

SOLICITUDE DE APROBACIÓN DE ANTEPROXECTO DO TRABALLO DE FIN DE GRAO GRAO EN ENXENARÍA INFORMÁTICA.

Entréguese na Administración da ETSE antes da data indicada no Artigo 15 do Regulamento de traballo fin de grao enxeñaría informática: **O Anteproxecto terá que ser aprobado pola Comisión de Traballos Fin de Carreira de Enxeñaría Informática como mínimo tres meses antes da data de depósito e de solicitude de trámite de defensa do TFG**

Datos do/a Alumno/a			
Nome: Orquídea Manuela Seijas Salinas			DNI: 45957732Y
Enderezo: TSIA. DE LA IGLESIA, 3, P05.			Localidade: A ESTRADA
Provincia: PONTEVEDRA	C.P.: 36680	Teléfono: 677144767	Correo-e USC: orquideamanuela.seijas@rai.usc.es
Requisitos			
O alumnado poderá ter pendentes como máximo 75 créditos para completar os estudos, excluídos os correspondentes ao TFG			
Datos do Traballo de Fin de Grao			
Título: Emulador de una arquitectura ARMv8			
Titor(a): Juan Carlos Pichel Campos		Correo-e: juancarlos.pichel@usc.es	
Cotitor(a):		Correo-e:	
Cotitor(a):		Correo-e:	
Áreas de coñecemento: ATC			
Departamento: Electrónica y computación			
Empresa (se procede):			

A persoa que asina e cos datos que se indican solicita á Comisión de Traballos Fin de Grao de Enxeñaría Informática a aprobación do anteproxecto que se acompaña.

Santiago de Compostela, de de 20__

O/A alumno/a

Vº e Pr. O/A Titor(a) e Cotitor(a) do proxecto

Asdo:

Asdo:

Á atención da Comisión de Traballos Fin de Grao de Enxeñaría Informática

DESCRIPCIÓN DO ANTEPROXECTO

(Adaptar o tamaño do documento segundo sexa necesario.)

TÍTULO:

Emulador de una arquitectura ARMv8

Introdución

Para crear software eficaz y eficiente es necesario conocer, como mínimo, el funcionamiento básico de la arquitectura de un procesador. Este conocimiento es fundamental para la formación de un ingeniero informático, que necesita saber lo que sus acciones en un lenguaje de alto nivel provocan a nivel de una microarquitectura. En muchas ocasiones, el funcionamiento de los procesadores se explica a través de un procesador concreto, con el fin de facilitar el aprendizaje. Además, se utiliza como elemento de apoyo un emulador pedagógico[1]. Este debe permitir observar la evolución de la memoria y de los registros durante la ejecución de las instrucciones.

La tecnología de los procesadores ha evolucionado en los últimos años, por lo que surge la necesidad de proveer de un emulador para un procesador actual. Para este propósito se ha escogido el procesador ARMv8, ya que es la versión más moderna de esta familia de procesadores utilizada en un porcentaje muy importante en dispositivos móviles. Además, está previsto el aumento de su uso en los próximos años[2]. El diseño de este procesador se realizó siguiendo una serie de principios que permiten conseguir el delicado equilibrio entre el número de instrucciones necesaria para la ejecución de un programa, el número de ciclos de reloj necesarios por una instrucción y la velocidad del reloj[3]. Estos principios son:

- *Simplicity favors regularity*: muchas de las propiedades del conjunto de instrucciones del ARMv8 están motivadas por el fin de mantener las mismas características (mismo tamaño, mismos requisitos de registros para operaciones aritméticas, etc), lo que permite una arquitectura relativamente simple.
- *Smaller is faster*: ARMv8 tiene únicamente 32 registros con el fin de ser más rápido.
- *Good design demands good compromises*: un ejemplo de esto es el compromiso entre permitir el uso de direcciones y constantes en instrucciones de tamaño considerable manteniendo todas las instrucciones del mismo tamaño.

Así pues, tal y como se puede ver, el diseño de un emulador pedagógico asociado a esta arquitectura permitiría un aprendizaje asociado a una arquitectura más sencilla que otras (como la x86 de Intel), de tamaño reducido y con ejemplos claros de compromiso para conseguir un resultado lo suficientemente óptimo.

