



XV Seminario de Invierno CAPAP-H

CREATOR: un entorno de desarrollo integrado para la docencia y la investigación en procesadores RISC-V

Diego Camarmas-Alonso
dcamarma@inf.uc3m.es

Cáceres, 29-31 de enero de 2025

Ayuda “Entorno de desarrollo integrado para la docencia y la investigación en procesadores RISC-V” (PDC2023-145832-I00)



GOBIERNO
DE ESPAÑA

MINISTERIO
DE CIENCIA, INNOVACIÓN
Y UNIVERSIDADES



Financiado por
la Unión Europea
NextGenerationEU



Plan de Recuperación,
Transformación y
Resiliencia



AGENCIA
ESTATAL DE
INVESTIGACIÓN

- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real*
- 3 Servicio de Laboratorio Remoto
- 4 *Work in Progress*
- 5 Estadísticas de CREATOR

- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real*
- 3 Servicio de Laboratorio Remoto
- 4 *Work in Progress*
- 5 Estadísticas de CREATOR

El origen de CREATOR

- La enseñanza de ensamblador tiene gran impacto
 - Entender aspectos de la arquitectura: *pipeline*, etc.
 - Entender aspectos del *software*: compilador, etc.
- Problemas de la enseñanza en ensamblador
 - Tiempo limitado en docencia
 - Enseñanza principalmente práctica (laboratorios y ejercicios)
- Para abordar estos desafíos
 - Simuladores intuitivos
 - Simuladores genéricos (RISC-V, MIPS, etc.)

CREATOR

- Simulador educativo multiplataforma
- Permite desarrollar programas ensamblador de forma interactiva
 - Compilación con detección de errores
 - Ayuda en línea de las instrucciones
- Permite simular el programa implementado
 - Depurar de forma intuitiva
 - Ver el estado de registros y memoria
 - Ayuda en el uso de pila
 - Comprobación del convenio de paso de parámetros
- Modificar y/o añadir la arquitectura y juego de instrucciones

CREATOR - Visualización de Arquitecturas

CREATOR 4.0 RISC-V (RV32IMFD)

didactic and generic assembly programming simulator

About us

Assembly

Simulator

Edit Architecture

Save Architecture

Configuration

Info

Architecture Info

Memory Layout

Register File

Instructions

Pseudoinstructions

Directives

Name	Co	Extended CO	Nwords	Instruction syntax	Properties	Clk Cycles	Fields	Definition
add	0110011	000000000	1	add rd rs1 rs2 add,INT-Reg,INT-Reg,INT-Reg		1	View Fields	rd = rs2 + rs1
addi	0010011	000	1	addi rd rs1 inm addi,INT-Reg,INT-Reg,inm-signed		1	View Fields	rd = rs1 + inm;
and	0110011	000000011	1	and rd rs1 rs2 and,INT-Reg,INT-Reg,INT-Reg		1	View Fields	rd = rs1 & rs2;
andi	0010011	111	1	andi rd rs1 inm andi,INT-Reg,INT-Reg,inm-signed		1	View Fields	rd = rs1 & capi_int2uint(inm);
auipc	0010111		1	auipc rd inm auipc,INT-Reg,offset_words		1	View Fields	rd = PC + (inm << 12);
				heli rs1 rs2 inm				if (rs1 == rs2)

CREATOR - Edición de Arquitecturas



The screenshot shows the CREATOR 4.0 RISC-V (RV32IMFD) interface. A dialog box titled "Edit Architecture" is open, displaying a JSON configuration for the architecture. The configuration includes fields for Name, Bits, Description, Data Format, Memory Alignment, and Main Function. The "Name" field is highlighted in red. The "Description" field contains the text: "RISC-V is an instruction set architecture (ISA) based on the RISC type and its hardware is free. This architecture was created in 201". The "Data Format" field is set to "big_endian" and the "Memory Alignment" field is set to "1". The "Main Function" field is set to "main". The dialog has "Cancel" and "Save" buttons at the bottom right.

```
1 {
2   "arch_conf": [
3     {
4       "name": "Name",
5       "value": "RISC-V (RV32IMFD)"
6     },
7     {
8       "name": "Bits",
9       "value": "32"
10    },
11    {
12      "name": "Description",
13      "value": "RISC-V is an instruction set architecture (ISA) based on the RISC type and its hardware is free. This architecture was created in 201"
14    },
15    {
16      "name": "Data Format",
17      "value": "big_endian"
18    },
19    {
20      "name": "Memory Alignment",
21      "value": "1"
22    },
23    {
24      "name": "Main Function",
25      "value": "main"
26    }
27  ]
28 }
```

¿Qué es CREATOR?

