrada y salida del componente, y una descripción detallada de su funcionalidad, restricciones y modos de fallo. Del mismo modo sugiere que dicha información debe registrarse en el mismo código del programa a manera de comentario y previamente a la definición del prototipo de la función.

Por otro lado, para la definición de interfaces existen diferentes lenguajes tales como: MIDL, del Inglés Lenguaje de Definición de Interfaces de Microsoft, *Open Service Interface Definition*, WSDL, en español lenguaje de descripción de servicios web, *Platform Independent Component Moledling Language*; el más sobresaliente de ellos y del cual se han derivado en su mayoría es el Lenguaje de Descripción de Interfaces o IDL por su nombre en Inglés.

El IDL es un lenguaje desarrollado por el *Object Management Group*, **OMG** de sus siglas en inglés con el fin de contar con un mecanismo de abstracción para la implementación de sistemas distribuidos. La principal función es la de separar la implementación de la definición a través de un contrato entre el cliente y el servidor en el que se describen los datos y las interfaces de los objetos de una aplicación [47].

2.5.2 Interfaz Hardware

De acuerdo con [45], una interfaz hardware es el punto de contacto entre dos sistemas o módulos de un sistema. Es una conexión física y funcional entre dos aparatos o sistemas independientes. Particularmente en el campo de la electrónica y las telecomunicaciones, una interfaz es el puerto o circuito físico a través del que se envían o reciben señales desde un sistema o subsistema hacia otros. No existe una interfaz universal, existen diferentes estándares tan complejas como: Interfaz USB o interfaz SCSI o mucho más sencillas como las interfaces que definen los pines de entradas y salidas de un dispositivo electrónico.

Así también, una interfaz puede ser definida como un intérprete de condiciones externas al sistema, por ejemplo a través de transductores y otros dispositivos, permitiendo una comunicación con actores externos, como personas u otros sistemas. Para el caso de una prótesis mioeléctrica, caso de estudio del presente artículo, la interfaz entre el hombre y la mano protésica son simplemente los sensores EMG.

No existe una manera estandarizada de definir una interfaz hardware. Sin embargo, entre los lenguajes más difundidos para la descripción de componentes hardware se encuentran *VHDL* y *Verilog*. Dichos lenguajes definen la interfaz de un componente hardware a través del concepto de *entity* en el caso de *VHDL* y *module* en *Verilog*. Dicha entidad o módulo se asemeja a la definición de la interfaz o prototipo de una función de software, la cual se concibe como una caja negra en la que solamente se especifican las entradas y salidas y no se describe su funcionamiento [48].

Frecuentemente, desarrolladores de hardware se encuentran con documentos que describen un componente hardware a través de su interfaz; este tipo de documentos se conoce como Hoja Técnica (Datasheet), el cual presenta una descripción detallada del comportamiento de un dispositivo hardware, sus limitaciones, modos de fallo, condiciones eléctricas entre otra información.

3 Mejora para el proceso de desarrollo de prótesis de mano.

De acuerdo con lo expuesto anteriormente, tanto una interfaz hardware como software tienen el mismo objetivo: definir las responsabilidades de un componente sin especificar la manera en que dichas responsabilidades se llevan a cabo.

Al definir un conjunto de interfaces se realiza un aporte a la especificación de la arquitectura del sistema más que a la manera de realizar su implementación. Por ejemplo, en el inicio de la etapa de diseño de un sistema hardware/software, no es relevante la estrategia a seguir para cumplir con determinada funcionalidad o la herramienta con que se realizará dicha implementación. Lo que realmente importa es que se defina esa función y se especifique el tipo de información que es requerida como entrada al componente y la información que éste entrega luego de su ejecución. Sin embargo, es de gran interés para los desarrolladores de sistemas hardware/software que además de la información de entrada y salida del componente se especifique el tiempo de respuesta de la función, las condiciones eléctricas para las señales hardware y el protocolo de comunicación.

Teniendo en cuenta lo anterior y considerando la necesidad de definir un conjunto de interfaces hardware/software documentadas e independientes de la plataforma de

implementación se decide utilizar el lenguaje de definición de interfaces, IDL, para especificar tanto interfaces hardware como software y realizar su documentación por medio del *Firmware Development Standard* propuesto en [46].

