HercuLeS

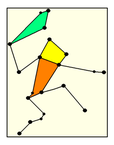
Automated synthesis of FSMD-based accelerators for hardware compilation

Nikolaos Kavvadias[1] *(nikos@nkavvadias.com) and Kostas Masselos*[2] *(kmas@uop.gr)*

[1]CEO, Ajax Compilers, Athens, Greece

[2]Dept. of Computer Science and Technology, University of Peloponnese, 22100 Tripoli, Greece

What is HercuLeS?



An extensible, high-level synthesis (HLS) environment for whole-program hardware compilation that allows pluggable optimizations and analyses

• Overcomes limitations: (1) nonstandard source languages, (2) insufficient representations, (3) maintenance difficulties, (4) mandating the use of code templates, (5) vendor dependence

1. Connects to GIMPLE/GCC as a frontend scenario

2. Uses the succinct NAC intermediate representation

3. Optimizations added as self-contained external modules

4. Hardware compilation engine is entirely graph-based

5. The generated HDL is human-readable and completely vendor- and technology-independent

Automation for FSMD-based accelerators

• The generated accelerators adhere to an extended FSMD model of computation

• This model allows for:

– Synchronous embedded memory accesses

– Intermodule communication (hierarchical FSMDs)

– Hardware-optimizing transformations such as operation chaining

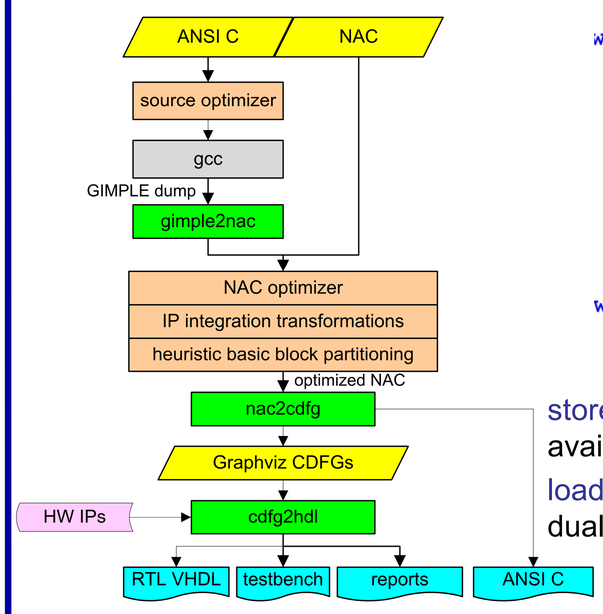
– Automatic IP integration

• More on optimizations:

– Optimizations possible at the source, NAC, graph and VHDL level

– Pluggable loop-oriented transformations, advanced arithmetic optimizations, graph rewriting, VHDL “hacks”

