

Trabajo de Fin de Máster
Máster en Computación

**DESARROLLO DEL SOFTWARE
DE UN SISTEMA
PARA CALIBRACIÓN
DE MEDIDAS HORIZONTALES**

Octubre 2013

**Facultad de Ciencias
Universidad de Cantabria**

Autor: Pablo Gutiérrez Peón

Director: Michael González Harbour

Introducción

Descripción del sistema

Selección de tecnologías

Diseño de la arquitectura software

Conclusiones y trabajo futuro

Contexto de desarrollo

Trabajo de colaboración



Grupo de Computadores
y Tiempo Real (CTR)

&



Laboratorio de
Automática

Desde 1987, múltiples robots para automatización de procesos.

Destinatario del proyecto actual



Laboratorio de
Metrología

- Proporciona medidas de precisión.
- Calibración y certificación de máquinas y equipos en las áreas dimensional, temperatura, electricidad, presión y fuerza.

Medidora de coordenadas horizontales (I)

Medición de longitudes (área dimensional)

- ⦿ Grandes longitudes (de hasta 4 m en una sola medición)
- ⦿ Medición de reglas, cintas métricas, barras y piezas en general.

Medidora de coordenadas horizontales (II)

Aspecto físico del instrumento



Objetivo principal

Desarrollo del software del sistema de calibración sobre medidora de coordenadas horizontales.

Servo digital

Monitorización y control del movimiento mediante servo digital con interfaz **CAN** y protocolo **CANopen**.

Interfaz gráfica

Basada en **GtkAda** y que permita monitorizar y controlar el servo, además de visualizar la imagen del punto de calibración procedente de una **cámara**.

Resultados adicionales

Driver Windows NT para tarjeta E/S digitales

- Tarjeta modelo PCI-1761 de Advantech.
- 8 entradas y 8 salidas digitales.

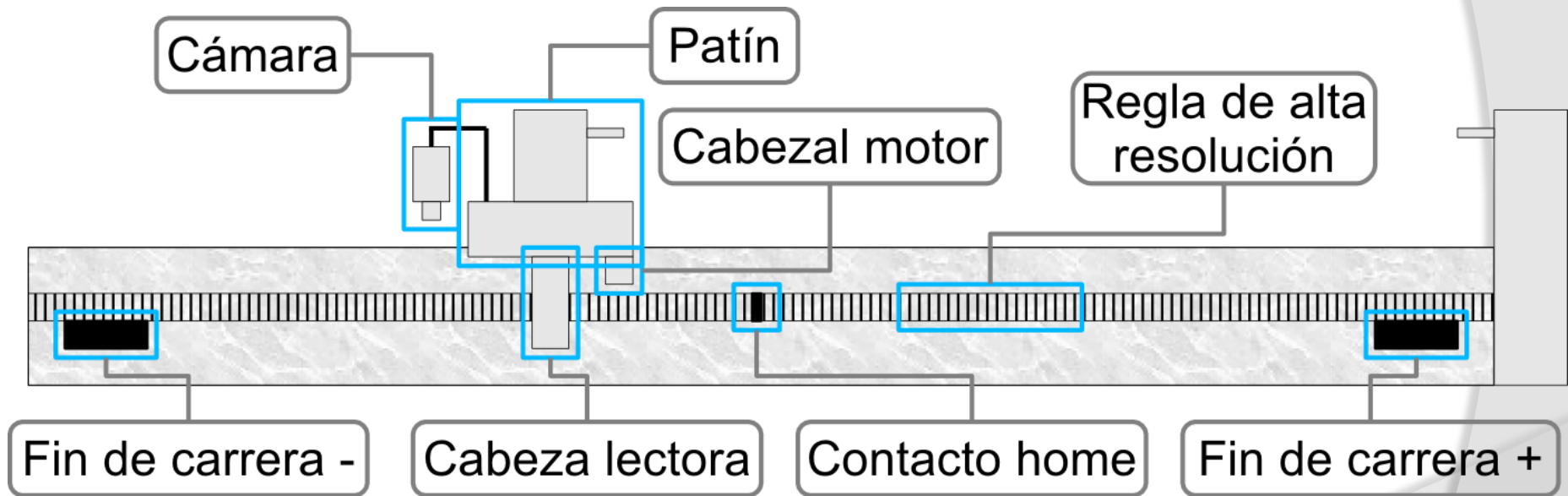
Librerías de manejo de hardware

- **Cámara digital** DALSA basada en estándar de transmisión de vídeo y control de dispositivos GigE Vision.
- **Tarjeta de comunicaciones CAN** PCI-1680U de Advantech.