CREATOR - Edición de Programas

The screenshot shows the CREATOR RISC-V (RV32IMFD) IDE interface. The main window displays assembly code with the following content:

```

20
21 # loop header
22 loop1: beq t1, t2, end1 # if(t1 == t2) --> jump to fin1
23
24 # loop body
25 mul t6, t1, t4 # t1 * t4 -> t6
26 lw t6, 0(t5) # Memory[t5] -> t6
27 add s7, s7, t6 # t6 + s7 -> s7
28
29 # loop next...
30 add t1, t1, t3 # t1 + t3 -> t1
31 addi t5, t5, 4
32 beq s8, s8, loop1
33
34 # loop end
  
```

Numbered callouts highlight the following features:

- 1**: Points to the assembly code editor.
- 2**: Points to the context menu for the code editor, showing options like Copy, Cut, Paste, Select all, Undo, Redo, and Block code comment.
- 3**: Points to the File menu, specifically the "Get code as URI" option.
- 4**: Points to the Library menu, showing options like Create, Load Library, and Remove.
- 5**: Points to the Instruction Help window, which displays a list of RISC-V instructions such as add, addi, and and.

CREATOR - Ejecución de Programas

CREATOR 4.0 RISC-V (RV32IMFD)
 didaCic and generic_riscv_instsimulator

Architecture: Assembly | Reset | Inst. | Run | Flash | Stop | Examples | Calculator | Configuration | Info

1

Step	Address	Label	User Instruction	Loaded Instructions
	0x4		li t4 4	addi t4 x0 4
	0x8		la t5 w3	auipc t5 0xfff
	0xc			addi t5 t5 0xff4
	0x10		li s7 0	addi s7 x0 0
	0x14		li t1 0	addi t1 x0 0
	0x18		li t2 5	addi t2 x0 5
	0x1c	loop1	beq t1 t2 end1	beq t1 t2 6
	0x20		mul t6 t1 t4	mul t6 t1 t4
	0x24		lw t6 0 (t5)	lw t6 0 (t5)
STOP	0x28		add s7 s7 t6	add s7 s7 t6
	0x2c		add t1 t1 t3	add t1 t1 t3
	0x30		addi t5 t5 4	addi t5 t5 4
	0x34		beq x0 x0 loop1	beq x0 x0 -7

2

INT/Ctrl Registers | FP Registers | Memory | Stats | Energy (CLK Cycles)

Register value representation: Signed | Unsig. | IEEE 754 | **Hex**

Register name representation: Name | Alias | **All**

PC: 0000001c

zero x0 00000000	ra x1 ffffffff	sp x2 0000000c	gp x3 00000000
tp x4 00000000	t0 x5 00000000	t1 x6 00000001	t2 x7 00000005
fp x8 00000000	s1 x9 00000000	a0 x10 00000000	a1 x11 00000000
a2 x12 00000000	a3 x13 00000000	a4 x14 00000000	a5 x15 00000000
a6 x16 00000000	a7 x17 00000000	s2 x18 00000000	s3 x19 00000000
s4 x20 00000000	s5 x21 00000000	s6 x22 00000000	s7 x23 00000000
s8 x24 00000000	s9 x25 00000000	s10 x26 00000000	s11 x27 00000000
t3 x28 00000001	t4 x29 00000004	t5 x30 00000004	t6 x31 00000001

3

Clear | Enter

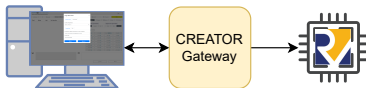
- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real***
- 3 Servicio de Laboratorio Remoto
- 4 *Work in Progress*
- 5 Estadísticas de CREATOR

CREATOR - *Hardware Real*

- Entornos con integración *hardware* existentes
 - Entornos de desarrollo profesionales
 - Demasiado complejos de usar para los estudiantes
 - Entornos de aprendizaje basados en Arduino o Python
 - Lenguajes de alto nivel en vez de ensamblador
- CREATOR como entorno de desarrollo intermedio
 - Integra lenguaje ensamblador y *hardware* real
 - Permite a los estudiantes ver las implicaciones de su código cuando ejecuta sobre *hardware* real