Dado que se pretende unificar la definición de las interfaces hardware y software, en el presente proyecto se propone definir un nuevo tipo de dato el cual describa de una mejor manera a una señal analógica y a su vez permita modelar una interfaz de un componente hardware. El tipo de dato propuesto se denomina *Analog*.

Un tipo de dato *Analog* corresponde a todo tipo de señal eléctrica que varía de manera continua e infinitesimalmente, ya sea en voltaje o en corriente. Por ejemplo, una onda de voltaje senoidal continua es una señal analógica y podría ser considerada como un dato de tipo *Analog*.

Respecto a definir el tiempo de respuesta de una función, su importancia radica en que muchos de los sistemas hardware/software deben responder en tiempo real, es decir, responder antes de un tiempo especificado. Por ejemplo, si en la interfaz del componente *Caracterizador* se define la responsabilidad de *caracterizar señal ME*, la cual podría tener diversas estrategias que cumplan con la misma funcionalidad, algunas podrían ser muy complejas demanden mucho tiempo de ejecución, lo cual haría que se exceda el tiempo máximo de ejecución y que todo sistema se ralentice generando una respuesta fuera de lo esperado.

A partir de las anteriores consideraciones se propone la Tabla 3.1 con el fin de documentar las responsabilidades, funciones u operaciones que conforman las interfaces hardware/software.

Nombre Módulo:	Nombre con que se define la operación.
Retornos:	Descripción del valor de retorno o salida de la operación.
Argumento 1:	Descripción del primer argumento o señal.
Argumento 2:	Descripción del segundo argumento o señal.
Argumento n:	Descripción del n-ésimo argumento o señal.
Tiempo Respuesta:	Tiempo máximo de respuesta de la operación
Creador:	Nombre del autor, fabricante, referencia comercial.
Fecha de creación:	Fecha en la que se desarrolla o fabrica la operación.
Descripción:	Descripción de las responsabilidades de la operación.
Nota:	Se especifican las restricciones de ejecución y los posibles modos de fallo.

Tabla 3.1 Plantilla para la documentación de interfaces

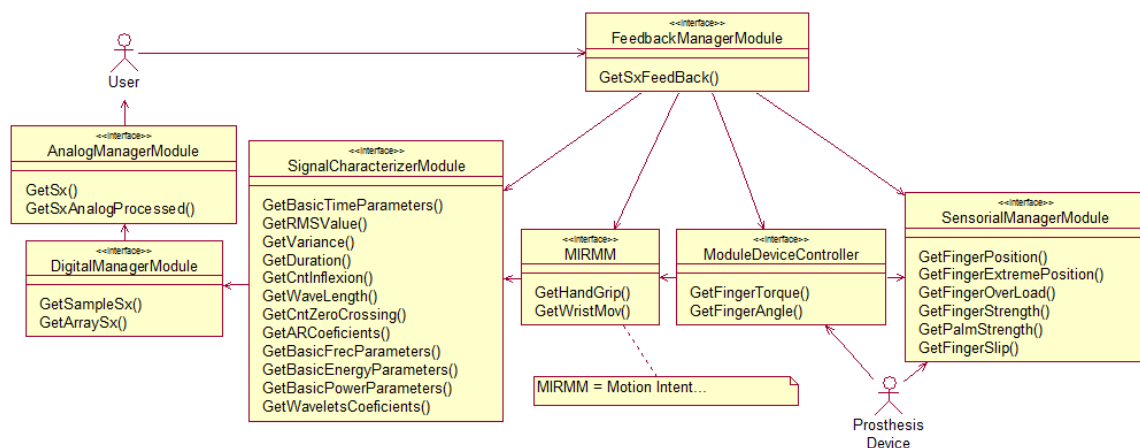


Figura 3.1 Diagrama UML del conjunto normalizado de interfaces para los componentes de un dispositivo prostético de mano.

Haciendo uso de la base conocimiento expuesta en el ítem Construcción de la base de conocimiento del presente documento se identificaron siete módulos los cuales conformarían a un dispositivo prostético de mano.