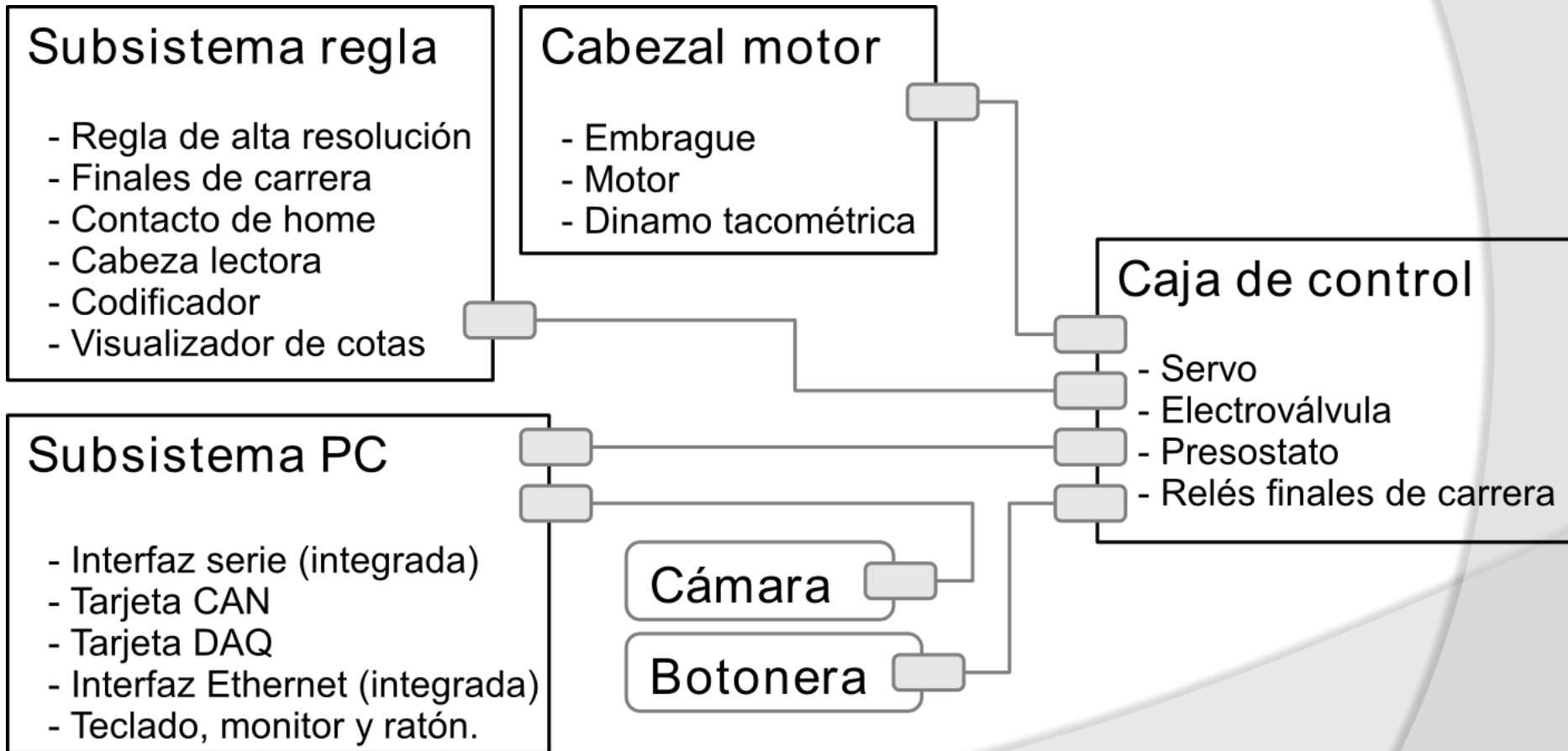
Librería Ada para soporte CANopen

- Soporte a subconjunto de CANopen necesario en este proyecto.
- Modular y extensible.