CREATOR - *Hardware Real*: Diseño

- Nuevo cuadro de diálogo en CREATOR
 - Envía el programa compilado en CREATOR al servicio web
- Nuevo servicio web (*gateway*)
 - Intermediario entre la interfaz de usuario y los *drivers* del dispositivo *hardware*



CREATOR - *Hardware Real*: Interfaz de Usuario

- Permite definir los parámetros necesarios para realizar las acciones **Flash** y **Monitor**
- Se conecta y comunica con el servicio web para cargar y ejecutar el programa en el dispositivo *hardware* desde CREATOR

Target Board Flash

(1) Select Target Board:
ESP32-C3 (RISC-V)

Prerequisites Run

(2) Target Port: (please verify the port on your computer)
COM1

(3) Flash URL:
http://localhost:8080

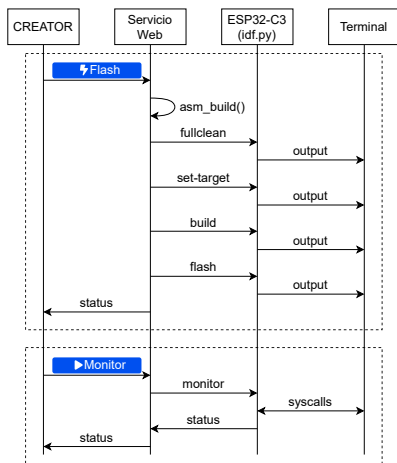
Flash Monitor

To stop the program execution press ctrl +] in the terminal

CREATOR - *Hardware Real*: Servicio Web

- Permite ejecutar sobre el dispositivo *hardware* las acciones
 - Flash
 - Monitor

- Imagen Docker disponible en DockerHub
 - Servicio web
 - *Drivers*

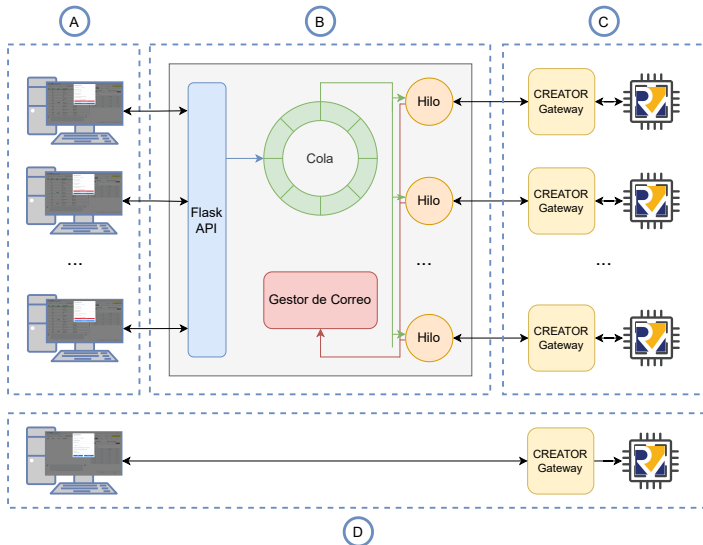


- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real*
- 3 Servicio de Laboratorio Remoto**
- 4 *Work in Progress*
- 5 Estadísticas de CREATOR

CREATOR - Servicio de Laboratorio Remoto

- El uso de CREATOR con *hardware* real puede ser complejo
 - Los estudiantes tienen que adquirir el dispositivo
 - Coste de tiempo y monetario
 - Los estudiantes tienen que configurar el entorno de trabajo
 - Disminuye el tiempo de aprendizaje de ensamblador
- Nuevo servicio de laboratorio remoto en CREATOR
 - Cargar y ejecutar los programas ensamblador de forma remota sobre *hardware* real
 - Dispositivos *hardware* proporcionados por el profesorado
 - Focalizar el tiempo en el aprendizaje de ensamblador
 - Proporcionar un entorno funcional desde el primer día

CREATOR - Laboratorio Remoto: Diseño



CREATOR - Laboratorio Remoto: Interfaz de Usuario

Envío

Target Board Flash

Remote Device **Local Device**

(1) Remote Device URL:

(2) Select Target Board:

(3) E-mail to receive the execution results:

Ejecución y Envío

Target Board Flash

Remote Device **Local Device**

(1) Remote Device URL:

(2) Select Target Board:

(3) E-mail to receive the execution results:

Last program status: **Queue position: 1**

CREATOR - Laboratorio Remoto: Gestor de Correo

- Envía la salida de la ejecución en el *hardware* real por correo electrónico
- Ventajas:
 - Permite enviar varios programas simultáneos
 - No es necesario esperar a terminar la ejecución del programa remoto

[CREATOR] Remote device results

Externo > Recibidos x



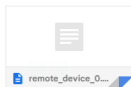
creator.arcos.inf.uc3m.es@gm... 11:41 (hace 2 minutos) ☆ ↶ ⋮
para mí ▾

Remote device ID=0 has been successfully completed, the execution results are attached.

