

# Técnicas de Alto Rendimiento en el Diseño de Procesadores (Doctorado 2004/2005)

**“Flags de Optimización con GCC y su repercusión en el rendimiento de la CPU”**



**José Luis Vázquez Poletti**

[jl vazquez@fdi.ucm.es](mailto:jl vazquez@fdi.ucm.es)

<http://asds.dacya.ucm.es/jlvazquez/>

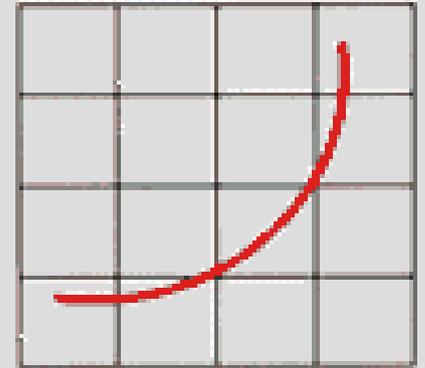
# Índice

- Introducción
- Resultados y Conclusiones



# Introducción

- Sin optimizaciones:
  - Reducir coste compilación
  - Depurado correcto
- Con optimizaciones:
  - Mejorar rendimiento / tamaño
- Benchmarks de SPEC2000



**spec**

- “Standard Performance Evaluation Corporation”
- Benchmark: Versiones reducidas y particulares de programas comunmente usados

# Introducción

- Benchmarks elegidos:

- VPR

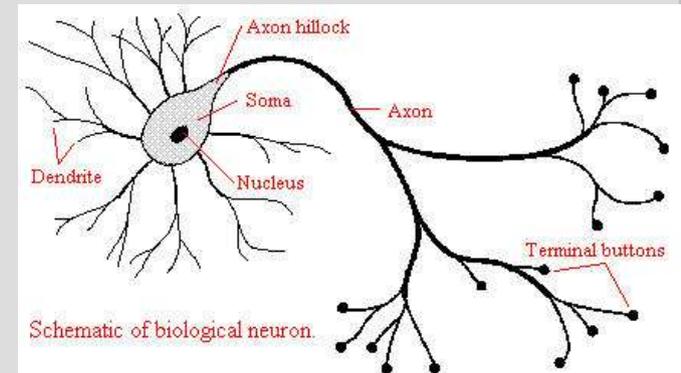
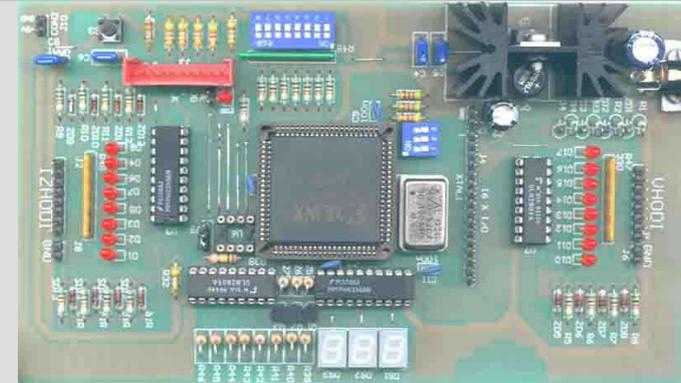
- Entero

