1 Agradecimientos

Quiero agradecer al estado en democracia y a los ciudadania que lo compone por haberme brindado la educacion publica y gratuita la cual me brindo por medio de los docentes una educacion de calidad y competitiva.

Recuerdo mis primeros dias en FaMAF, hice mi ingreso en Agosto por adelantado para tener mis vacaciones extendidas en verano y no cursar en Febrero el ingreso.

Los 3 primeros años fueron durisimos. Luego comence a trabajar para hacer algo distinto. Volvi con muchas ganas despues de ponerme de novio buscando mas conocimiento y experiencia. Termine de cursar y volvi a trabajar, en cosas mucho mejores.

El espiritu de los docentes me mantuvo vivo, esto es fruto de eso.

Muchas cosas decantan con el tiempo, sobretodo las ideas mas profundas.

Ciencias de la computacion es la carrea que volveria a elegir si tuviese que volver a empezar, por el temple de mis docentes y lo que significa la computacion en estos tiempos.

Este camino no termina aca, "te sobran años Bovina" una frase de un profesor que me enseño que el tiempo aun lo tengo, que vale y lo tengo que usar muy bien.

Gracias FaMAFC, familia, amigos y director de tesis!

Paciencia Paciencia Más Paciencia!

era verdad ...

2 Marco de trabajo e investigacion

En el proceso de investigacion del fenomeno fisico de Resonancia Magnetica Nuclear el investigador manipula modulos electronicos digitales de medicion precisos, estables y en algunos casos si intervienen mas de uno a la vez entre ellos sincronizados por medio de interfaces digitales. Estos modulos electronicos digitales colaboran con la creacion del contexto necesario para investigar lo planeado segun la necesidad del investigador. En general junto con los modulos electronicos intervienen otros modulos electronicos de naturaleza analogica tales como amplificadores operacionales, mezcladores de señales, filtros pasa bajos entre otros.

La correcta conexion entre los diferentes modulos electronicos digitales, analogicos y su configuracion durante el proceso de experimentacion son responsabilidad del investigador y una tarea de suma importancia para el exito de la experiencia a realizar.

En este marco de responsabilidades del investigador y el modulo digital a ser utilizado es deseable y necesaria mecanismos sencillos de manipulacion del mismo y su monitoreo.

2.1 Rol del Software

El rol primario del software en el contexto descripto previamente es el de manipular el modulo digital a traves de la PC utilizada por el investigador de forma local o remota.

El rol secundario la administracion de usuarios del modulo digital, monitoreo de los experimentos en curso y resultados.

3 Partes de un experimento

Existen ciertos elementos tanto humanos como materiales en la realizacion de un experimento junto con un simple aunque muy importante uso de datos, es decir que cada uno de estos elementos interactuan compartiendo informacion que pueden o no determinar acciones.

3.1 Tipo de Experiencia

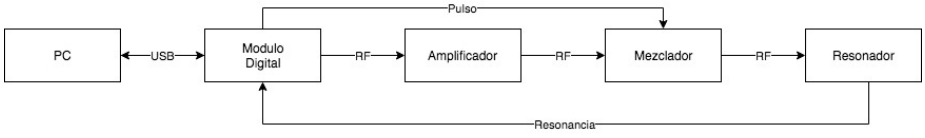
El tipo de experiencia a realizar esta determinada por el investigador y/o area, junto a parametros de planificacion como por ejemplo la duracion de la misma, la materia a estimular, el tipo de espectromentro, frecuencias de estimulacion, numero de instantaneas por periodos especificos, tiempos de estimulacion, relajacion, estabilizacion.

3.2 Sincronizacion

Algunas experiencias requieren o se complementan con la sincronia entre mudulos digitales de medicion via interfaces digitales.

4 Diagrama de conexiones

Este diagrama representa la interconexion entre los diferentes modulos digitales y analogicos que forman parte de una experiencia planificada.