Sincerely,
CREATOR Team

<https://creatorsim.github.io/>

1 archivo adjunto • Analizado por Gmail

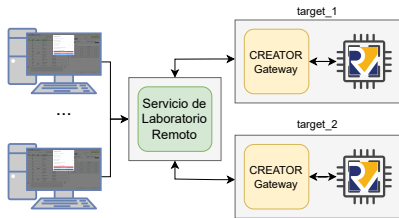


↶ Responder

↷ Reenviar

CREATOR - Laboratorio Remoto: Despliegue

- Servicios a desplegar
 - Servicio de laboratorio remoto
 - *Gateway* y microcontroladores
- Permite despliegue descentralizado
- Imágenes Docker de los servicios disponibles en DockerHub



- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real*
- 3 Servicio de Laboratorio Remoto
- 4 ***Work in Progress***
- 5 Estadísticas de CREATOR

¿Qué nos espera en CREATOR?

- Nuevo compilador de ensamblador
 - Mejorar los mensajes de error
 - Permitir particularidades de otros lenguajes ensamblador
- Interrupciones y ejecución en modo privilegiado
 - Hacer visibles aspectos del sistema operativo
- Nuevas instrucciones vectoriales y soporte para 64 bits
 - Uso de RISC-V en las GPU y las NPU para IA
 - La mayoría de los procesadores actuales son de 64 bits

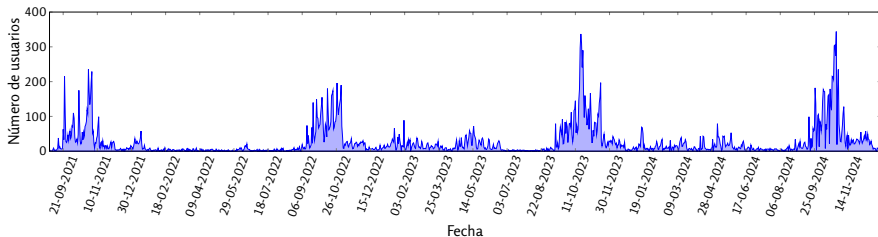
¿Qué nos espera en CREATOR?

- Integración con Sail
 - Lenguaje de definición de diferentes ISA
 - La mayoría de lenguajes ensamblador están definidos en Sail
 - Utilizar el motor de ejecución de Sail en CREATOR
- Soporte para la integración de diferentes placas y microcontroladores RISC-V
 - Poder depurar el programa ejecutado en el dispositivo remotamente desde CREATOR
 - Simular en CREATOR los diferentes GPIO del dispositivo
 - Soporte de la API de Arduino

- 1 ¿Qué es CREATOR?
- 2 Integración con *Hardware Real*
- 3 Servicio de Laboratorio Remoto
- 4 *Work in Progress*
- 5 Estadísticas de CREATOR**

Estadísticas de CREATOR (2021 - 2024)

- Estadísticas anónimas con Google Analytics
- La arquitectura más usada actualmente es RISC-V
- Aumento de usuarios año a año



¿En qué Universidades se utiliza CREATOR?



¿Queréis probar CREATOR?

- Simulador: <https://creatorsim.github.io/creator>
- Documentación: <https://creatorsim.github.io>
- DockerHub: <https://hub.docker.com/r/creatorsim>
- Código fuente: <https://github.com/creatorsim>

CREATOR



CREATOR - URI





XV Seminario de Invierno CAPAP-H

CREATOR: un entorno de desarrollo integrado para la docencia y la investigación en procesadores RISC-V

Diego Camarmas-Alonso
dcamarma@inf.uc3m.es

Cáceres, 29-31 de enero de 2025

Ayuda “Entorno de desarrollo integrado para la docencia y la investigación en procesadores RISC-V” (PDC2023-145832-I00)