- Ubicación y conexión de circuitos FPGA

- ART

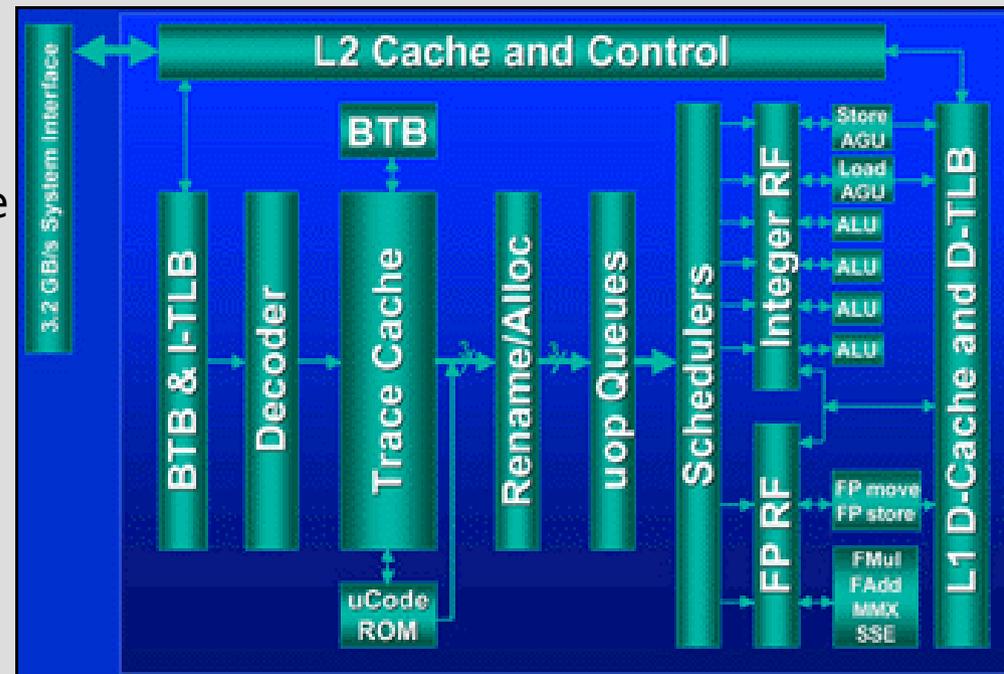
- Punto flotante

- Redes Neuronales



# Introducción

- Entorno de pruebas
  - Equipo: ACER Aspire 1661WLM
  - Características técnicas:
    - Procesador – Intel Pentium 4 (2,8 Ghz)
    - L1 I Cache (Trace Cache) – 12 K  $\mu$ ops
    - L1 D Cache – 8 K
    - L2 Cache – 512 K
  - Sistema Operativo: Mandrake Linux
    - Kernel: 2.6.8.1-12mdkenterprise