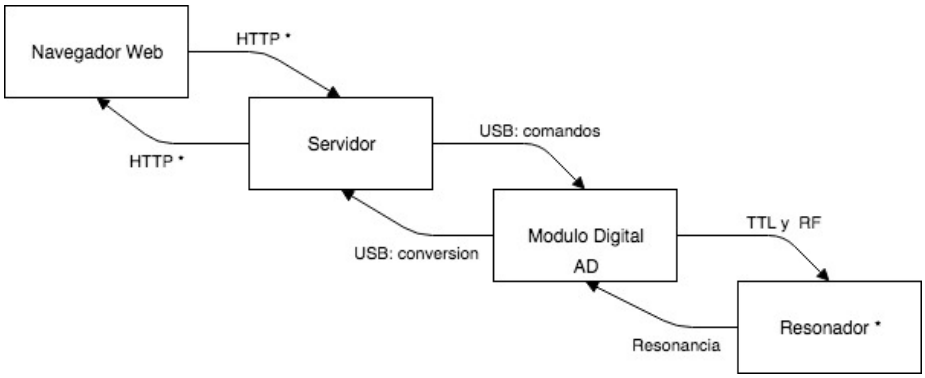


.

Figure 1: Diagrama de conexiones

5 Vision general del sistema

Este diagrama respresenta la interacion entre los diferentes elementos de hardware y software en una experiencia planificada.



.

Figure 2: Vision General del sistema

6 Descripcion del modulo digital

El siguiente diagrama de bloques muestra la configuracion interna del aparato junto a sus caracteristicas tecnicas.

7 Submodulo DDS2

Esta seccion describe en detalle el funcionamiento del submodulo dds2

8 Submodulo AD

9 Submodulo PP2

10 Submodulo USB

11 Definicion de un experimento

Esta seccion describe como se define un experimento.

En la actualidad una numerosa cantidad de proyectos utilizan notacion JSON para representar objetos por muchas razones entre ellas su facilidad para ser comprendido por parte de desarrolladores y el soporte buil-in para ser manipulado por navegadores web y nuestros lenguajes elegidos para desarrollar, python y javascript.[https://es.wikipedia.org/wiki/JSON]

Idealmente estos objetos son tambien sencillos de almacenar en bases de datos no relacionales aunque de esto haremos mencion mas adelante en el capitulo ZZ.

Haciendo uso de aritmetica podemos deducir que la representacion de un experimento es liviana puesto que el numero de instrucciones de un programa

P sera de N ¡ 999 instrucciones lo que nos daria un peso aproximado en MB ¡ Y MB por experimento representado en esta notacion.

La validacion del esquema JSON de un experimento, sera llevada a cabo por un modulo el cual haremos mencion en el capitulo ZZ.

Cabe destacar que los tipos de datos presentes en la especificacion de un experiemento son soportados sin perdidad de presicion por parte de la notacion JSON.[https://stackoverflow.com/questions/35709595/why-would-you-use-a-string-in-json-to-represent-a-decimal-number]

12 Algoritmo de traduccion de experimentos

Esta seccion describe como un experimento es traducido a instrucciones que el PP2 pueda almacenar y ejcutar.

Como mencionamos en el capitulo XX un programa almacenable y ejecutable por el PP2 es una secuencia de instrucciones I1 .. IN con N no mayor a K, donde K esta relacionado al limite de memoria del mismo es de XX MB.

Las instrucciones disponibles son:

*continue *loop *retl *fin

No todos los programas escritos con estas instrucciones son admitidos como validos para el PP2, por esto tenemos las siguientes reglas que deben cumplir los programas ejecutables por el PP2.

Regla 1: la instruccion fin siempre es la ultima instruccion.

Regla 2: la intruccion fin aparece solo una vez.

Regla 3: la intruccion fin siempre esta presente.

Regla 4: un bloque siempre inicia con instruccion loop y finaliza con retl

Regla 5: puede haber hasta 3 loops dentro de un loop

Estas reglas definen el siguiente automata:

Resultando en el siguiente algoritmo base:

traslate(P, L, S)

ins = next(P)

if ins = Continue: S.append(ins)

if ins = Retl: if(L!=0): S.append(ins) L– else: ret Error

if ins = Loop: if(L¡4): S.append(ins) L++ else: ret Error

ret S

13 Ejecucion de un experimento

Esta seccion describe los conceptos dry-run main-run y algunos escenarios de ejecucion de experimentos cubiertos por la implementacion.

Un modo de capturar fallos inesperados es simular la ejecucion de un experimento evitando la interaccion con el modulo digital via usb configurando el sistema en modo simulado.