- Módulo Gestor Analógico (MGA).
- Módulo Gestor Digital (MGD).
- Módulo Caracterizador de Señal (MCS).
- Módulo Gestor de Reconocimiento de Intención de Movimiento (MGRIM)
- Módulo Controlador del Dispositivo (MCD)
- Módulo Gestor Sensorial (MGS)
- Módulo Gestor de Realimentación (MGR)

A cada uno de los anteriores módulos se les identificó las responsabilidades o funciones con el fin de conformar el conjunto normalizado de interfaces el cual es presentado en la Figura 3.1. Una descripción completa de cada una de las interfaces se encuentra en [49]. Sin embargo, es presuntuoso pensar que el conjunto de interfaces propuesto contiene todas las posibles estrategias de soluciones, aunque sí es posible afirmar que es un conjunto muy completo en el que se adecúan a una gran cantidad de soluciones.

Cada una de los módulos expuestos en el conjunto normalizado de interfaces es definido y documentado a través de IDL y la plantilla propuesta en la Tabla 3.1. Para el propósito del presente documento, el cual se enfoca en proponer una mejora para el proceso de desarrollo de dispositivos prostéticos de mano, se enseñará la manera de definir y documentar al menos una de las interfaces.

Como referencia se utiliza el Módulo Gestor de Realimentación (MGR), el cual fue descrito en el ítem 2.4 del presente documento. A partir de esta descripción se crea un modelo de análisis en UML haciendo uso de las clases interfaz, interfaz y control, el cual se presenta en la Figura 3.2.

Dado que la información por realimentar puede ser de distinta índole, por ejemplo, puede estar relacionada con la señal mioeléctrica, la fuerza transmitida, el instante en que el dispositivo entra en contacto con el objeto o incluso el nivel de batería, la clase *FeedBackControl* debe conocer la información generada por los demás módulos a través de sus interfaces.

De igual manera, la clase *FeedBackControl* dependiendo del hardware de realimentación implementado, es decir, si se trata de un LED, el cual genera una señal visual, o un circuito que suministra EEF, tendrá la capacidad de acceder a las clases entidad tales como *Visual* o *FES*, que se ilustran en la Figura 3.2. Dichas clases contendrán la información correspondiente a la señal de realimentación.

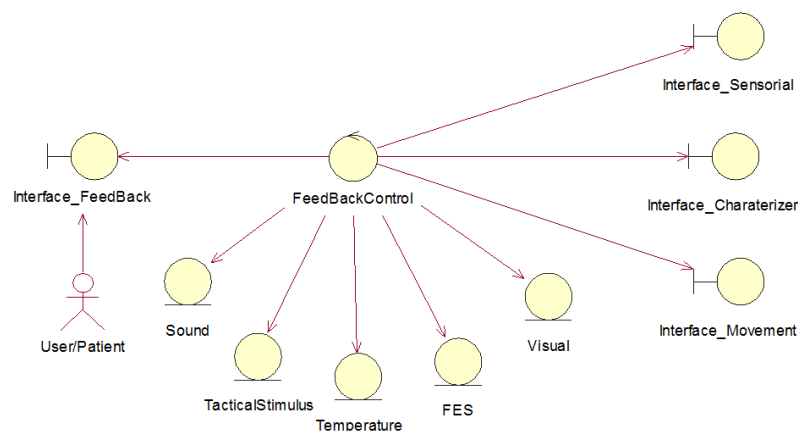


Figura 3.2 Modelo de Análisis del MGR

Por consiguiente, el Módulo Gestor de Realimentación (MGR) tiene como función fijar la intensidad de la señal transmitida al usuario de acuerdo con la información adquirida de los demás módulos y al tipo de hardware instalado en el dispositivo protésico.

A continuación se describe el MGR a través del lenguaje de descripción de interfaces, IDL, especificando el nombre y los argumentos de la responsabilidad o función que conforma la interfaz de realimentación.

```

Module FeedbackManager
{
  Interface Feedback
  {
    void GetSxFeedBack( in integer SxType, out Analog Value );
  }
}

```

A partir de la plantilla para documentación de interfaces en la Tabla 3.2 se presenta una posible documentación para la responsabilidad *GetSxFeedBack()*.