Overview



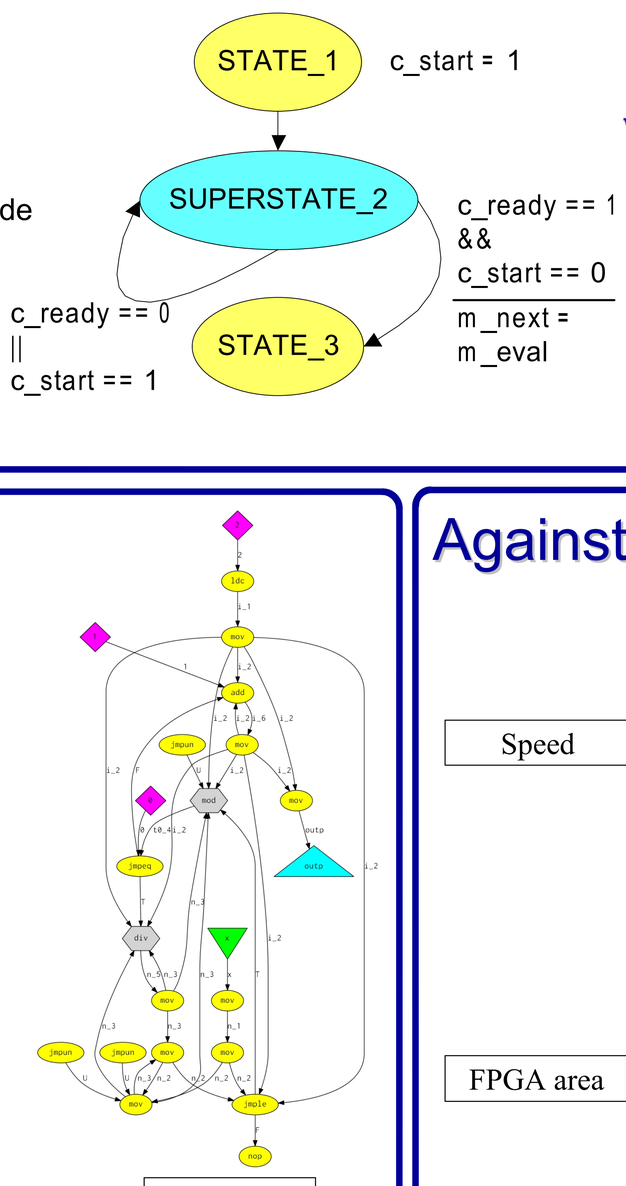
Memory accesses

```
when STATE_1 =>
  mem_addr <= index;
  waitstate_next <= not (waitstate_reg);
  if (waitstate_reg = '1') then
    mysignal_next <= mem_dout;
    next_state <= STATE_2;
  else
    next_state <= STATE_1;
  end if;
when STATE_2 =>
  ...
```

store: Raises BRAM write. Stored data are made available in the subsequent machine cycle

load: Requires a waitstate register to devise a dual-cycle substate (address + data cycles)

Hierarchical FSMDs



Chaining

```
...
when S_1_3 =>
  t3_next <= "000"&x_reg(15 downto 3);
  t4_next <= "0"&y_reg(15 downto 1);
  next_state <= S_1_4;
when S_1_4 =>
  t5_next <= x_reg - t3_reg;
  next_state <= S_1_5;
when S_1_5 =>
  t6_next <= t4_reg + t5_reg;
  next_state <= S_1_6;

when S_1_1 =>
  ...
  t3_next <= "000"&x_next(15 downto 3);
  t4_next <= "0"&y_next(15 downto 1);
  t5_next <= x_next - t3_next;
  t6_next <= t4_next + t5_next;
  ...
```

- Assigns dependent SSA operations to a single control step

- Merges successive ASAP states; a heuristic constrains long paths

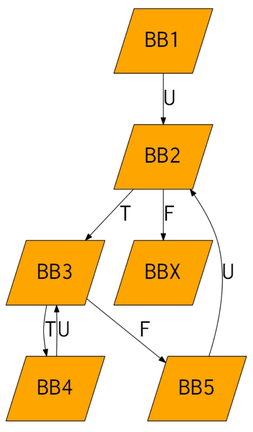
Example: Prime factorization

```
void pfactor(uint x,
  uint *outp) {
  uint i=2, n=x;
  while (i <= n) {
  while ((n%i)==0) {
    n = n / i;
    *outp = i;
    }
    i = i + 1;
  }
}
```

ANSI C code

```
procedure pfactor(
  in u16 x,out u16 outp){
  localvar u16 i,n,t0;
BB1:
  n<=mov x; i<=ldc 2;
BB2:
  BB3,BBX<=jmple i,n;
BB3:
  t0<=modu(n,i);
  BB4,BB5<=jmpeq t0,0;
BB4:
  n<=divu(n,i);
  outp<=mov i;
  BB3<=jmpun;
BB5:
  i<=add i,1; BB2<=jmpun;
BBX: nop;}
```

