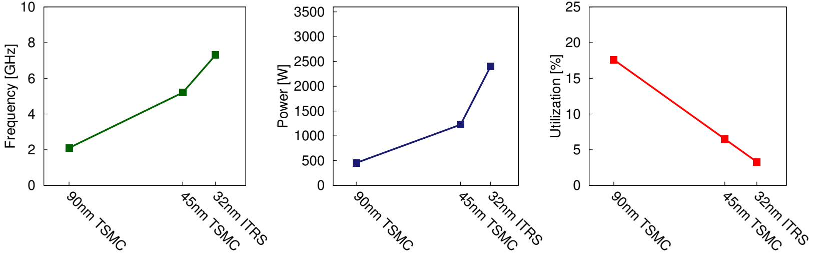
Dark Silicon and its Implications for Future Processor Design

Max Menges

22. December 2015

Motivation


Frequency [GHz]
10
8
6
4
2
0
90nm TSMC
45nm TSMC
32nm ITRS
Power [W]
3500
3000
2500
2000
1500
1000
500
0
90nm TSMC
45nm TSMC
32nm ITRS
Utilization [%]
25
20
15
10
5
0
90nm TSMC
45nm TSMC
32nm ITRS

What is dark silicon?

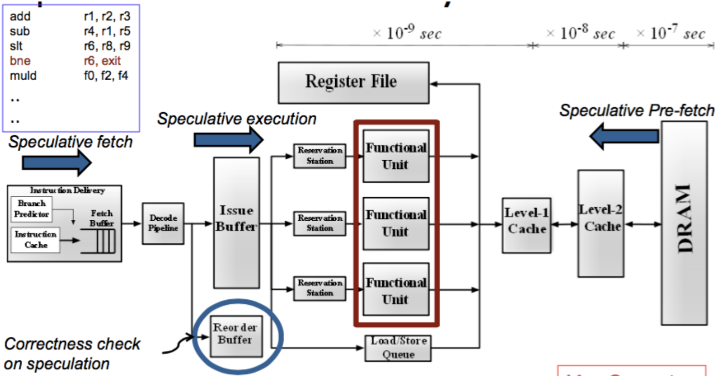
The **utilization wall** refers to the part of a chip which can actively be used within the power budget at full frequency. This is dropping exponentially with each process generation. The unused silicon that is left unpowered is referred to as **dark silicon**.

1 Introduction: What is dark silicon?

2 Background: Where does dark silicon come from?

3 Conservation Cores: Utilizing dark silicon.

What is in a CPU?



```
add     r1, r2, r3
sub     r4, r1, r5
slt     r6, r8, r9
bne     r6, exit
muld    f0, f2, f4
..
..
```

Speculative fetch

Speculative execution

Speculative Pre-fetch

Correctness check on speculation

Locality of reference (spatial/temporal)

Predictable control flow

Latency Hiding

Frequency Scaling

You Cannot Hide Energy!

Sudha Yalamanchili, *Architectural Alternatives for Energy Efficient Performance Scaling*, VLSI Conference, 2013

Power Consumption

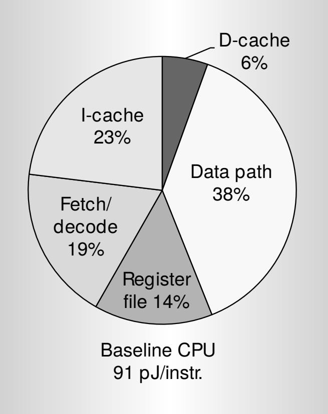
- Dynamic power when switching

$$P_{dyn} = \alpha C_L V^2 f$$

- Subthreshold leakage

$$P_{leak} \propto e^{\frac{V_{GS}-V_{th}}{nV_T}}$$

- Gate-oxide leakage due to quantum meachanical tunneling



Goulding-Hotta et al. *The GreenDroid Mobile Application Processor: An Architecture for Silicon's Dark Future*, Micro IEEE, vol.31, no.2, 2011

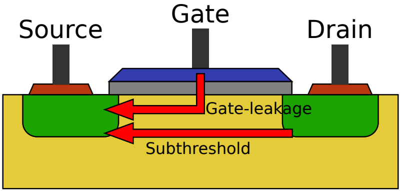
Technology Scaling

- Scale geometries by factor $S = 1.4$, e.g. from $90nm$ to $65nm$
- Ideally scale all voltages etc. accordingly
- Devices per chip at constant area A increases by $S^2 \approx 2$
- No increase in power due to constant energy density

$$P_S = \frac{1}{S}C \cdot \frac{1}{S^2}V^2 \cdot Sf = \frac{1}{S^2}P$$

Param.	Description	Rel.	Classical Scaling
W, L	Transistor dimensions		$1/S$
V_{dd}, V_{th}	Supply & threshold voltages		$1/S$
t_{ox}	Oxide thicknes		$1/S$
C	Gate capacitance	WL/t_{ox}	$1/S$
p	Power per device	CV^2f	$1/S^2$
P	Full die, full power	Dp	1
U	Utilization	B/P	1