Esquema físico



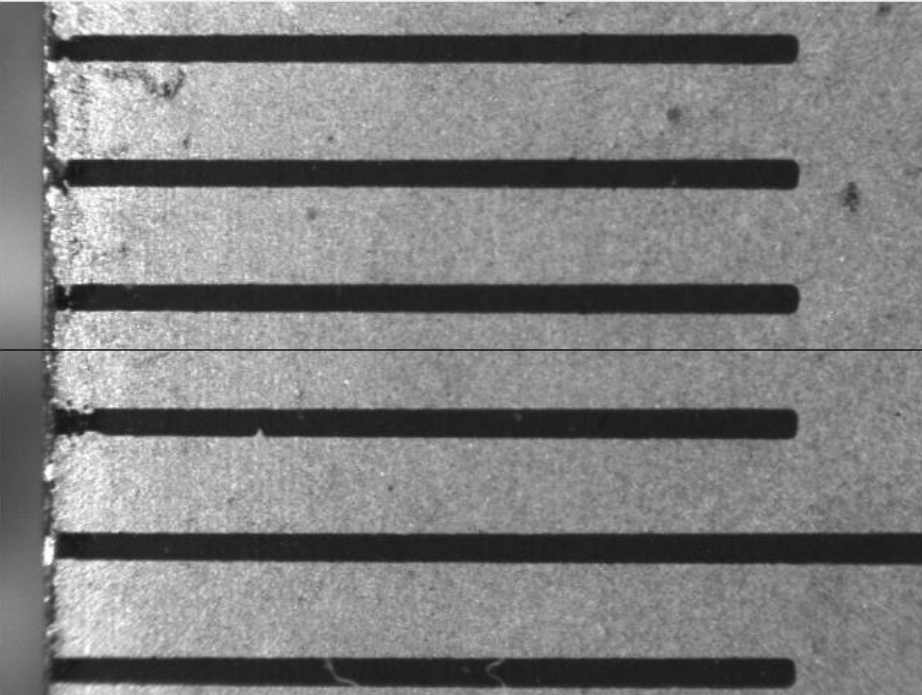

Esquema físico de componentes



Especificación software

UC Sistema de calibración de reglas

Sistema de calibración de reglas



Movimiento

Mover automático

Consigna (mm) =actual

Incremento (mm) Añadir

Mover manual

Velocidad

Operaciones

Hacer cero

Calibración

Identificación

Liberar cabezal

Sí No

Posición actual (mm)

Avisos

Lenguajes de programación

Ada

- ⦿ Lenguaje de referencia en el grupo CTR.
- ⦿ Para controladores industriales, robots e instrumentación.
- ⦿ No aplicable a todo el sistema debido a heterogeneidad de sus componentes.
- ⦿ GtkAda
 - Librería Gtk+ del escritorio GNOME.
 - Codificación de interfaz mediante ficheros XML.

C/C++

- ⦿ Lenguajes fáciles de encontrar en entornos hardware y de desarrollo de SO.
- ⦿ Tarjetas CAN y DAQ y cámara programables mediante C/C++.

Sistema operativo Windows

Argumentos de elección

- ⦿ Familiaridad de los usuarios finales.
- ⦿ Restricciones software.
 - Controladores y librerías disponibles para el SO (a veces de forma única).
- ⦿ Restricciones de tiempo real.
 - Control del movimiento realizado por el servo. Descarga de responsabilidad de operaciones de tiempo real.

Controladores de dispositivo WDM

- ⦿ Problema con tarjeta DAQ \Rightarrow Necesidad de controlador propio.
- ⦿ Windows Driver Model (WDM) \Rightarrow Plug & Play y abstracción sistema de ficheros.
- ⦿ Estructurado en funciones.
 - Inicialización y limpieza (entrar y salir del controlador).
 - Tratamiento de peticiones de E/S (abrir/cerrar, leer/escribir).
 - Manejadoras de mensajes Plug & Play (añadir dispositivo, detener, etc.).
- ⦿ Instalación automática mediante ficheros INF.

GigE Vision: sistema de vídeo

- ⦿ Estándar para cámaras digitales industriales.
- ⦿ Tecnología digital frente a analógica.
- ⦿ Extendida en el catálogo de cámaras industriales.
- ⦿ Emplea Ethernet.