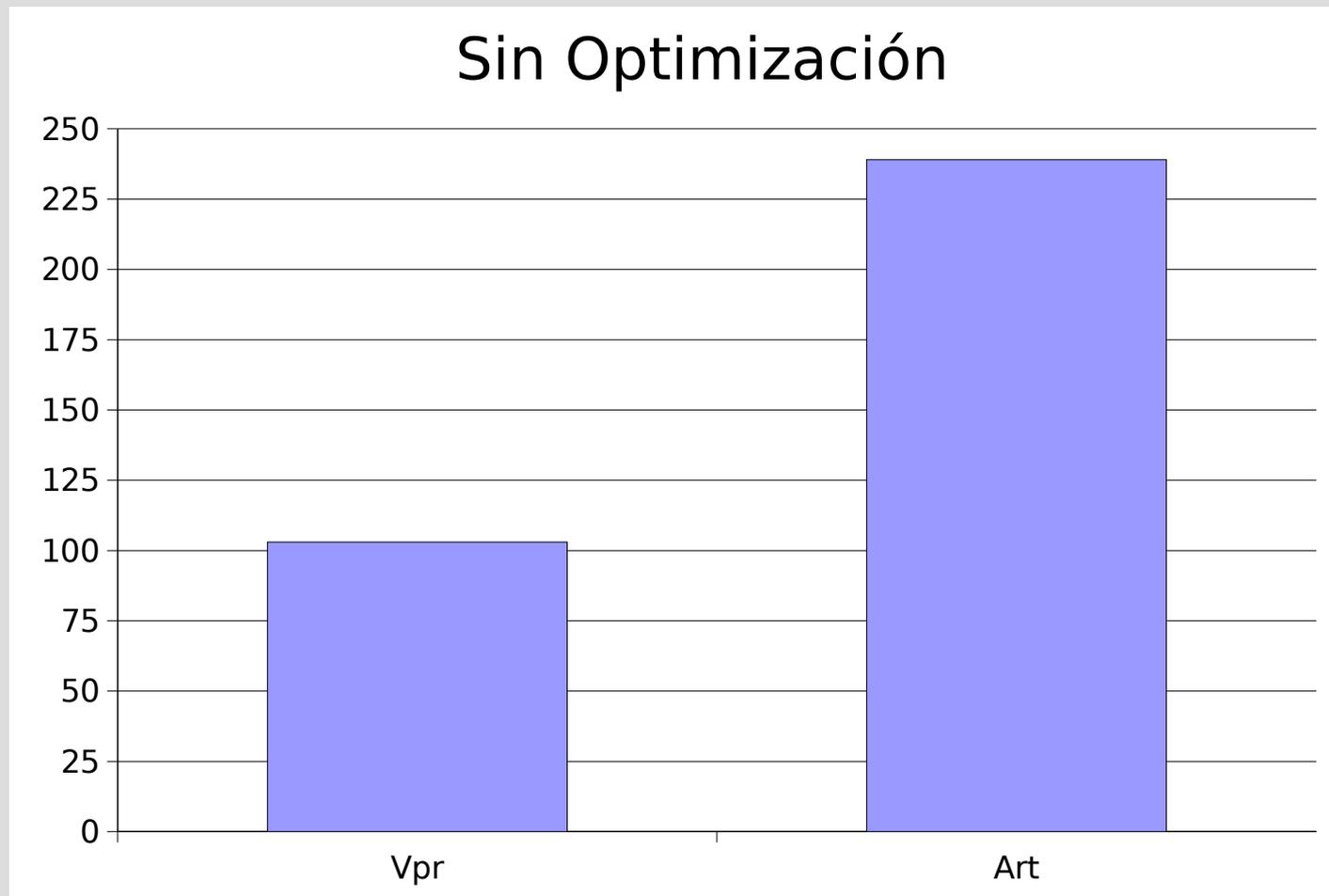
# Introducción

- GCC
  - “GNU Compiler Collection”
    - Antes sólo para C
  - Primera versión por Richard Stallman
  - Se apoya en “binutils”
    - Operar con archivos objeto
  - Arquitectura dividida en “frontend” y “backend”
  - Soporte para muchas arquitecturas y Sistemas Operativos



# Resultados y Conclusiones

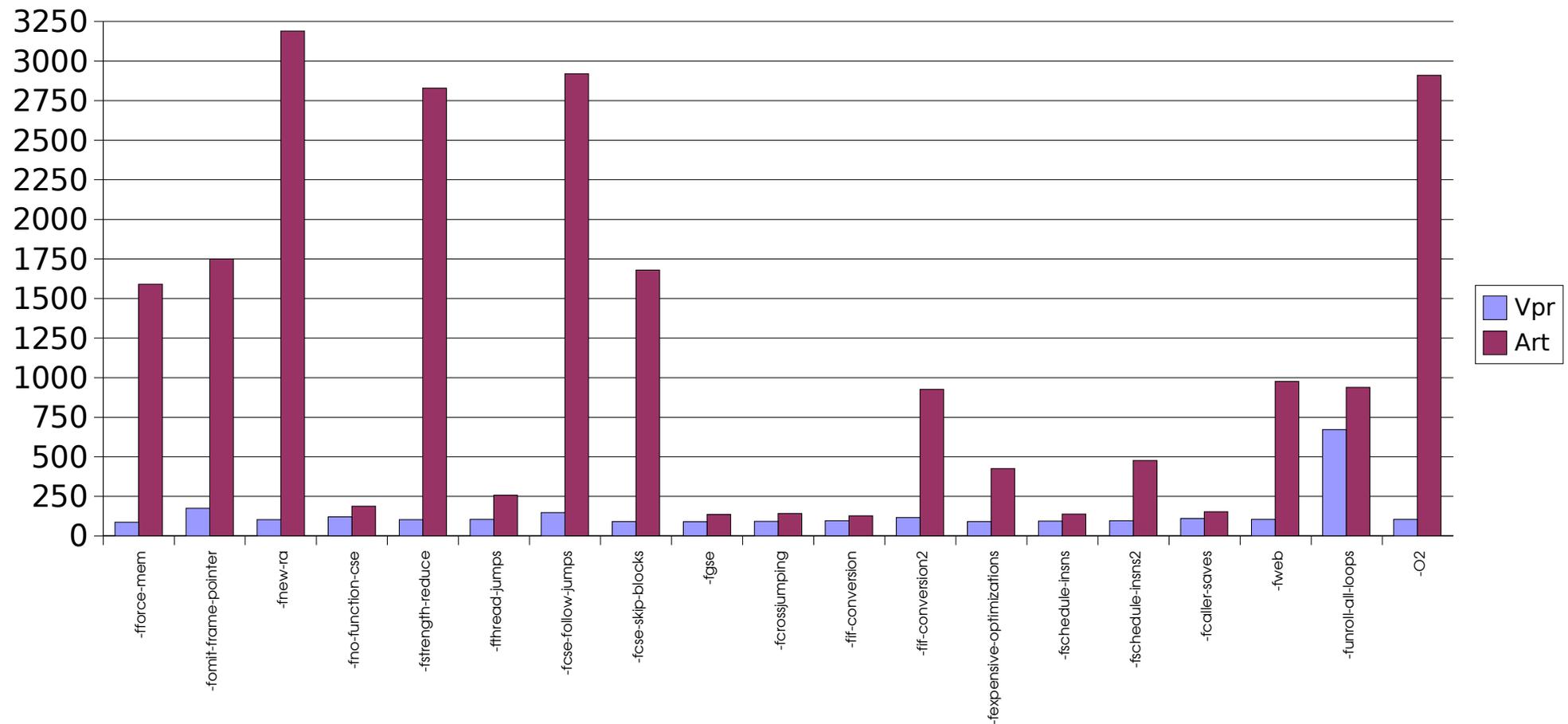
- Sin Optimización



# Resultados y Conclusiones

- Mejores Optimizaciones para Entero (VPR)

