



<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/finalde proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024

**Departamento:  
Ingeniería e Investigaciones tecnológicas**

**Programa de acreditación:  
CyTMA2**

**Código del Proyecto:  
C2-ING-115**

**Título del proyecto**

**Soporte del estándar C de RISC-V y estudio de fusión de macro  
operaciones**

**Informe de Final**

**Director:  
Lic. Carlos Eduardo Maidana**

El presente documento se propone relevar las actividades acontecidas a lo largo del desarrollo del proyecto de investigación, con especial foco en las transferencias producidas a los efectos de difundir los resultados alcanzados. Esto se enmarca en las Políticas de la Secretaría de CyT UNLaM, bajo el lema de que el conocimiento científico es conocimiento comunicado. En la práctica científica habitual, este es transferido mediante distintos tipos de producciones: publicaciones en eventos científicos, libros, capítulos de libros, entre otras, destacándose particularmente el Artículo Científico/paper.



<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/finalde proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLAM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024

## A. Resumen del proyecto<sup>1</sup> (Desarrolle en no más de dos páginas.)

Dimensiones mínimas:

- *Problema de investigación:*

RISC-V fue concebida como una arquitectura abierta, de fines educativos y de investigación, pero rápidamente está siendo adoptada por fabricantes de microprocesadores en productos comerciales y se prevé que su uso a futuro sea aún más notorio. Dentro de todos los factores posibles a los cuales le podemos atribuir este suceso, probablemente el más destacable sea el hecho de ser una arquitectura libre y abierta sin necesidad de pagar cánones (royalties) por el uso de la misma. Compañías tales como Alphabet, Microchip, Alibaba y Huawei, entre otras utilizan o están interesadas en esta arquitectura sumado al hecho de que la misma no deja de ser utilizada en el ámbito académico. Al tener estas cualidades, RISC-V ofrece la posibilidad de implementar hardware a medida siempre respetando el estándar, siendo esto muy atractivo para las empresas desarrolladoras de microprocesadores y para cualquiera que desee hacer uso de la misma. A partir de lo anteriormente mencionado la eficiencia en la implementación de RISC-V junto con el uso optimizado de la memoria dependen de los fabricantes y al comenzar a ser una arquitectura mucho más extendida fuera del ámbito académico estas cualidades comienzan a tener cada vez mayor preponderancia.

Uno de los puntos a analizar que tiene un impacto directo en la eficiencia y en la memoria es el tamaño de la palabra. Las palabras en RISC-V varían dependiendo de la versión implementada. Si la versión es RV32 los registros y el espacio de direcciones es de 32 bits, en cambio si la versión es RV64 estos cambian a 64 bits. Lo dicho anteriormente no quiere decir que el tamaño de las instrucciones esté circunscritos solamente a esas dos longitudes fijas, debido que para evitar las posibles limitaciones que esto provocaría, RISC-V permite extender las mismas en segmentos de 16 bits admitiendo así instrucciones de 32, 48 y 64 bits entre otras.

Sin embargo, al usar RISC-V y desarrollar software para esta arquitectura, se demuestra de manera empírica que entre el 50 y el 60% de las instrucciones podrían ser codificadas solamente con 16 bits, lo que resultaría en un ahorro considerable de entre el 20 y el 30% del tamaño del código, *comprimiendo* así, de alguna forma, las instrucciones. Con esto en mente se crea entonces para RISC-V el estándar C (RVC) que permite codificar un selecto grupo de instrucciones en 16 bits, pudiendo hacer uso de este estándar junto con instrucciones de otro tamaño. Para lograr esto en el caso de RV32 las instrucciones con 32 bits poseerán siempre los dos bits menos significativos en "11", mientras que todas las instrucciones de 16 bits tendrán cualquier otro valor en dichos bits.

---

<sup>1</sup>Actualizar todos los ítems en el *Banco de datos de actividades de CyT del SIGEVA UNLAM* (del Director y de los integrantes del Proyecto), en especial "Antecedentes y Producciones y Servicios". Ver:  [www.youtube.com/@cytunlam](https://www.youtube.com/@cytunlam) 



<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/finalde proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024

No menos cierto es que existen otros sets de instrucciones como x86, los cuales codifican las instrucciones con tamaños variables, permitiendo así representarlas con menor cantidad de bits y en última instancia comprimir un programa. El problema radica en que, por razones históricas, en esta arquitectura, las instrucciones de 8 y 16 bits ya han sido utilizadas y debido a que se debe mantener la compatibilidad retroactiva, las mismas no pueden emplearse para instrucciones más modernas. Si bien cabría la posibilidad de que aquellas palabras de 16 bits, que todavía no han sido utilizadas del set x86 o x64, puedan utilizarse para comprimir instrucciones de mayor tamaño, esto implicaría un diseño más complejo del decodificador de instrucciones, debido a que se debería adaptarlo para poder soportar estas nuevas versiones comprimidas de instrucciones ya existentes.

Otras arquitecturas como ARM utilizan un modo que permite representar instrucciones en 16 bits (modo Thumb). Dichas instrucciones son versiones comprimidas de sus representaciones en 32 bits, las cuales el decodificador identifica y convierte utilizando lógica combinatorial a su equivalente en 32 bits. Esto cumple con el objetivo de reducir el tamaño del código, pero como contrapartida el requisito es forzar al procesador a utilizar el modo Thumb. A tal fin existe un bit de control (bit `T`) que le indica al procesador si se encuentra trabajando en modo Thumb o en modo normal. Dicho bit puede cambiar de estado en cualquier momento y si se desea ejecutar un segmento del código que utiliza instrucciones de 16 bits, se debe alterar el estado del bit `T`, ejecutar las instrucciones en modo Thumb y luego modificar nuevamente el bit `T` para trabajar otra vez en modo normal. Esto lleva a algo de sobrecarga: para lograr una eficiencia por lo menos igual que al ejecutar una instrucción de 32 bits, se requiere que al menos 2 instrucciones consecutivas puedan ser traducidas de 32 bits a 16 bits.