Servos CAN de Advanced MotionCtrls.(I)



Características

- Servo digital.
 - Probar nueva tecnología que sustituya a los servos analógicos.
- Interfaz de comunicación CAN.

Servos CAN de Advanced Motion Ctrls.(II)

Interfaz CAN

- ⦿ Comunicación en sistemas de control.
- ⦿ Creado por Robert Bosch GmbH para el sector del automóvil.
 - Extendido en el mundo industrial.
- ⦿ Define **niveles físico y de enlace** OSI (cableado, acceso al medio, trama de datos).

Protocolo CANopen

- ⦿ Implementación de **capa de aplicación** OSI sobre CAN.
- ⦿ Concepto central \Rightarrow **Objetos** (@ memoria)
- ⦿ Cada dispositivo cuenta con un objeto por cada parámetro que deba ser almacenado (**diccionario de objetos**).
- ⦿ Tipos de mensajes: **NMT** (gestión de la red), **SDO** (lectura/escritura de objetos).

Servos CAN de Advanced Motion Ctrls.(III)

CANopen en los servos de AMC

- ⦿ Funcionamiento descrito a través del diccionario de objetos.
- ⦿ Control sobre...
 - Máquina de estados del servo.
 - Homing.
 - Modos de operación.
 - ...

Driveware como herramienta de configuración

- ⦿ Driveware + comunicación por puerto serie = Configuración del servo.
- ⦿ Permite ajustar...
 - Parámetros del motor.
 - Parámetros de retroalimentación de posición y velocidad.
 - Asignar entradas y salidas.
 - Gestión de eventos.
 - ...

Ciclo de vida (I)

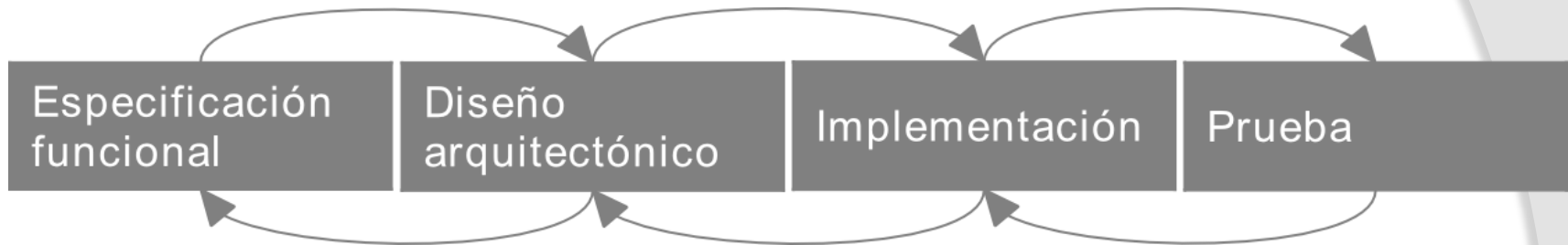
Fases finales: explotación y mantenimiento

- ⦿ Soporte técnico.
- ⦿ Aceptación de sugerencias de mejora.