## Entero



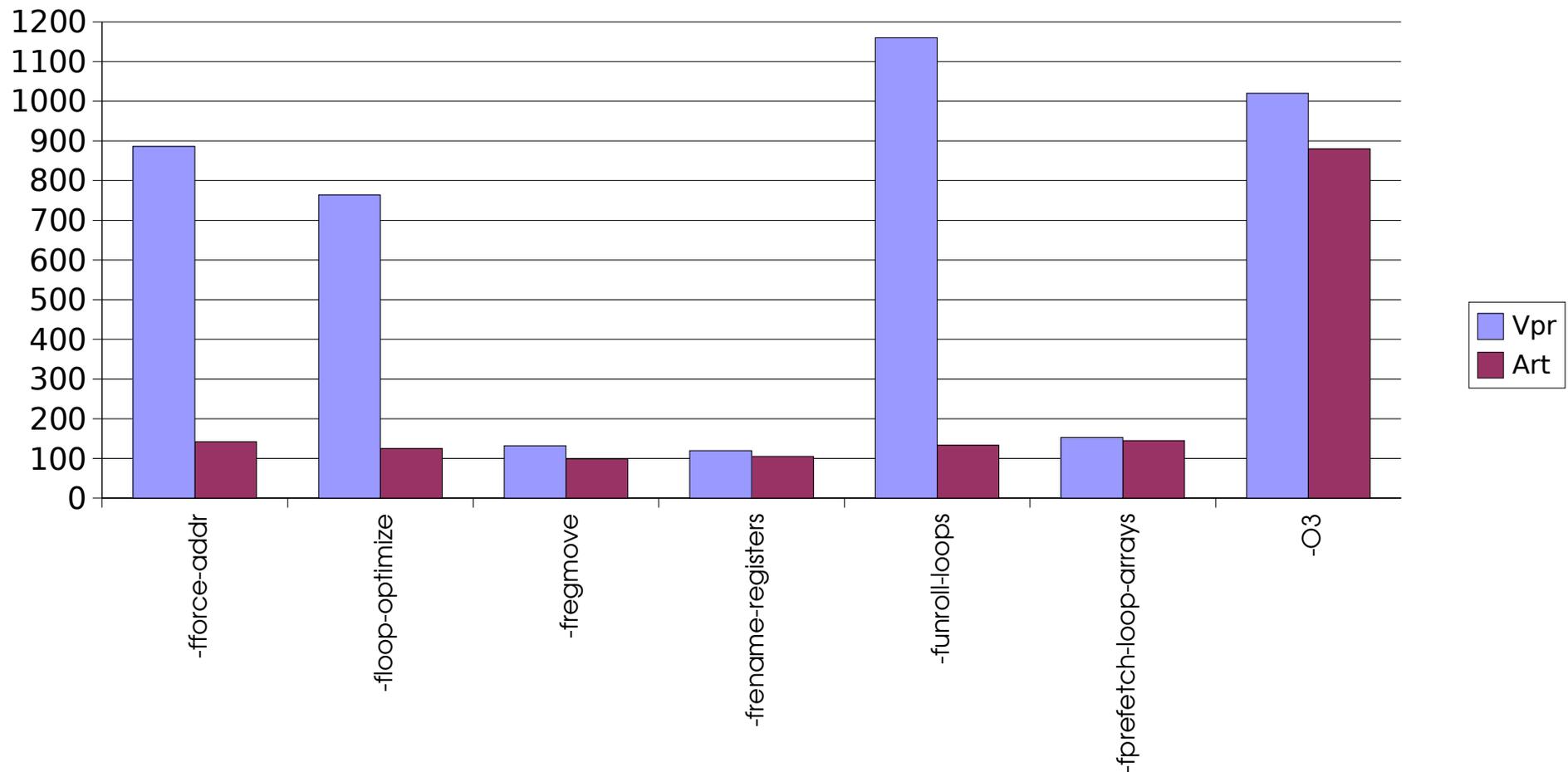
# Resultados y Conclusiones

- Mejores Optimizaciones para Entero (VPR) respecto a Sin Optimización:
  - **-fforce-mem** (16,8%)
  - **-fcse-skip-blocks** (12.5%)
  - **-fgse** (12,9%)
  - **-fcrossjumping** (10.7%)
  - **-fif-conversion** (6,7%)
  - **-fexpensive-optimizations** (11,9%)
  - **-fschedule-insns** (9,8%)
  - **-fschedule-insns2** (7,2%)

# Resultados y Conclusiones

- Mejores Optimizaciones para Punto Flotante (ART)

