

# Desarrollo de una herramienta de High-Level Synthesis basada en la síntesis de assembler RISC-V

Pereira Nuñez Machado, Martín

Universidad Tecnológica Nacional - Facultad Regional Buenos Aires, Digital Processing Lab

Docentes: Ing. Roberto Simone, Dr. Ing. Luciano Ferreyro, Dr. Ing Matías Hampel, Ing. Alejandra Gutierrez, Ing. Basilio Robino, Ing. Alejandro Furfaro

## Objetivos

Desarrollar una herramienta de HLS<sup>[1]</sup> que permita optimizar código escrito en C/C++ para ejecutarse en una arquitectura RISC-V sobre dispositivos System-on-Chip (SoC), integrando de esta manera un procesador RISC-V y una FPGA. El software posee rutinas que sintetizan un diseño en Verilog a Register Transfer Level (RTL), y se puede dividir en dos secciones principales (ver Figura 1):

- 1) la compilación de la porción de código que se ejecutará en el procesador,
- 2) la síntesis de la porción de código que se mapeará a hardware y deberá accederse desde una dirección de memoria del procesador.

## Marco Teórico

La herramienta explora el potencial de la síntesis de código ensamblador en una arquitectura RISC<sup>[2]</sup>.

Con este enfoque, se aprovechan optimizaciones del compilador. La estructura de una función de código ensamblador RISC simplifica las consideraciones en el desarrollo y mantenimiento de la herramienta. Existen precedentes únicamente de herramientas que utilizan representaciones intermedias de los compiladores.

RISC-V es una arquitectura de procesadores que poco a poco va a tomando mayor protagonismo, y aún no existen herramientas para futuros SoC's (como los de AMD o Intel, que poseen un procesador y una FPGA en el mismo chip) que integren un procesador de dicha arquitectura.

Para tal fin, es importante definir un mecanismo de comunicación entre el procesador RISC-V y la FPGA.

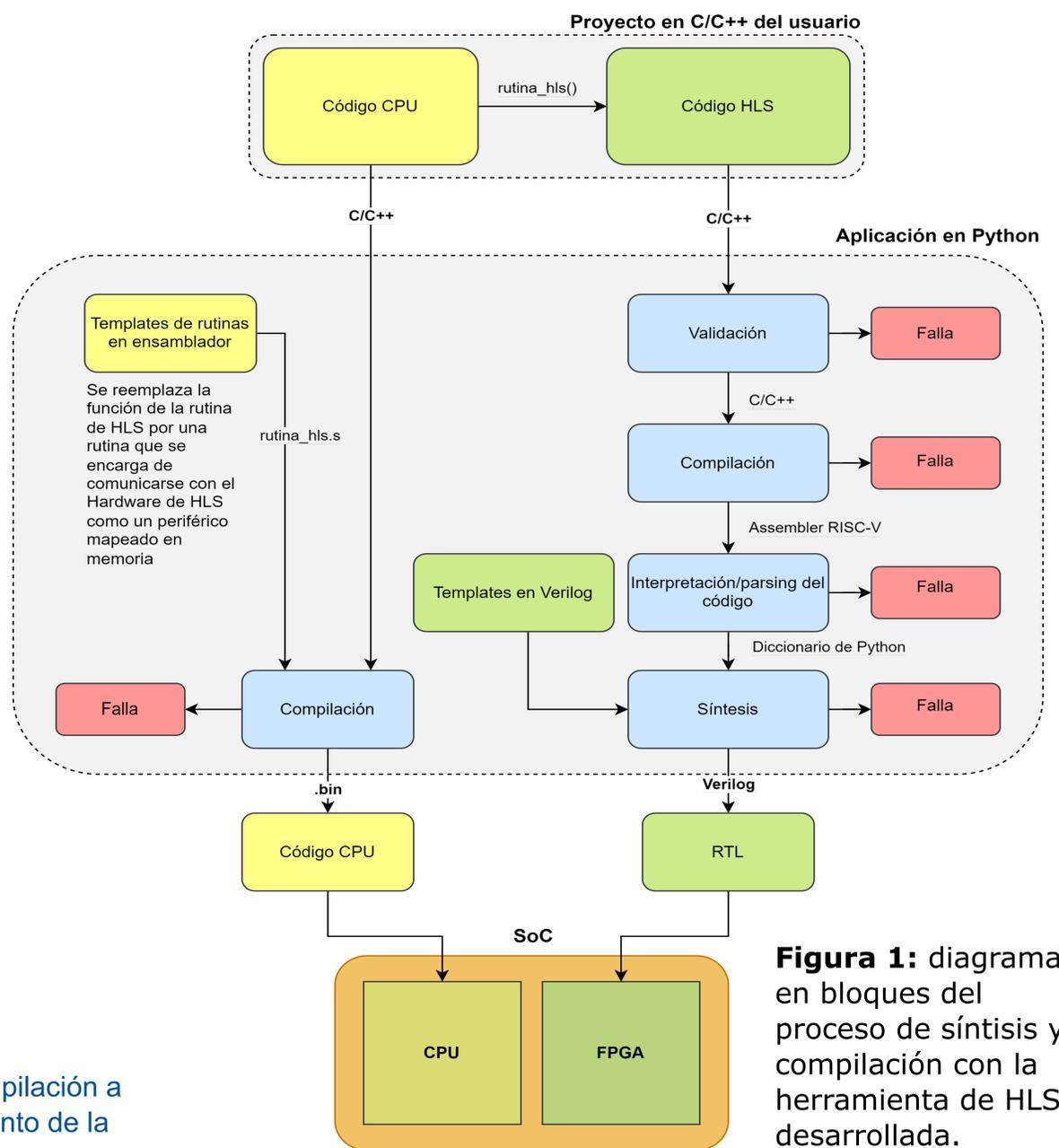
Este trabajo propone además de una herramienta HLS para este procesador, un estándar basado en un mapeo en memoria y una unidad de control (ver Figura 2).