Módulo:	<i>Feedback Manager</i>	
Interfaz:	<i>Feedback</i>	
Responsabilidad:	<i>GetSxFeedBack(in integer SxType, out integer Value)</i>	
Arg. 1	SxType	Tipo de señal de realimentación.
Arg. 2	Value	Valor analógico que adquiere la señal de realimentación.
Tiempo de Respuesta:	No existen restricciones temporales estrictas, sin embargo, se recomienda que el tiempo de respuesta sea menor a 200ms, el cual corresponde a la ventana de análisis de la señal ME [50].	
Descripción:	<p>La presente responsabilidad tiene como finalidad obtener el tipo y el valor de la señal de realimentación al usuario. Tales señales pueden ser:</p> <ul style="list-style-type: none"> ▪ Estimulación Eléctrica Funcional (FES) ▪ Estimulación Táctil. ▪ Señales visuales ▪ Sonido. ▪ Temperatura <p>La información a realimentar puede estar relacionada con cualquiera de los estados del dispositivo protésico, por ejemplo: señal mioeléctrica, señal de fuerza, señal de vibración o deslizamiento, entre otras.</p>	
Nota:	A pesar que existen muchas sensaciones que se han perdido en un proceso de amputación se considera en [36] que las sensaciones de contacto, fuerza de agarre y deslizamiento son las más útiles para este tipo de dispositivo.	

Tabla 3.2 MGR. Obtener señal de realimentación

3.1 Conclusiones

En general, este trabajo servirá como marco de referencia para desarrollos e investigaciones futuras en el área de los dispositivos protésicos, el cual sería un importante aporte al conocimiento científico global en sistemas biónicos. Específicamente el conjunto normalizado de interfaces permitirá a posteriores iniciativas en el área optimizar la fase de integración en el proceso de

desarrollo de un simulador de prótesis robótica, así como la creación de una base teórica para aplicaciones en otras áreas.

Dada la baja interdependencia entre los módulos propuesto en el conjunto normalizado de interfaces se viabiliza la construcción de un diseño simple, segmentado, flexible y mantenible. Consecuentemente se mejora la sincronización entre los integrantes del grupo de desarrollo dado que cada uno de ellos puede trabajar de manera independiente, implementado su propia estrategia y acogiéndose a la definición del conjunto de interfaces propuesto.

Las interfaces planteadas permiten probar en un módulo en particular distintos algoritmos o estrategias de control, sin necesidad de modificar el resto de los módulos, de esta manera, conseguir el mejor ajuste o puesta a punto para el dispositivo protésico real

Las especificaciones de la interfaz suministrarán información a los desarrolladores del componente para que conozcan qué servicios están disponibles en otros componentes y cómo pueden acceder a ellos. Las especificaciones claras y no ambiguas de la interfaz del componente reducen las oportunidades de malas interpretaciones entre componentes que suministran un servicio y los que podrían utilizar ese servicio.

Generar el modelo de análisis haciendo uso del Lenguaje Unificado de Modelado UML permite abstraer en un solo plano de una manera detallada los componentes que integran un dispositivo protésico de mano.

4 Bibliografía

- [1] J Daintith. Encyclopedia. A Dictionary of Computing. [Online]. <http://www.encyclopedia.com/>
- [2] S. Orozco. Todos son Reemplazables, pocos reutilizables: Los componentes. [Online]. <http://www.liderdeproyecto.com/uml/uml010.html>.
- [3] C. E. Cerrano, *Modelo integral para el profesional en ingeniería*. Popayán: Editorial Universidad del Cauca, 2005.
- [4] J. S. Cabrera, "Conjunto normalizado de interfaces para los componentes de un simulador virtual de prótesis de mano robótica," Universidad del Cauca, Popayán, Documento para Examen de Valoración 2008.
- [5] A Alonso, R Sánchez, P Espino, R. D. la Rosa, and L L, *Myoelectric prostheses trainer for hand and arm amputees*, Mapfre Medicina, Ed., 2002, vol. 3.
- [6] F. Sebelius, L. Erikson, and C. Balkenius, "Neural control of a virtual prosthesis," *ICANN 98: Perspectives in Neural Computing*, 1998.
- [7] S. Ferguson and G. R. Dunlop, "Grasp recognition from myoelectric signals," in *Proc. 2002 Australasian Conference on Robotics and Automation*, Auckland, November 2002, pp. 83-87.
- [8] M Hauschild, R Davoodi, and G. E. Loeb. (2007) IEEE Transactions on Neural Systems and Rehabilitation Engineering. [Online]. <http://bme.usc.edu/assets/005/57578.pdf>
- [9] T. R. Farrell and R. F. Weir, "The optimal controller delay for myoelectric prostheses," *IEEE Transactions on neural systems and rehabilitation engineering*, vol. 15, no. 1, pp. 111-118, 2007.
- [10] L Birkedal et al. Sitio Web del Departamento de Ciencias de la Salud y la Tecnología de la Universidad de Aalborg. [Online]. [Disponible en : http://www.miba.auc.dk/](http://www.miba.auc.dk/)
- [11] M. Zecca, *On the development of a cybernetic prosthetic hand*. Pisa: Scuola Superiore Sant'Anna, 2003.
- [12] D Stegeman and H Hermens. (2007) Standards for surface electromyography: the european

