



Testing de emulador de microprocesaro Leon3 para desarrollo de software satelital y simuladores

Pruebas de sistema

Autor:

Ing. Iriarte Fernandez, Nicolás Ezequiel (NicolasIriarte95@gmail.com)

Director:

Esp. Lic. Horro, Nicolás Eduardo (INVAP.S.E.)

Docentes:

Alejandro Permingeat; Esteban Volentini; Mariano Finochietto y Rafael Oliva.

*Este documento fue realizado en el curso Testing de Software en Sistemas Embebidos
el 11 de Abril de 2024, cuarto bimestre.*

Índice

Introducción	4
Diseño de casos de prueba	4

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento.	11 de Abril de 2024

Documentos anexos

Ref.	Nombre	Descripción
AD.01	NEMU-SRD-1	Especificación de requerimientos de software.
AD.02	NEMU-UCD-0	Defenición de casos de uso y arquitectura de software.
AD.03	NEMU-MTP-0	Master Test Plan.

Cuadro 1. Documentos anexos.

Introducción

En el presente documento se detallarán once ensayos a nivel sistema de aceptación del componente de software “CPU”, perteneciente al emulador de microprocesador Leon3. El objetivo de estos ensayos es verificar que el componente cumple con los requerimientos de software especificados en el documento **NEMU-SRD-1**.

Previo a dicho documento, se recomienda la lectura del documento “Master Test Plan”, anexado en la tabla de referencias del presente documento como **NEMU-MTP-0**.

Diseño de casos de prueba

A continuación se detallan los casos de prueba a nivel sistema que se realizarán sobre el componente de software “CPU”, respecto a ciertos requerimientos específicos que se complementan en conjunto para un caso de uso.

Requerimientos a testear:

- **NEMU-SR-01:** El software deberá proveer acceso a los registros del procesador emulado.
- **NEMU-SR-03:** El software deberá poder cargar los mismo binarios que el microprocesador físico.
- **NEMU-SR-06:** El software deberá emular las instrucciones de lectura y escritura de memoria.
- **NEMU-SR-07:** El software deberá emular las instrucciones de suma, resta y multiplicación.
- **NEMU-SR-08:** El software deberá emular las instrucciones de salto condicionales, tales como operadores *if* y *jumps*.

Para el desarrollo de los ensayos, se emplea la técnica de diseño de “Classification-Tree method” **CTM**, la cual se compone de los pasos:

1. Identificación de los aspectos bajo prueba.
2. División del dominio de entrada de acuerdo con los aspectos.
3. Especificación de los casos de prueba lógicos.

Mediante esta técnica, se logra una cobertura de los casos de prueba de manera eficiente y efectiva. Generando la cantidad mínima de casos de prueba necesarios para cubrir el dominio de entrada.