## Punto Flotante



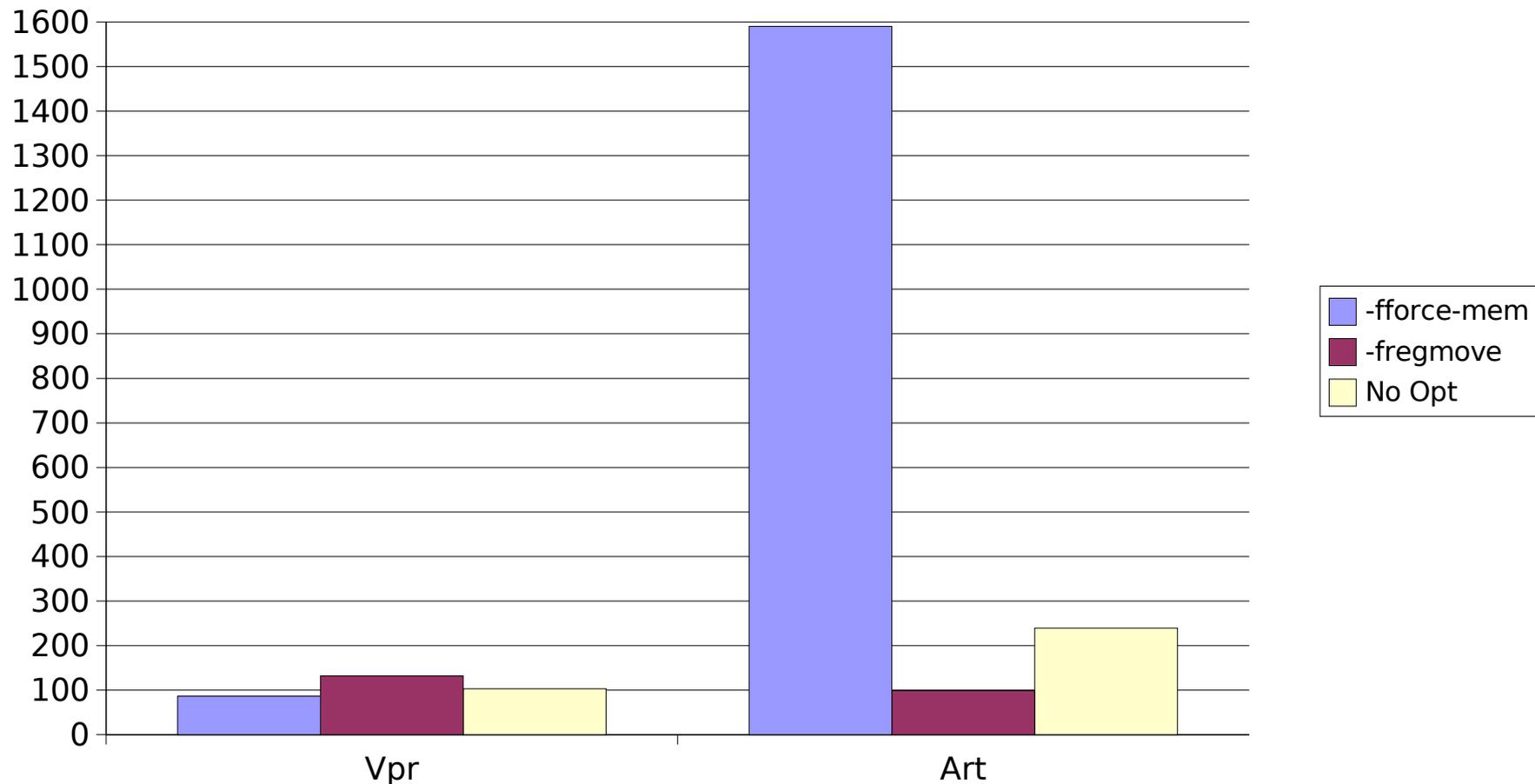
# Resultados y Conclusiones

- Mejores Optimizaciones para Punto Flotante (ART) respecto a Sin Optimización:
  - **-fforce-addr** (97%)
  - **-fno-function-cse** (51%)
  - **-fgcse** (103%)
  - **-floop-optimize** (114%)
  - **-fcrossjumping** (97%)
  - **-fif-conversion** (111%)
  - **-fregmove** (139,9%)
  - **-fschedule-insns** (101%)
  - **-fcaller-saves** (86%)
  - **-frename-registers** (139%)
  - **-funroll-loops** (106%)
  - **-fprefetch-loop-arrays** (94%)

