

# Forth im FPGA

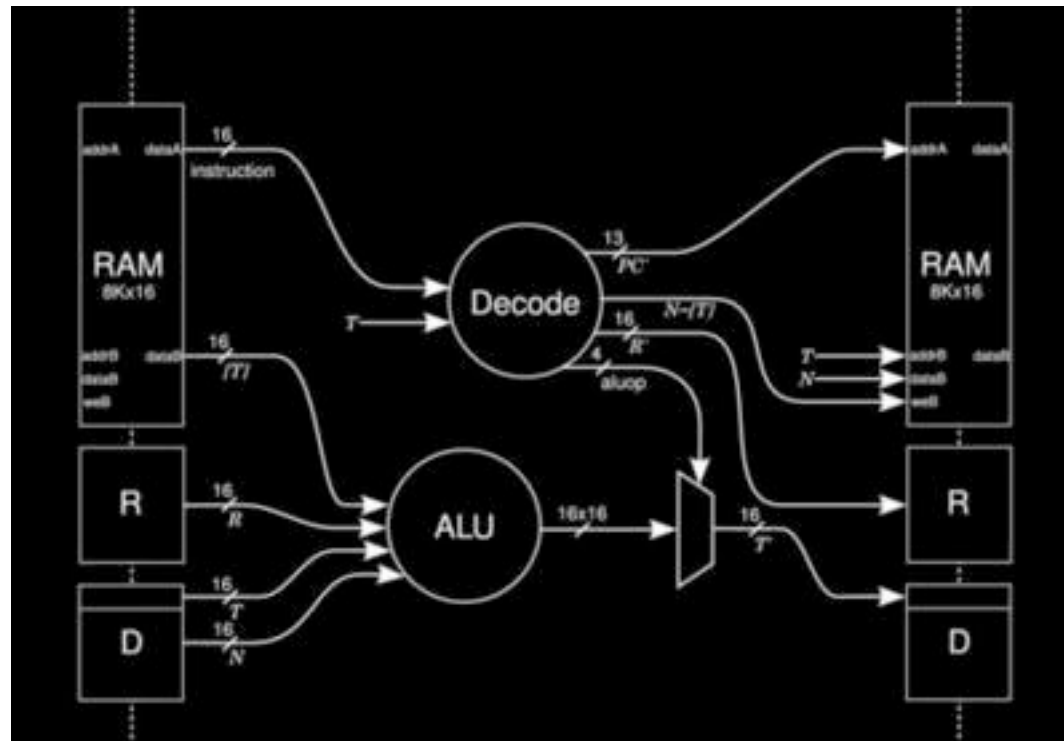
## 100 Mhz Forth-Maschine

Ulrich Hoffmann <[uho@forth-ev.de](mailto:uho@forth-ev.de)>

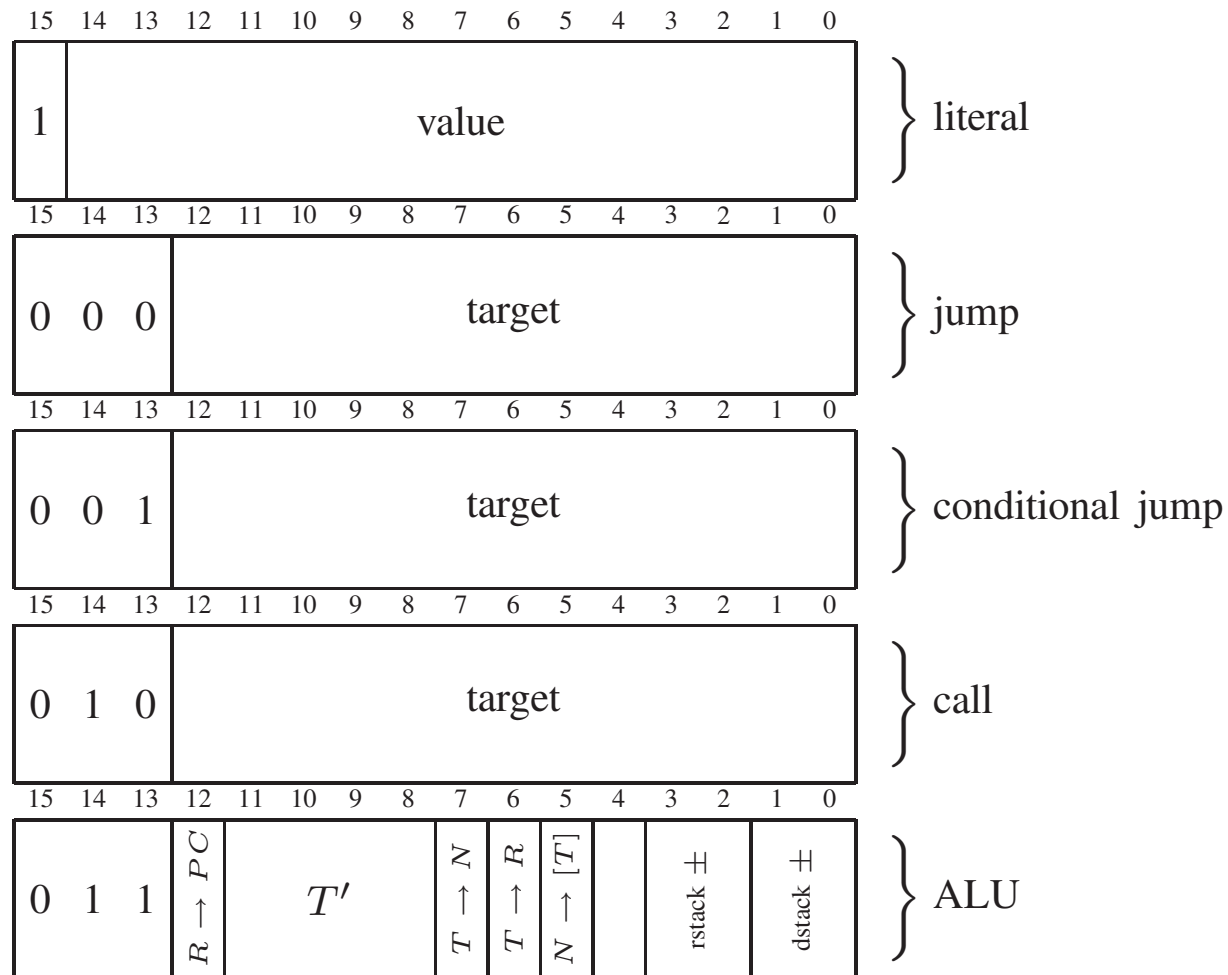
# J1-Softcore

- James Bowman euroForth 2010
- 16 Bit Forth CPU im Geiste vom Novix NC4000
- viele Forth-Befehle als Maschineninstruktionen
- Auch @ und ! sind  
I-Zyklus-Befehle (dual-ported RAM).
- Verilog < 200 Zeilen

# J1-Aufbau



# J I -Instruktionssatz



# J1-ALU-Operationen

code	operation
0	$T$
1	$N$
2	$T + N$
3	$T \text{ and } N$
4	$T \text{ or } N$
5	$T \text{ xor } N$
6	$\sim T$
7	$N = T$
8	$N < T$
9	$N \text{ rshift } T$
10	$T - 1$
11	$R$
12	$[T]$
13	$N \text{ lshift } T$
14	depth
15	$N \text{ u} < T$

# Gameduino

- James Bowman's  
    Arduino-Shield für Retro-Gaming
- Xilinx-Spartan 3 FPGA
- VGA (400x300 pixels in 512 colors), Sound
- Gameduino Library
- JI-Softcore
- aus Library ansprechbar
- Open Source



# jleForth

- GitHub-Projekt jleForth versucht ein eForth auf dem JI zu realisieren
- nur für einen JI-Simulator
- Portierung auf Papilio-Pro FPGA beigesteuert
- Xilinx-Spartan 6
- 66Mhz
- Open Source



# Portierung auf LOGI PI-FPGA

- LOGI PI-Aufsteck-Board für Raspberry Pi
- Xilinx Spartan 6
- 100Mhz
- Open Source





# cmForth

- Chuck Moore's Forth für den NC4000
- 30 Screens
  - Compiler, Interpreter
  - optimierendes Compiler
  - UART
  - Massenspeicher
  - MetaCompiler (extrem einfach)
- Ein Juwel:
  - Mikrokosmos von Hard und Software
  - Selbstreproduzierendes System

# Ziel: cmForth für JI

- Portierung von cmForth auf JI
- Keine Änderung am JI
- so wenig Änderungen an cmForth wie nötig
  - UART in Hardware
  - Anderer Maschinencode
- In Arbeit

**Demo**

# Ausblick

- cmForth auf J1
- Modifikationen
- Architektur-Experimente
- FPGAs haben 18-Bit-RAM → 18 Bit CPU?

# Diskussion