Fase inicial: desarrollo

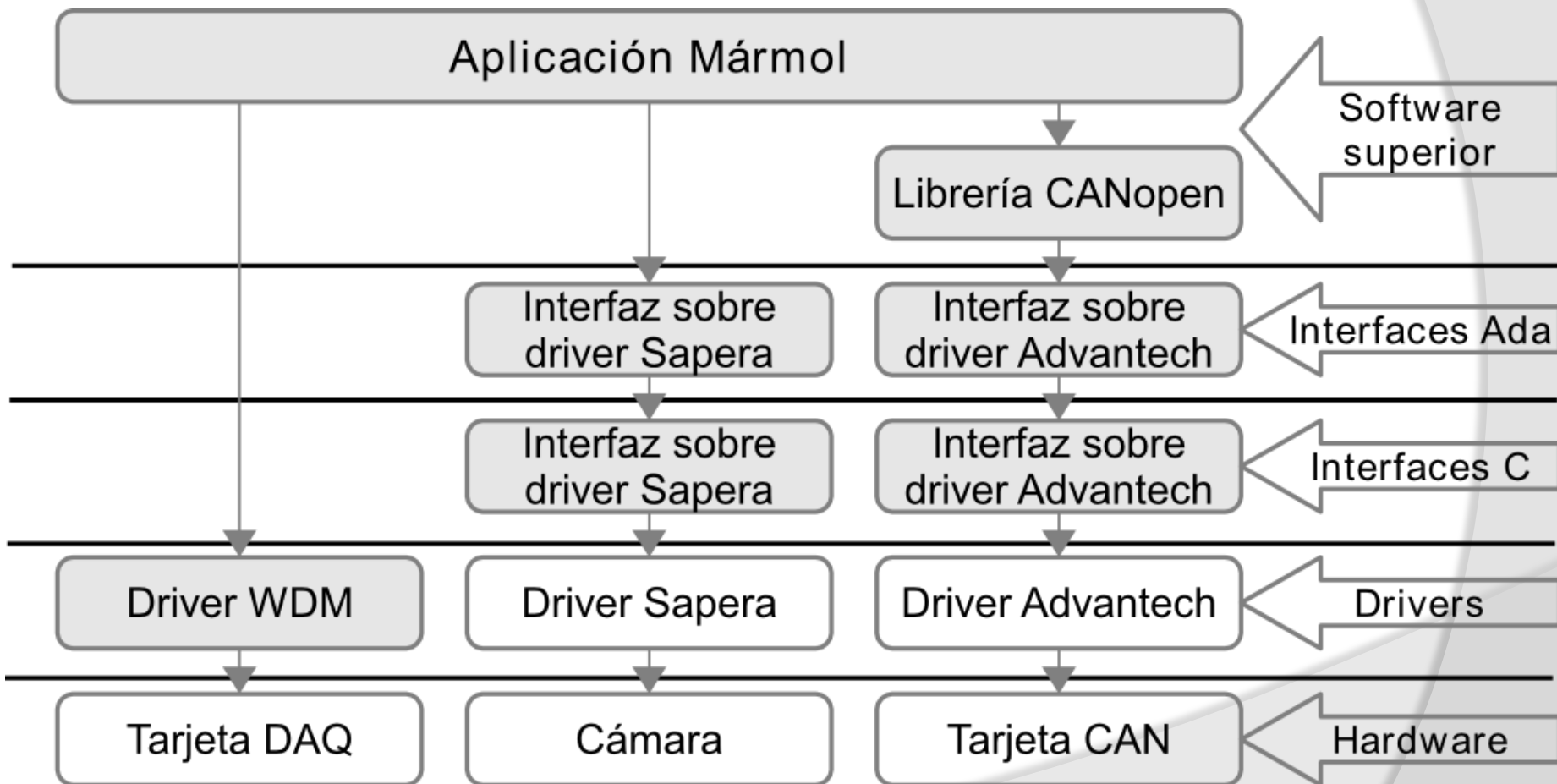
- ⦿ 4 procesos:
 - Especificación funcional.
 - Diseño arquitectónico.
 - Implementación.
 - Pruebas.
- ⦿ Abordados mediante modelo incremental.
 - Evolución iterativa del software desde modelos y prototipos hacia el sistema final.

Ciclo de vida (II)