# Resultados y Conclusiones

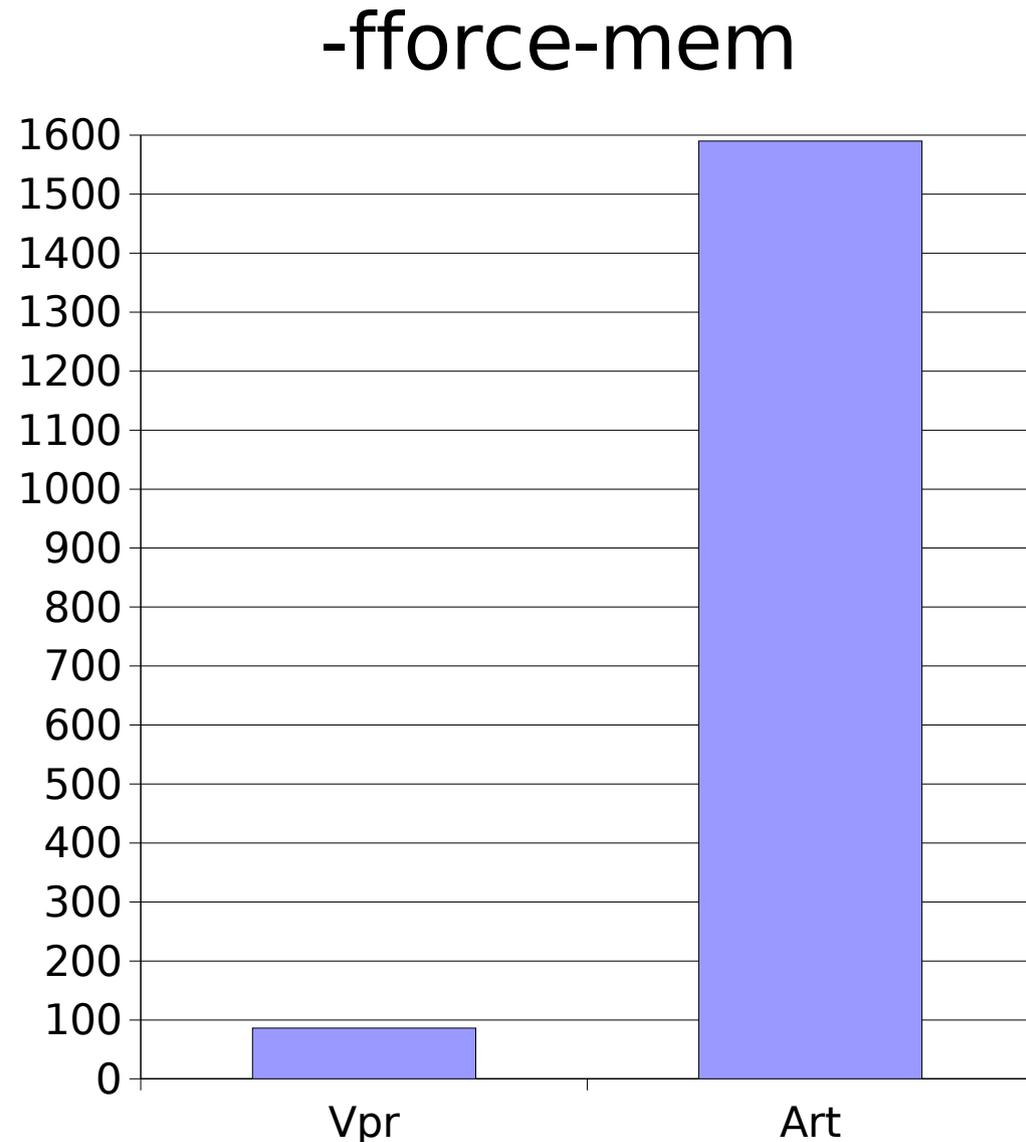
- La mejor para cada tipo

Las mejores



# Resultados y Conclusiones

- **-fforce-mem**
  - Fuerza a los operandos de memoria para que sean copiados en registros antes de realizar operaciones aritméticas sobre los mismos
  - Referencias a memoria convertidas en potenciales subexpresiones comunes