En modo simulado el experimento se ejecuta en un entorno controlado con el objetivo detectar escencialmente 3 situaciones no deseadas:

* una exepcion de software no capturada * parametros fuera de rango para alguno de los submodulos * un error en la traduccion del experimento a un programa ejecutable en el PP2

Este experimento simulado no se ejecuta en un hilo separado puesto que la demora es baja.

De manera alternativa el modo simulado puede verse tambien como un test de integracion entre las unidades de software desarrolladas.

Si el experimento simulado tiene exito entonces se procede con la ejecucion efectiva del experimento el cual debemos tener algunos detalles en cuenta:

* puede durar horas * puede ser cancelado en cualquier momento * debe estar sincronizado con la base de datos

El siguiente diagrama clases muestra el diseño de un experimento simulado y un experimento.

14 Hilo de ejecucion de experimento

Esta seccion describe porque y como el concepto de hilos aplica a la ejecucion de un experimento.

La duracion de un experimento puede ser de horas, esta caracteristica involucra tres problematicas a resolver:

* evitar time-out del lado del cliente * desacoplar la atencion de una peticion de usuario de la ejecucion de un experimento * lograr una experiencia de usuario agradable

Modelando el proceso de ejecucion con un hilo separado del principal es una aproximacion valida para resolver el problema y por lo tanto tener en cuenta los siguientes:

* seguimiento del estado del hilo * control sobre el numero de hilos creados * gestion de los datos durante la ejecucion del hilo

Implementar un administrador de ejecucion de experimentos es la solucion en este escenario.

El siguiente diagrama muestra el diseño del administrador.

15 Diagrama de flujo de ejecucion de un experimento

El siguiente diagrama describe la ejecucion de un experimento dry-run y main-run

16 Requisitos del sistema

Podemos ver los requerimientos en generales

* permitir a mas de un usuario utilizar el sistema * permitir a usuarios la gestion de experimentos * modelar un experimento y su ejecucion * proveer acceso remoto al sistema

De estos requerimientos se desprenden los siguientes especificos:

* el usuario tiene una sesion asignada al ingreso del sistema y revocada a la salida del mismo. * varios usuarios pueden acceder al sistema al mismo tiempo

* el usuario puede ver si hay un experimento en ejecucion y el detalle del mismo * el usuario tiene un espacio de trabajo donde puede crear/ver/actualizar/eliminar sus experimentos * el usuario puede cancelar el experimento en ejecucion * el usuario puede solicitar un reporte parcial de un experimento en ejecucion * el usuario puede solicitar el reporte final de un experimento finalizada la ejecucion * el usuario puede verificar el estado de un experimento: * el usuario es notificado antes de la ejecucion de un experimento cuando: * salida voluntaria por error: * no es una secuencia ejecutable por el PP2 * ya existe algun experimento ejecutandose * algun parametro de configuracion es invalido en algun submodulo PP2, DDS2, AD, USB * fallo en conexion via USB * salida involuntaria por error: * hubo alguna excepcion de sistema no controlada * no se pudo establecer conexion con el servidor * un experimento finalizado no actualizo su estado impidiendo la ejecucion de solcitido

* un experimento tiene una secuencia definidida con sus parametros * un experimento tiene una marca de tiempo de creacion/actualizacion * un experimento eliminado no es recuperable * un experimento tiene un autor que es el usuario que lo creo * un experimento tiene estado created cuando esta almacenado en base de datos * un experimento es solo visible para el usuario que lo creo * un experimento tiene un titulo * un experimento tiene una descripcion * un experimento no tiene un historial de edicion asociado

* un resultado es el producto de la ejecucion de un experimento * un resultado es parcial cuando la ejecucion del experimento asociado no finalizo * un resultado es final cuando la ejecucion del experimento asociado finalizo * un resultado tiene un unico experimento asociado

* un reporte parcial es el grupo de datos de un resultado parcial * un reporte final es el grupo de un resultado final * un reporte contiene: * log de la ejecucion * datos del AD en formato CSV * experimento ejecutado

* el sitema tiene servicios REST para: * crear/ver/editar/eliminar experimentos * iniciar/cancelar ejecucion de un experimento * cancelar la ejecucion de todos los experimentos * descargar reportes parciales y finales * proveer autenticacion a todos los servicios

* el sistema tiene una interfaz de usuario web

17 Planificacion del desarrollo

De acuerdo con los requerimientos descriptos en el capitulo XX se propone implementar el sistema con:

* Django framework - Pythton 2.7 * React js - Javascript ECM6 * Git

Las tecnologias elegidas son ampliamente soportadas por la comunidad de desarrolladores, de codiglo libre y soporte multiplataforma en PC.