En el caso de RISC-V, siendo esta una arquitectura más moderna que las anteriores, y por lo tanto ya habiendo ganado experiencia, se optó por tomar lo mejor de cada caso. Las instrucciones comprimidas en RISC-V se traducen a instrucciones originales de forma transparente y combinatorial, sin complejizar el diseño del decodificador de instrucciones. Esto permite anexar de manera elegante y eficiente un nuevo estándar a un procesador ya diseñado. A diferencia de ARM, no se utiliza un bit especial para forzar la arquitectura en modo comprimido resolviéndose esto del lado de la microarquitectura, detectando si la instrucción a ejecutar se encuentra comprimida o no en tiempo de ejecución.

En el Anexo III se detalla de una manera minuciosa como se diseñó e implementó en nuestro procesador RISC-Vp esta extensión del set de instrucciones, obteniendo de esta forma un doble objetivo, dotar a la universidad de un IP Core de un procesador y obtener documentación y experiencia para incorporar contenidos a las materias: Arquitectura de Computadoras de la carrera de ingeniería en informática y Lenguajes Descriptivos de Hardware de la carrera de Ingeniería en Electrónica.

- *Metodología:*



<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/finalde proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024

Considerando que esta investigación se plantea como continuación de dos proyectos previos en los cuales se obtuvo un núcleo RISC-V, se plantearon las siguientes etapas para lograr los objetivos del proyecto:

- Estudiar las distintas extensiones de RISC-V con particular interés en la extensión C de instrucciones comprimidas.
- Estudiar el funcionamiento del toolchain GNU para RISC-V y los distintos ABIs disponibles para generar código compatible con RV32IC
- Estudiar el funcionamiento del depurador GDB en su versión cliente/servidor con el fin de utilizarlo en kits de desarrollo RISC-V
- Diseñar un módulo que soporte instrucciones del tipo C en el procesador RISC-Vp.
- Realizar ensayos sobre la lógica programable de código generado por el toolchain para ASM RISC-V soportando instrucciones del tipo comprimidas
- Estudiar la técnica de fusión de instrucciones comprimidas
- Encontrar situaciones donde la fusión de instrucciones comprimidas brinda mejor performance y plantear cambios en la microarquitectura para aprovechar esta mejora en la eficiencia.
- *Grado de ejecución de los objetivos planteados:*  
Los objetivos se cumplieron en un 100%. En el Anexo III se detalla un informe completo de los trabajos realizados.
- *Resultados:*  
Desde el punto de vista técnico los resultados fueron satisfactorios ya que se pudieron implementar los objetivos planteados desde el inicio del proyecto.  
En cuanto a la difusión de los resultados, debemos tener en cuenta que muchas veces la fecha de finalización con éxito de un proyecto no coincide con las fechas en las que los congresos del área se desarrollan, con lo cual, la divulgación de estos resultados se preveen para los congresos del año 2025.  
En cuanto a la incorporación de este contenido a las materias del área, se estima para el segundo cuatrimestre del año 2025.  
Al cierre de este proyecto se está redactando un paper para publicar en la revista de divulgación científica Reedi del Departamento de Ingeniería e Investigaciones Tecnológicas de la UNLaM.



<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/finalde proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024

**B. Informar cada producción con filiación UNLaM que derive de la presente investigación (artículo de revista/papers, libro, parte de libro, trabajos en eventos publicados/ponencia, etc.).**

**Anexar los textos de las producciones en SIGEVA UNLAM.<sup>2</sup>**

Tipo de Producción	Artículos publicados en revistas
Título	
Autor/es	
Editorial	
Fecha	
Situación	Elija un elemento.
DOI y/o Enlace/link (solo si está publicado)	

**C. Vinculación<sup>3</sup>:** Indicar conformación de redes, intercambio científico con actores externos, con otros grupos de investigación; desarrollos; con el ámbito productivo o con entidades públicas, etc. Desarrolle en una página.

**D. Otra información. Incluir toda otra información que el Director considere pertinente.**

**E. Cuerpo de anexos:**

- Anexo I:
  - FPI-013: Evaluación de alumnos integrantes. (de corresponder)
  - FPI-014: Comprobante de liquidación y rendición de viáticos. (de corresponder)
  - FPI-015: Rendición de gastos del proyecto de investigación.
  - FPI-038: Formulario de reasignación de fondos en Presupuesto. (de corresponder)
- Anexo II: FPI 017<sup>4</sup> Alta patrimonial de los bienes adquiridos con presupuesto del proyecto.
  - Disposición del Decano y nota de elevación del Director del Proyecto justificando “alta y/o” baja de cada integrante del equipo de investigación.

Firma y aclaración Director Proyecto  
Lic. Carlos Eduardo Maidana

Fecha: 28/02/2025

<sup>2</sup>Los archivos deberán estar en formato PDF, a texto completo. Podrán ser publicados en el Repositorio Digital UNLaM, bajo Licencias Creative Commons. Será evaluada la inclusión en el Repositorio aquellas publicaciones que poseen una licencia diferente o declaración de copyright.

<sup>3</sup>Entendemos por acciones de “vinculación” aquellas que tienen por objetivo dar respuesta a problemas, generando la creación de productos o servicios innovadores en articulación con el entramado socioproductivo.

<sup>4</sup> Solo ante la presentación del Informe Final.