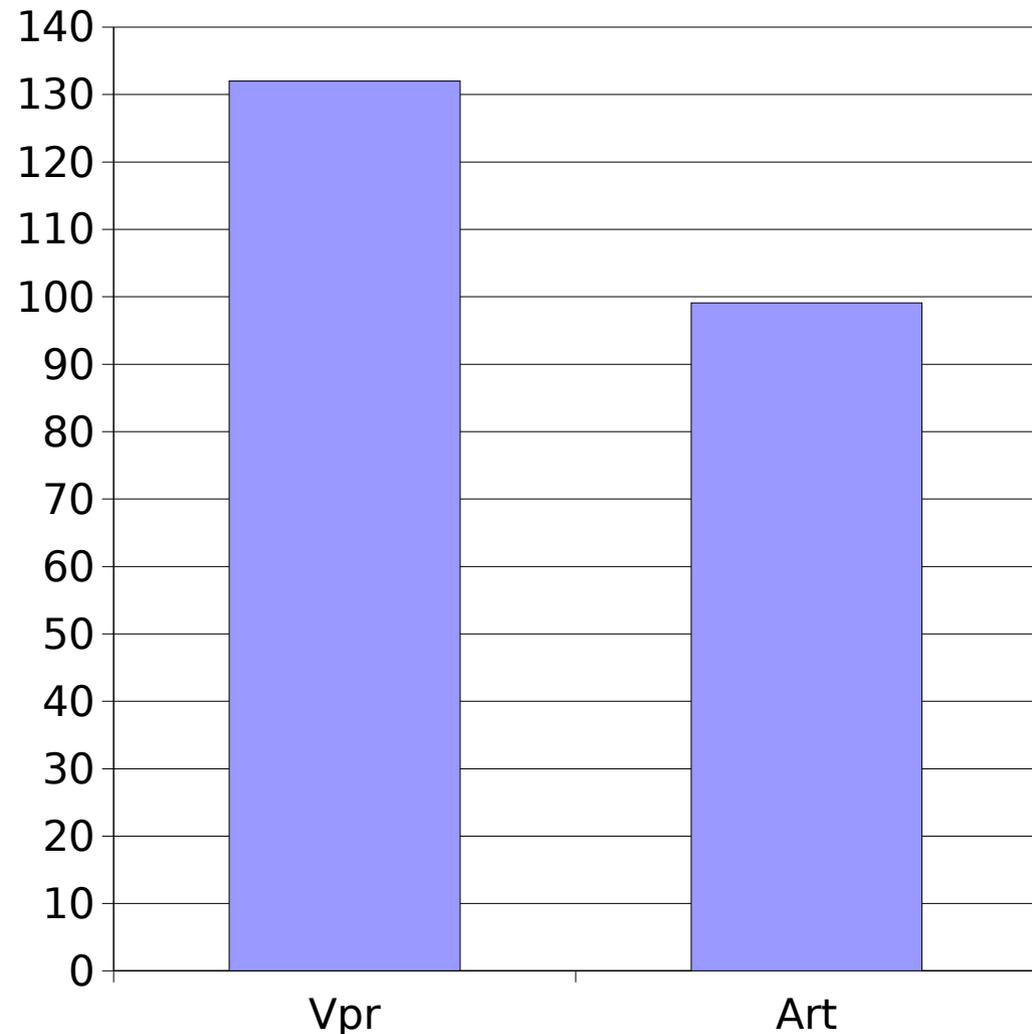


# Resultados y Conclusiones

- **-fregmove**

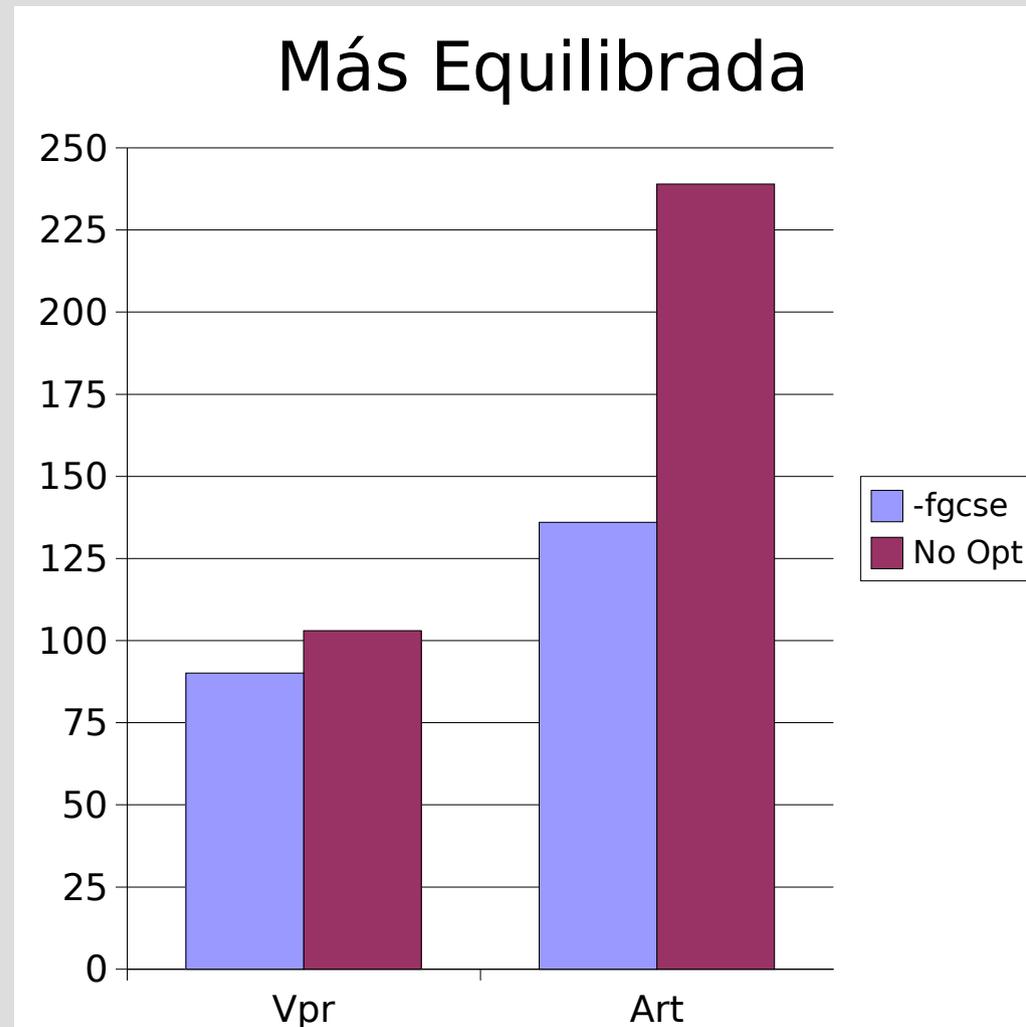
- Intenta reasignar los números de registro
  - En “move”
  - En instrucciones manejadas como operandos
- Maximizar permanencia de datos en cada registro

## -fregmove



# Resultados y Conclusiones

- Optimización más equilibrada (**-fgcse**)
- Realiza una pasada en la que se eliminan subexpresiones globales comunes
  - Propagación de constantes globales y copias
- Contiene otras:
  - **-fgcse-lm**
    - Mover “loads” que pueden emparejarse con “stores”
  - **-fgcse-sm**
    - Mover “stores” fuera de bucles



# Resultados y Conclusiones

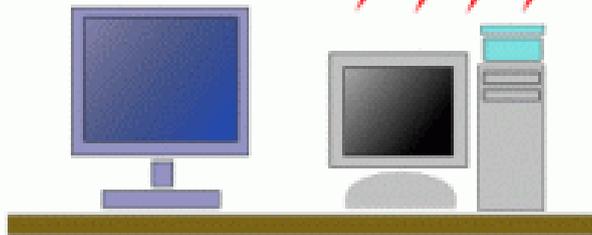
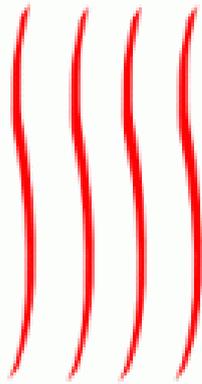
- Desaconsejado el uso de **-O2** y **-O3**
  - **-Os** sólo reduce tamaño del programa