Con el objetivo de asegurar que lo primero que se desarrolle cumpla son los requeriminetos asociados con el control del Modulo Digital, luego proveer una manipulacion remota y finalmente una interfaz de usuario para el manejo a traves del navegador web, se organizo el desarrollo de la siguiente manera:

* Modulo Digital * Servidor * Interfaz Grafica Web

El mapeo entre los requerimentos y el orden propuesto genera el siguiente esquema de tareas a realizar.

* Modulo Digital - Configuracion del proyecto y entornos - Implementacion control AD - Implementacion control DDS2 - Implementacion control PP2 - Implementacion abstraccion Experimento - Implementacion de interfaz USB - Implementacion de un programa principal - Gestion de logs - Integracion con Servidor

* Servidor - Configuracion del proyecto y entornos - Servicios REST para la edicion de experimentos - Servicios REST para control de ejecucion de experimentos - Servicios REST para la generacion de reportes - Modelado de Base de datos - Configuracion de sesiones - Configuracion de acceso a los servicios REST - Gestion de logs - Integracion con Interfaz WEB - Integracion con Modulo Digital

* Interfaz WEB - Configuracion de proyecto y entornos - Implementacion de transiciones - Componente de edicion de experimentos - Componente de control de ejecucion de experimentos - Componente de login - Componente de visualizacion de errores - Componente de definicion de experimento - Implementacion de control de estados - Integracion con Servidor

18 Planificacion de Testing

La planificacion de testing del sistema tiene varios enfoques en respuesta a diferentes objetivos:

* brindar una experiencia predecible y agradable al usuario final * lograr una integracion coherente entre modulos * aumentar cobertura de requerimientos de usuario y sistema * facilitar la comprension del comportamiento esperado del sistema

Por unidad existen las siguientes validaciones:

Modulo Digital

* entradas de configuracion para el PP2, DDS2 y AD * entradas de programa ejecutable por el PP2 * existencia de conexion via interfaz USB durante la ejecucion de experimentos * salidas de señales RF * salidas de pulsos TTL * input de conversion de datos del AD * salida de reportes de los canales A y B.

Servidor

* autenticacion obligatoria en los servicios * ejecucion secuencial de los experimentos * aislamiento de espacio de trabajo de usuarios * coherencia en los estados de los experimentos * generacion de reportes * gestion de procesos de ejecucion de experimentos

Interfaz de usuario web

* visualizacion coherente de elementos y estilos * tiempos cortos de carga de las vistas * visualizacion de alertas de usuario * transiciones entre vistas

Dados los diferentes enfoques son necesarias se eligieron las siguientes para probar:

* Postman para simular comunicacion http con servicios REST * Arduino para visualizar los pulsos TTL via Serial Plotter * Osciloscopio para medir señales RF generadas en los experimentos. * Generador de señales para simular adquisicion de datos del AD. * Google Chrome y herramientas para validar estados de la interfaz grafica. * Mock del servidor escrito en Flask-Python para simular servicios REST.

19 Arquitectura del Modulo Digital

20 Arquitectura del Servidor

21 Arquitecturra de Interfaz Grafica

22 Integracion de los submodulos

La integracion de los modulos del sistema se implementa a diferentes niveles:

Repositorio

Un submodulo git es un repositorio dentro de otro que lo incluye, con su propio historial de cambios. Asi mismo cuando hay un cambio en un submodulo el contenedor lo contemplara siempre y cuando lo agregue al historial de este. El repositorio afectado por esta metodologia es el del Servidor incluyento al Modulo Digital y a la Interfaz Grafica.

El arbol de directorios del repositorio queda asi:

Codigo:

El servidor provee el ManagerThread para poder interactuar con el Modulo Digital y con el modelo de la base de datos segun el estado de la ejecucion del experimento en curso.

Para integrar con la Interfaz grafica lo hace de manera estandar via configuracion del servidor que provee seguridad y autenticacion de los servicios via CSRF-Token presente en el archivo principal index.html de la Interfax Grafica.