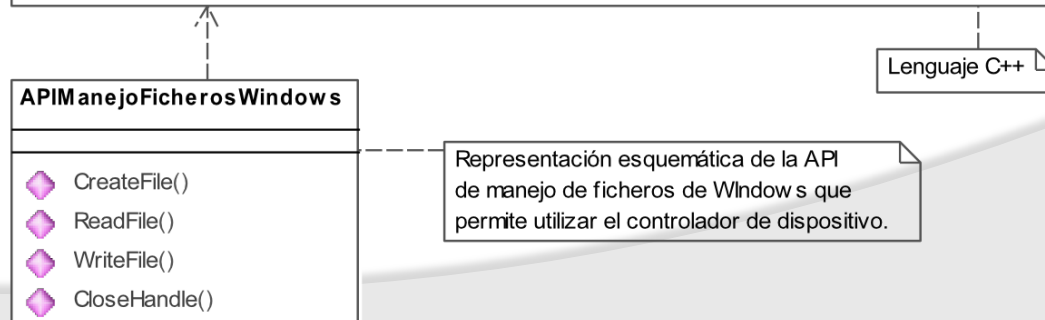
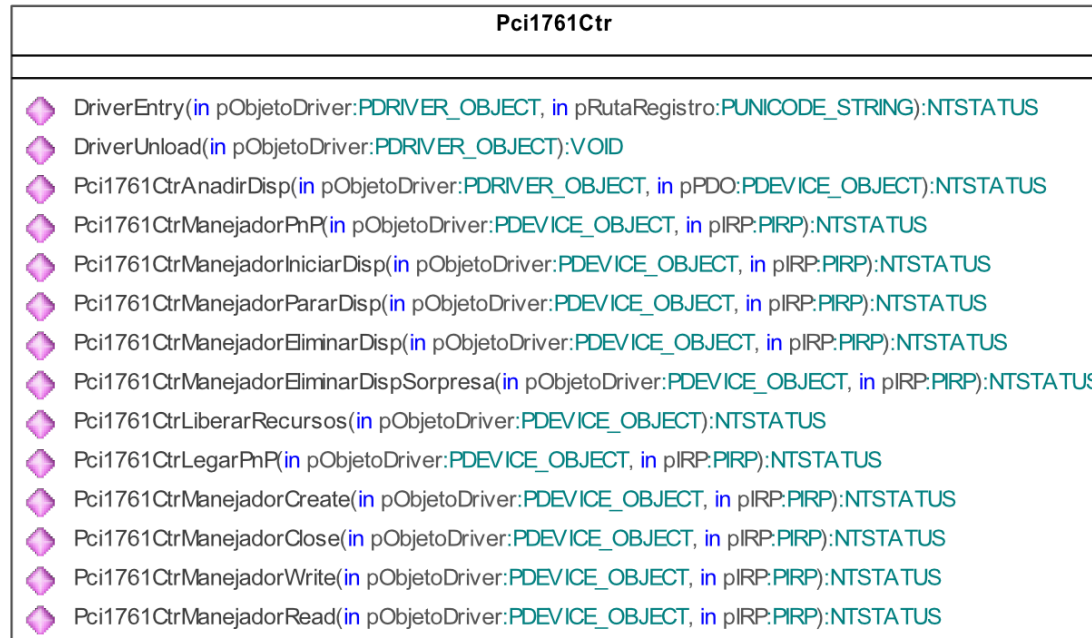


1. Software para manejo de prototipo hardware.
Demostración de la tecnología del servo.
2. Prototipo rápido de la GUI.
Recibir sugerencias de usuarios finales.