  - Optimizaciones agrupadas
    - Algunas mejoran el rendimiento
    - Otras lo empeoran
  - Ventaja: No perder tiempo en estudiar el entorno
  - Aun así:
    - **-O2** – Para Entero
    - **-O3** – Para Punto Flotante

# Gracias por vuestra atención

## SOBRECALENTAMIENTO

¿Qué te pasa?  
Te veo un poco  
caliente hoy.

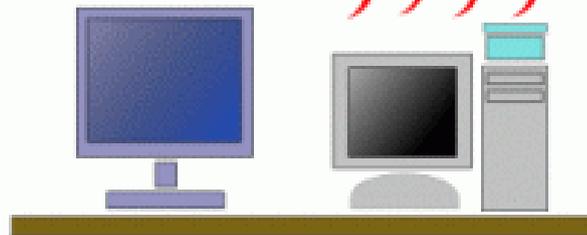
Si al final tendrán  
razón los que  
dicen que los  
Athlon se  
calientan mucho ...



¿Qué me pasa?. Mira lo que  
me han mandado hacer:

```
# /etc/init.d/athcool stop  
# cd /usr/src/linux  
# make
```

Y además estoy compilando  
OpenOffice, KDE y Gnome ...



¿Y a qué viene toda esa  
sobrecarga de trabajo?

Me temo que se ha estropeado  
el microondas que hay en  
la sala del café ...

Chicos ...  
La comida ya  
está caliente ...

