# Diseño y simulación de una microarquitectura cuántica

* **Introducción**

Que es la computación cuántica. Porque se espera que supere a los sistemas actuales. Porque diseñar una microarquitectura cuántica. Estado actual de desarrollo de la computación cuántica.

* **Objetivos**

Simular un microprocesador de forma flexible, capaz de integrar unidades de procesamiento cuántico, de forma que podamos experimentar en el código ensamblador de algún procesador didáctico como el DLX las posibilidades que la computación cuántica ofrece y adaptar los algoritmos cuánticos actuales a este lenguaje más cercano al informático que al físico.

El sistema debe poder simular la frágil estabilidad de los estados a nivel cuántico, modificándolos si no se opera con ellos suficientemente rápido, lo que nos permitirá sacar conclusiones sobre la velocidad requerida a la máquina y adaptar los algoritmos para compensar estos errores.

* **Física y computación cuánticas**

Un resumen suficientemente extenso sobre física cuántica que permita entender sin dificultad los conceptos que siguen. Nociones necesarias sobre computación cuántica, reglas básicas y formas típicas de representar y diseñar los algoritmos de esta naturaleza.

* **Microarquitectura clásica paralela**

Que es el procesamiento encadenado. Planificación dinámica: el algoritmo de Tomasulo. Resumen de las características del procesador clásico donde integraremos el cuántico como una *unidad funcional.*

* **Diseño y simulación**

Diseño del simulador del DLX en Java. Diseño del simulador de la unidad funcional cuántica. Estudio y modificación del repertorio de instrucciones del DLX para integrar la nueva unidad funcional.

Estudio sobre los más importantes algoritmos cuánticos actuales: los algoritmos de Deutsch, Shor y Grover, en ensamblador DLX. Velocidad necesaria del procesador para una ejecución aceptable.