3. Aplicación completa sobre prototipo hardware.
4. Aplicación completa sobre entorno hardware final.

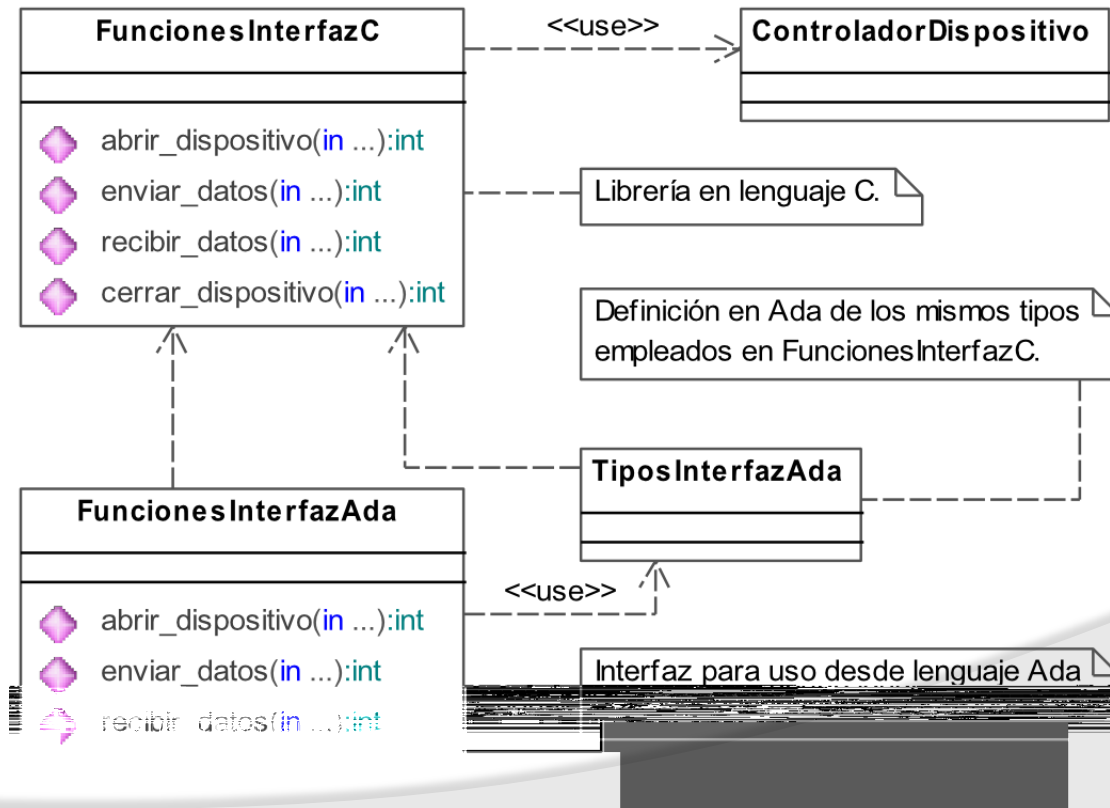
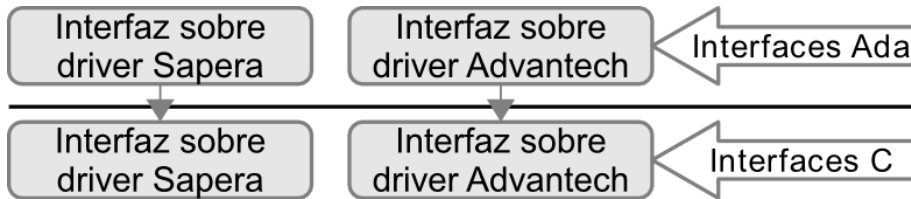
Visión general de la arquitectura