23 Analisis de Performance

Se propone como analisis de performace un experimento con las siguientes caracteristicas:

* 500 promedios * 2 loops con un pulso cada uno * bloque de 8K * 4 fases

Los consumos de CPU - RAM - DISCO

24 Casos de Prueba

De acuerdo con los enfoques generales y por unidad se diseñaron los siguientes casos de prueba

* experimento con configuracion invalida del AD * experimento con configuracion invalida del DDS2 * experimento con secuencia invalida para el PP2 * experimento un pulso largo * experimento un pulso corto * experimento con pulsos largos y cortos intercalados * experimento promedio con un pulso * experimento promedio con un pulso variable en duracion * experimento promedio con un pulso variable en fase * experimento un pulso dentro de un loop * experimento un pulso dentro de un loop de nivel 4 * experimento con un RETL sin LOOP asociado * experimento con un LOOP sin RETL asociado * experimento con un LOOP de nivel 5

* solicitud de servicio sin token de autenticacion * solicitud de autenticacion con usuario no registrado * solicitud de autenticacion con usuario registrado pero contraseña invalida * solicitud de fin de sesion * solicitud de creacion de un experimento * solicitud de edicion de un experimento * solicitud de edicion de un experimento inexistente * solicitud de edicion de un experimento en ejecucion * solicitud de eliminacion de un experimento * solicitud de eliminacion de un experimento inexistente * solicitud de eliminacion de un experimento en ejecucion * solicitud de ejecucion de un experimento habiendo uno en ejecucion * solicitud de ejecucion de un experimento inexistente * solicitud de ejecucion de un experimento * solicitud de cancelacion de una ejecucion en curso * solicitud de reporte durante la ejecucion * solicitud de reporte finalizada la ejecucion * solicitud de reporte de un experimento que nunca se ejecuto * solicitud de reporte de una ejecucion cancelada * solicitud de reporte de una ejecucion fallida

* ingreso al sitio con usuario y contraseña validos * ingreso al sitio con usuario ó contraseña invalidos * salida del sitio con boton login

* crear un experimento

* visualizacion de lista de experimentos creados * filtrado de lista de experimentos creados

* ver un experimento seleccionado de la lista * edicion de un experimento seleccionado de la lista * eliminacion de un experimento seleccionado de la lista * ejecucion de un experimento seleccionado de la lista

* visualizacion de lista de resultados * ver detalles de un resultado * visualizacion de menu de gestion de experimentos * visualizacion de menu de gestion de resultados * visualizacion de menu de navegacion * visualizacion de alertas

25 Errores conocidos

* si el sistema es interrumpido por un apagado imprevisto durante una ejecucion es posible que un estado sea inconcistente

* en vista edicion hacer click en boton "guardar y ejecutar" demora 5 segundos la transicion a vista de resultados

* las alertas deben ser cerradas para obtener nuevas si las hubiese

* soporte parcial a diferentes tamaños de pantalla

26 Limitaciones de la platforma

Existen limitaciones de diversos tipos:

Plataforma

* driver y dll para windows posiblemente reemplazable usando libusb * administracion de archivos limitada * soporte librerias de matematica y graficos limitada

Modulo Digital

* requiere la implementacion de un manager para su utilizacion * requiere permisos para creacion de archivos de logs y reportes

Server

* gestion del proceso del servidor es manual * la gestion de usuarios es manual via interfaz Django * seguimientos de recursos de sistea es manual

Interfaz Grafica

* programacion orientada a componentes * JSX una variacion de JavaScript * JQuery no es soportado

27 Mejoras a futuro

Modulo digital

* interfaces de servicio * independencia como programa de consola * test automaticos * integracion continua

Server

* cola de experimentos * soporte a otros Modulos Digitales u Aparatos * post-procesado de datos en tiempo real * test automaticos * integracion continua

Interfaz Grafica

* soporte a telefonos moviles * visualizacion de graficos * tests automaticos * vista de manipulacion de resultados * vista de gestion de perfil * tests automaticos * integracion continua

28 Analisis de esfuerzo

Niveles: * Muy Alto * Alto * Medio * Bajo * Trivial

Modulo Digital * Usb: Alto * PP2: Alto * AD: Alto * DDS2: Muy Alto * Reporte: Bajo * Hilo: Alto

Server: * Servicios: Alto * Login: Alto * Integracion: Muy Alto * Modelado: Medio * Manager Hilos: Alto

Interfaz Grafica: * Vistas: Alto * Integracion: Alto * Alertas: Alto