## Resultados y logros

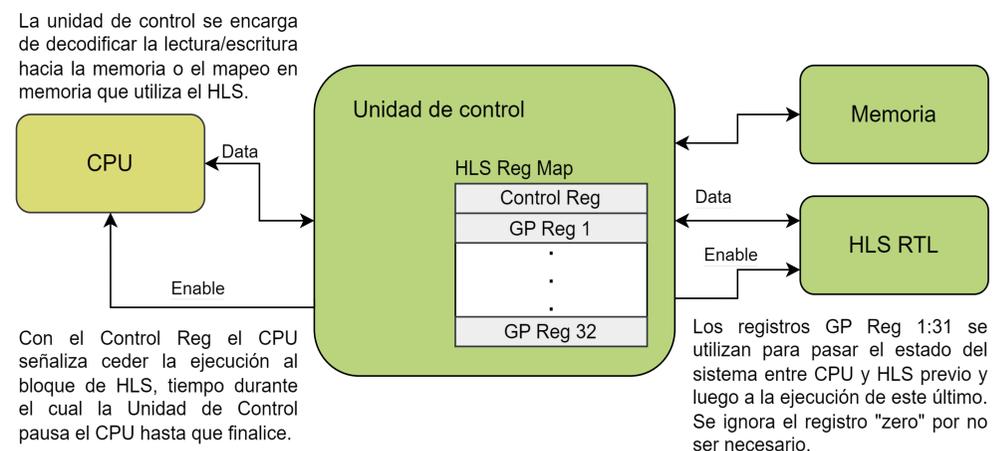
Se verificó que el paso intermedio de realizar una compilación a código ensamblador facilita el desarrollo y mantenimiento de la aplicación.

A fin de establecer una métrica de desempeño, se realizó una comparación de tiempos de ejecución comparando una simulación del firmware obtenido con nuestra herramienta de HLS, y una estimación de uso de procesador (ver tabla siguiente). Para el CPU, se considera el mejor caso para el cual se establece un ciclo de reloj por cada instrucción.

Iteraciones	CPU	HLS
N	$(21 + 28 * N)$ clks	$(4 + 3 * N)$ clks
10	301 clks	34 clks
100	28021	3004



**Figura 1:** diagrama en bloques del proceso de síntesis y compilación con la herramienta de HLS desarrollada.



**Figura 2:** diagrama del módulo RTL desarrollado para resolver la comunicación procesador RISC-V con FPGA.

Contacto: Proyecto Final - UTN-FRBA <https://www.frba.utn.edu.ar/electronica/proyecto-final/>