NAC IR

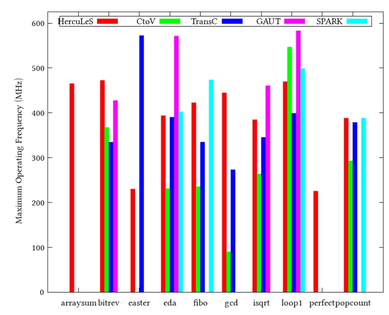


CFG

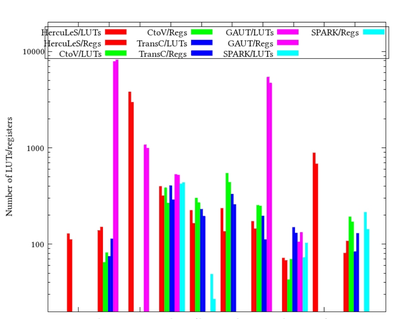
SSA CDFG

Against academic tools

Speed



FPGA area



Additional information

HercuLeS is marketed by Ajax Compilers, Unl.. Its inaugural release is expected in Q3, 2012 (IP design services are already provided).

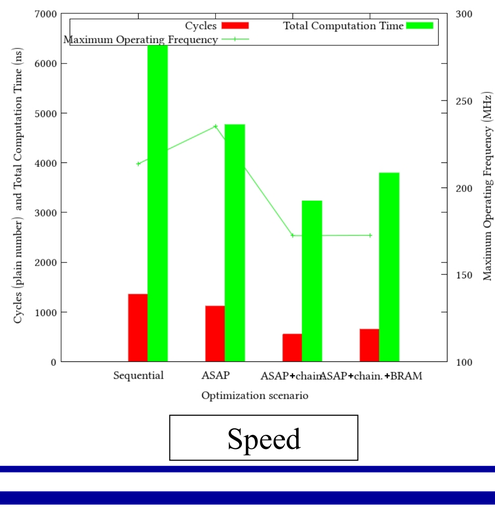
New features include: support for IEEE-754/custom floating-point arithmetic and source-level optimization.

[1] HercuLeS website.
http://www.nkavvadias.com/hercules/
[2] HercuLeS freely accessible web interface.
http://www.nkavvadias.com/cgi-bin/herc.cgi
[3] List of available algorithmic hardware IPs.
http://www.nkavvadias.com/hercules/algorithmic-ips.html

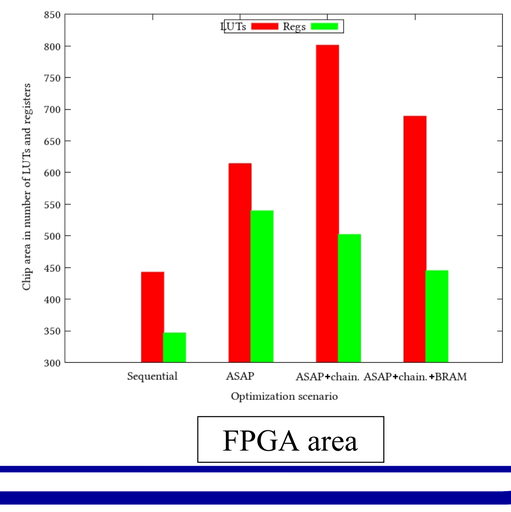
Performance of FSMDs

Benchmark	C	NAC	Graphviz	VHDL
edgedet	35	145	873	1921
float2half	25	71	157	370
fsme	65	159	1483	2730
half2float	12	32	55	174
icbrt	18	36	83	213
isqrt	20	28	84	199
mandel	60	108	259	639
matmult	40	94	763	1511
sierpinski	51	70	300	630
smwat	68	159	753	1615
walsh	32	71	326	704
yuv2rgba	27	98	240	679

Benchmarks and line statistics



Speed



FPGA area