- project "surface emg for non-invasive assessment of muscles (seniam). [Online].
<http://www.med.uni-jena.de/motorik/pdf/stegeman.pdf>
- [13] H. Romo, J. Realpe, and P. Jojoa, "Aplicación de señales emg superficiales y su aplicación en control de prótesis de mano," *Avances en Sistemas e informática*, vol. 4, no. 1, pp. 127-136, 2007.
- [14] C. J. De Luca, *Encyclopedia of Medical Devices and Instrumentation*, 2nd ed., John G. Webster, Ed. Boston: Wiley, 2006, vol. 6.
- [15] C. J. De Luca. (2007, Octubre) Sensor Related Factors and Fidelity of the sEMG Signal. [Online].
http://nmrc.bu.edu/tutorials/sensor_facts.pdf
- [16] Care M. H. (2008) Biomedical electrodes.. [Online]. <http://www.3m.com/>
- [17] Seniam. (2007) Surface ElectroMyoGraphy for the Non-Invasive Assessment of Muscles. [Online]. <http://www.seniam.org/>
- [18] K Englehart, B Hudgins, P Parker, and M Stevenson, "Classification of the myoelectric signal using time-frequency based representations," *Special Issue of Medical Engineering and Physics on Intelligent Data Analysis in Electromyography and Electroneurography*, 1999.
- [19] C. J. De Luca, "The use of Surface Electromyography in Biomechanics," *Journal of Applied Biomechanics*, vol. 13, pp. 135-163, 1997.
- [20] J. A. Cortez, F. A. Medina, and J. A. Chaves, "Del análisis de fourier a las wavelet : Análisis de fourier," *Scientia Et Technica*, vol. 13, no. 34, pp. 151-156, 2007.
- [21] J. R. Azor. (Noviembre, 2009) La transformada de gabor. [Online].
<http://www.um.edu.ar/math/wavelets/gabor.htm>
- [22] A. F. Salazar, "Diseño e implementación de un sistema de control que utilice señales electromiográficas para activar una prótesis mioeléctrica," Escuela de Ingeniería de Antioquia, Envigado, Trabajo de grado 2006.
- [23] S Bing and K Ikeuchi, *A framework for Recognizing Grasps*. Pennsylvania, 1991.
- [24] C Gaviria, J Diaz, and V Mosquera, "Agarre estable de objetos con una prótesis de mano robótica: Informe final," Universidad del Cauca, Popayán, Reporte de resultados de investigación, Proyecto Prótesis Robótica de Mano 2008.
- [25] A. Orozco and E. Giraldo, "Caracterización dinámica de señales electromiográficas usando filtros adaptativos," *Scientia et Technica*, no. 34, pp. 19-24, Mayo 2007.
- [26] Md. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "EMG Signal Classification for Human Computer," *European Journal of Scientific Research*, vol. 33, no. 3, pp. 480-501, 2009.
- [27] J. Kohlmorgen and B. Blankertz. (2007) Bayesian Classification of Single-Trial Event-Related Potentials in EEG. [Online]. <http://eprints.pascal-network.org/>
- [28] J. M. Marín, "Análisis Discriminante Lineal," Universidad Carlos III de Madrid, Madrid, Notas de clase 2009. [Online]. <http://halweb.uc3m.es/>
- [29] L Hargrove, K Englehart, and B Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," , 2005, pp. 5009-5012.
- [30] F. J. Betancourt and E Giraldo, "Reconocimiento de patrones de movimiento," *Scientia et Técnica*, vol. 26, pp. 53-58, 2004.
- [31] A Soares, A Andrade, E Lamounier, and R Carrijo, "The Development of a Virtual Myoelectric Prosthesis Controlled by an EMG Pattern Recognition System Based on Neural Networks," in *Journal of Intelligent Information Systems*, vol. 21, 2003, pp. 127 - 141.