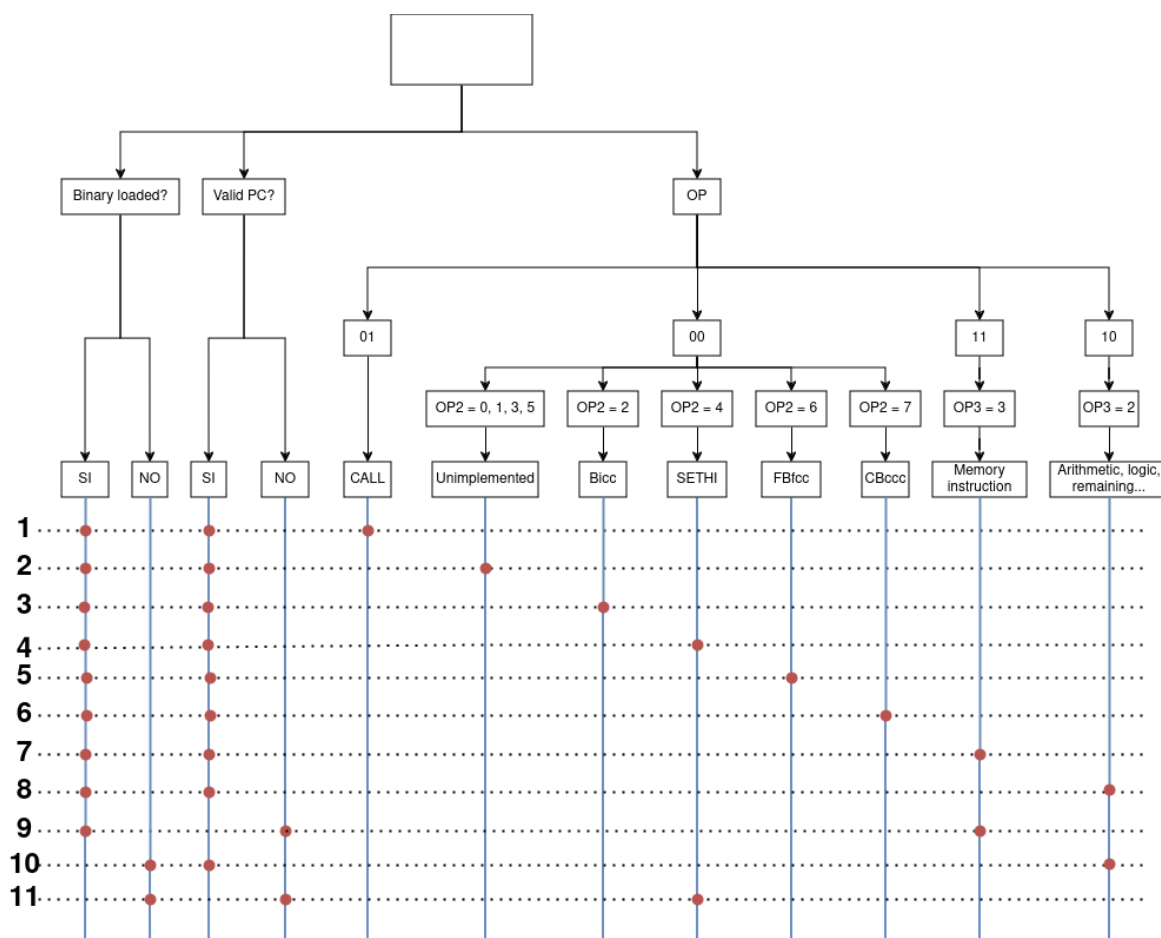


Figura 1. Diagrama CTM.

NOTA: Debido a la extensa cantidad de intrucciones disponibles para el procesador seleccionado, el arbol fué acotado y/o simplificado para toman unicamente las instrucciones más representativas y variadas, no siendo la totalidad de las instrucciones disponibles.

Dada la Figura ??, se procede a la generación de los casos de prueba lógicos, los cuales se detallan en la siguiente tabla:

CASO	ASPECTO	RESULTADO ESPERADO
1	Binario cargado PC valido OP: 01	Ejecución exitosa
2	Binario cargado PC valido OP: 00 OP2: 01	Error, instrucción invalida
3	Binario cargado PC valido OP: 00 OP2: 02	Ejecución de instrucción Bicc exitosa
4	Binario cargado PC valido OP: 00 OP2: 04	Ejecución de instrucción SETHI exitosa
5	Binario cargado PC valido OP: 00 OP2: 06	Ejecución de instrucción FBfcc exitosa
6	Binario cargado PC valido OP: 00 OP2: 07	Ejecución de instrucción CBccc exitosa

CASO	ASPECTO	RESULTADO ESPERADO
7	Binario cargado PC valido OP: 11 OP3: 03	Ejecución de instrucción exitosa
8	Binario cargado PC valido OP: 10 OP3: 02	Ejecución de instrucción exitosa
9	Binario cargado PC invalido OP: 11 OP3: 03	Error, PC invalido
10	Binario no cargado PC valido OP: 10 OP3: 02	Error, binario no cargado
11	Binario no cargado PC invalido OP: 00 OP2: 04	Error, binario no cargado