Obxectivos

El objetivo general de este Trabajo de Fin de Grado es el diseño y la implementación de un emulador de un subconjunto de instrucciones máquinas de un procesador con arquitectura ARMv8. Este objetivo general puede dividirse en los siguientes objetivos específicos:

- Ejecutar programas escritos en lenguaje ensamblador que contengan el subconjunto de instrucciones seleccionadas.
- Detectar errores en el código introducido en el emulador.
- Observar la evolución del contenido de la memoria y de los registros durante la ejecución de las instrucciones.
- Proporcionar una interfaz de usuario amigable que permita tener una experiencia satisfactoria al utilizar el emulador.
- Proporcionar un manual de usuario que permita conocer todas las funcionalidades del emulador.

Descripción técnica

A continuación, se describirán las fases técnicas por las que habrá que pasar con el fin de obtener una aplicación de escritorio que permita emular un subconjunto de instrucciones del procesador ARMv8:

1. **Gestión temporal del proyecto:** esta fase se llevará a cabo a lo largo del proyecto, ya que es necesario observar y gestionar el uso del tiempo: se llevará a cabo una planificación detallada del resto de fases y se irá siguiendo la misma a lo largo del proyecto.
2. **Análisis de requisitos:** para desarrollar correctamente el Trabajo de Fin de Grado será necesario realizar un análisis de otros emuladores de procesadores, como Simula3MS[4] o SIMUPROC[5], prestando especial atención a Simula3MS, ya que es el que se utiliza actualmente en el grado para fines educativos. También se estudiarán interfaces gráficas, con el fin de conseguir una experiencia de usuario agradable: interfaces gráficas asociadas a IDEs (*Integrated Development Enviroment*) y a aplicaciones de escritorio actuales. En este punto, se listarán los requisitos asociados al proyecto.
3. **Diseño:** esta fase se subdivide en dos claramente diferenciadas: diseño de interfaz de usuario y diseño del software a implementar. En primer lugar, se realizará el diseño de la interfaz gráfica de usuario utilizando Adobe XD. Cuando ya se cuente con esta información, se diseñará el software estudiando algunos patrones de diseño primero con el fin de conseguir un software bien estructurado. Para esto se utilizará Star UML.
4. **Implementación:** este emulador pedagógico será una aplicación de escritorio que será desarrollada principalmente con Kotlin y JavaFX en el IDE IntelliJIDEA. Se considerarán como dos fases de implementación diferentes la de la interfaz de usuario y la del analizador del subconjunto de instrucciones que permita emular el procesador ARMv8.
5. **Validación y pruebas:** se realizarán tests de usabilidad y tests sobre el código con el fin de comprobar su correcto funcionamiento.
6. **Documentación:** será necesario escribir un manual de usuario. Además, se documentará el código y el proceso con el fin de generar la memoria del proyecto.

Medios materiais necesarios.

Un ordenador con S.O Windows y Linux: Es necesario para realizar el diseño, la codificación y pruebas de la aplicación. En esta máquina se instalará el siguiente software:

- IntelliJ IDEA: Codificación y pruebas.
- Star UML: Diseño de software.
- Adobe XD: Diseño de interfaz gráfica de usuario.

Bibliografía.

- [1] Concheiro, R., Loureiro, M., Amor M., González. P. (2005). "Simula3MS: simulador pedagógico de un procesador". Dtpo. Electrónica y Sistemas. Universidade da Coruña. <http://bioinfo.uib.es/~joemiro/aenui/procJenui/Jen2005/cosimu.pdf> última consulta: 17/02/2018
- [2] Fitzpatrick, J. (2011). "An Interview with Steve Furber". Communications of the ACM. <https://cacm.acm.org/magazines/2011/5/107684-an-interview-with-steve-furber/fulltext> última consulta: 16/02/2018.
- [3] Patterson, D., Hennesy, J. "Computer Organization and Design: The hardware/software interface ARM edition", Morgan Kaufmann publishers. 2017.
- [4] Simula3MS. "Introducción" <http://simula3ms.des.udc.es> última consulta: 19/02/2018
- [5] Vlaye (2005) "¿Qué es SimuProc?". <https://sites.google.com/site/simuproc/> última consulta: 19/02/2018

Observacións.

No procede.

FASES DO TRABALLO E ESTIMACIÓN TEMPORAL

Un traballo de fin de grao suporá 401,25 horas de traballo autónomo do alumnado e 11,25 horas de traballo presencial (titorías e avaliación).

Dedicación semanal prevista (en horas/semana): 26

Estimación temporal (en semanas)	16
Gestión temporal	0.75
Análisis de requisitos	2
Diseño	3
Implementación	6
Pruebas	3
Documentación	1.25

(Para a xestión do alcance do proxecto deberase incluír en tódolos casos unha EDT segundo se recolle no PMBOK do PMI. Capítulo 5, Xestión do Alcance do Proxecto)