- [32] H Hanafusa and H Asadas, "A robotic hand with elastic fingers and its application to assembly process," in *IFAC Symposium on Information and Control Problems in Manufacturing Technology*, Tokyo, 1977.
- [33] M Mason and J.K Salisbury, "Robot Hands and the Mechanics of Manipulation," Cambridge University, Cambridge, Informe de Investigación 1985.
- [34] L.E. Rodriguez and A Casals, "Sensing and control of a prosthetic hand with myoelectric feedback," *IEEE Conference Biomedical Robotic And Biomechatronic*, pp. 607 - 612, 2006.
- [35] J. F. Franco, E Giraldo, and A Orozco, "Reconocimiento de patrones de movimiento del sistema fisiológico brazo-antebrazo, a partir de señales electromiográficas," Universidad Tecnológica de Pereira, Pereira, 2004.
- [36] L.E. Rodriguez, "Evaluación de funcionalidad de las prótesis de mano mioeléctricas y realimentación de los esfuerzos a los pacientes por medio de estimulación eléctrica de baja potencia," Universidad Politécnica de Cataluña., Barcelona, Tesis doctoral 2005.
- [37] A Duarte et al., "Sensores de contacto, fuerza y deslizamiento para manos protésicas," in *Iberdiscap*, Madrid, 2000.
- [38] E. Aguilar, "Modelo y control de posición y de fuerza de una prótesis robótica de mano," Universidad del Cauca, Popayán, Examen de valoración 2008.
- [39] Thomas H. Speeter, "A Tactile Sensing System for Robotic Manipulation, The International Journal of Robotics Research," , vol. 9, 1990, pp. 25-36.
- [40] J Jockusch, J Walter, and H Ritter, "A Tactile Sensor System for a Three-Fingered Robot Manipulation," in *Proc. Int. Conf. on Robotics and Automation (ICRA) IEEE*, 1997.
- [41] P. Henri, A. Nahvi, R. Rohling, P. Zhang J.M. Hollerbach, "Utah/MIT Dextrous Hand Project," Utah and MIT, Boston, 2004.
- [42] J.I. Gomez, J.A. Mendoza, and A Salazar, "Biorealimentación en el análisis y rehabilitación muscular," *Scientia Et Technica*, 2007.
- [43] H Brittain, R.R Caldwell, A.B Cameron, and V.A Dunfield, "Sensory-feedback system compatible with myoelectric control," *Med. & Biol.Eng. & Comput*, vol. 18, pp. 65 - 69, 1980.
- [44] Jack Ganssle, *Embedded System: World Class Desings*. Burlington: newnespress, 2007, vol. 1.
- [45] Asería Informática. (2009, Julio) Optimizando la tecnología de la información. [Online]. http://www.asesoriainformatica.com/definiciones_i.htm
- [46] J. Ganssle. (2004, Enero) 80% of all embedded systems are delivered late. [Online]. <http://www.ganssle.com/fsm.pdf>
- [47] OMG Object Management Group. (2009) CORBA 3.0 - OMG IDL Syntax and Semantics chapter. [Online]. <http://www.omg.org/cgi-bin/doc?formal/02-06-39.pdf>
- [48] W. H. Glauert. (2009) VHDL Tutorial. [Online]. <http://www.vhdl-online.de/tutorial/>
- [49] J. S. Cabrera, "Conjunto Normalizado de Interfaces para los componentes de un simulador virtual de prótesis de mano robótica," Universidad del Cauca, Popayán, Tesis de Maestría. 2009.
- [50] Richard F. Weir Todd R. Farrell, "The Optimal Controller Delay for Myoelectric Prostheses," *IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING*, vol. 15, no. 1, pp. 111-118, 2007.