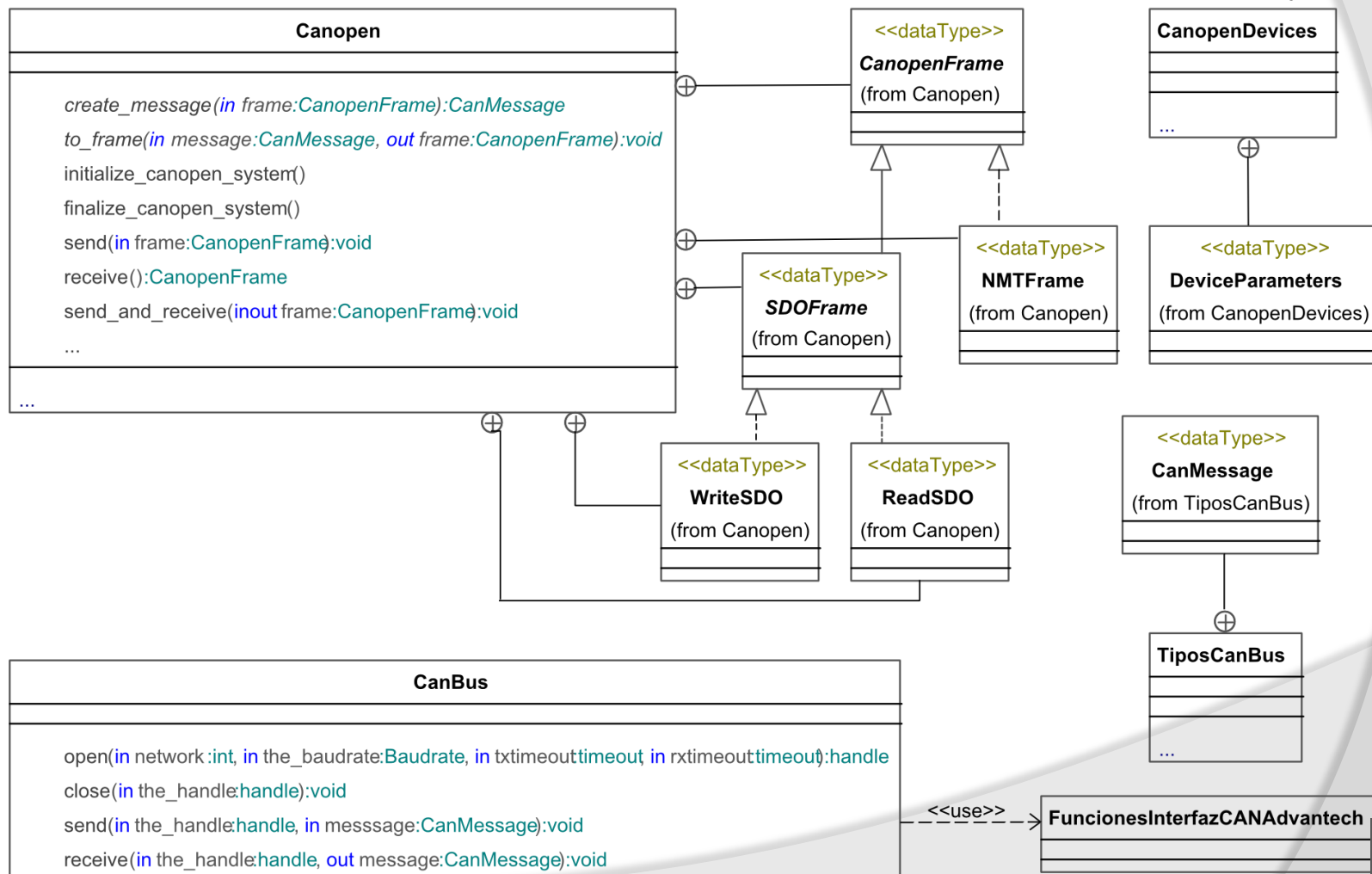
Controladores de dispositivo



Interfaces sobre controladores de dispositivo



Librería CANopen



Mármol (I)

Aplicación Mármol

Software superior

Diagramas de clases

- ◉ Ordenan responsabilidades, evita ambigüedades y facilita el mantenimiento.
- ◉ Diseño dividido en 3 partes:
 - GUI
 - Servo
 - Global

Diagramas de comunicación

- ◉ Muestran la interacción entre las instancias de las clases. Permite verificar la validez del diseño de clases realizado.
- ◉ Diseño dividido en 3 partes:
 - GUI
 - Servo
 - Global

Mármol (II)

Aplicación Mármol

Software superior

Diagramas de estados

- ⦿ Modelado de máquinas de estados. Clarificar transiciones entre estados:
 - ¿Qué antecedentes han llevado a esta situación?
 - ¿Qué posibilidades se despliegan?
- ⦿ Máquinas de estados manejadas:
 - Estado de control del servo (interna del dispositivo servo).
 - Estado del módulo servo.
 - Estado de la aplicación.

Diagramas de actividad

- ⦿ Representan de forma gráfica algoritmos.
- ⦿ Describe el algoritmo del hilo de control sobre el servo.

Mármol (III)

Aplicación Mármol

Software superior

Clases activas

- ⦿ Contienen hilos de ejecución
 - Dan lugar a actividades de control.
 - Implican concurrencia.

Secciones protegidas

- ⦿ Surgen por la existencia de concurrencia.
 - Regulan el acceso a datos compartidos.
 - Establecen exclusión en el uso de sus funciones.

Conclusiones

Objetivo principal satisfecho

Laboratorio de Metrología de ENSA dispone de un software de calibración para la medidora de coordenadas horizontales.

- Útil.
- Cumple sus expectativas.
- Pueden y hacen uso de él.

En base a la tecnología planteada

Tecnologías investigadas y funcionalidad evaluada.

- Servo digital CANopen.
- GtkAda.

Nuevos módulos software

- Controlador de dispositivo Windows NT para tarjeta de E/S digitales.
- Librerías C/Ada para manejo de cámaras digitales GigE Vision y tarjeta de comunicaciones CAN.
- Librería CANopen para Ada.

Trabajo futuro

- ⦿ Sistema de máximos y mínimos.
 - Para calibración de barras.
- ⦿ Movimientos mediante jostick.
 - Controlando el sentido y velocidad del movimiento.
- ⦿ Línea de referencia de la imagen de vídeo.
 - Para facilitar la búsqueda del punto de calibración.
- ⦿ Sistema de doble coordenada.
 - Por cada coordenada X, tomar una coordenada Y.
- ⦿ Programador de movimientos automáticos.
 - Para automatizar más el proceso de calibración.

Trabajo de Fin de Máster
Máster en Computación

**DESARROLLO DEL SOFTWARE
DE UN SISTEMA
PARA CALIBRACIÓN
DE MEDIDAS HORIZONTALES**

FINAL DE LA EXPOSICIÓN