Dennard Scaling



- **Dennard's Law:** The power density in a transistor stays constant as geometries shrink
- Breakdown of Dennard scaling due to leakage current at about 2005-2007, around the 65nm process
- Limited by subthreshold leakage current and QM tunneling effects at thin gate oxide

Technology Scaling II

- Post Dennard scaling is leakage limited, V_{dd} and V_{th} cannot be lowered
- Continue to scale geometries by factor $S = 1.4$
- Utilization will decrease with a factor of $1/S^2$ with each new process generation

Param.	Description	Rel.	Classical Scaling	Leakage Limited
W, L	Transistor dimensions		$1/S$	$1/S$
V_{dd}, V_{th}	Supply & threshold voltages		$\mathbf{1/S}$	$\mathbf{1}$
t_{ox}	Oxide thicknes		$1/S$	$1/S$
C	Gate capacitance	WL/t_{ox}	$1/S$	$1/S$
p	Power per device	CV^2f	$\mathbf{1/S^2}$	$\mathbf{1}$
P	Full die, full power	Dp	$\mathbf{1}$	$\mathbf{1/S^2}$
U	Utilization	B/P	$\mathbf{1}$	$\mathbf{1/S^2}$

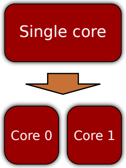
Muticore CPUs

- Single core CPUs derive speedup from frequency gains

Single core

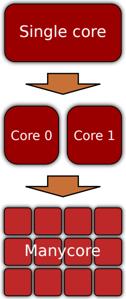
Muticore CPUs

- Single core CPUs derive speedup from frequency gains
- Transition to multicore CPUs
  - Reduce clock frequency to 80%
  - Power: $P_M = 0.512 \cdot P_S$
  - Gain 1.6x performance by adding a second core



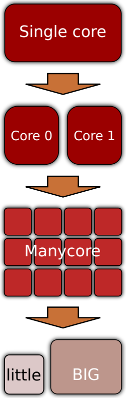
Muticore CPUs

- Single core CPUs derive speedup from frequency gains
- Transition to multicore CPUs
  - Reduce clock frequency to 80%
  - Power: $P_M = 0.512 \cdot P_S$
  - Gain 1.6x performance by adding a second core
- Low frequency, throughput orientated manycores for regular floating point arithmetics



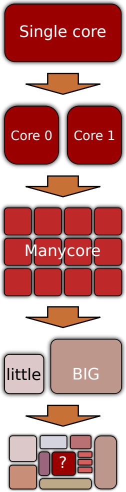
Muticore CPUs

- Single core CPUs derive speedup from frequency gains
- Transition to multicore CPUs
  - Reduce clock frequency to 80%
  - Power: $P_M = 0.512 \cdot P_S$
  - Gain 1.6x performance by adding a second core
- Low frequency, throughput orientated manycores for regular floating point arithmetics
- Heterogeneous cores for energy efficient computations



Muticore CPUs

- Single core CPUs derive speedup from frequency gains
- Transition to multicore CPUs
  - Reduce clock frequency to 80%
  - Power: $P_M = 0.512 \cdot P_S$
  - Gain 1.6x performance by adding a second core
- Low frequency, throughput orientated manycores for regular floating point arithmetics
- Heterogeneous cores for energy efficient computations
- **Specialized hardware**



1 Introduction: What is dark silicon?

2 Background: Where does dark silicon come from?

3 Conservation Cores: Utilizing dark silicon.

The GreenDroid

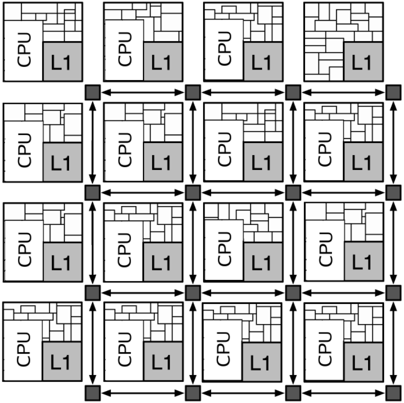
The GreenDroid is a proposed energy efficient chip design targeted at Andriod mobile phones

- Power limitation of $3W$
- Android OS ideal as a limited software platform
  - User applications run in a VM
  - Common applications: Web browser, e-mail, media player
  - Short replacement cycle
- Utilize *dark silicon* in energy efficient *conservation cores*

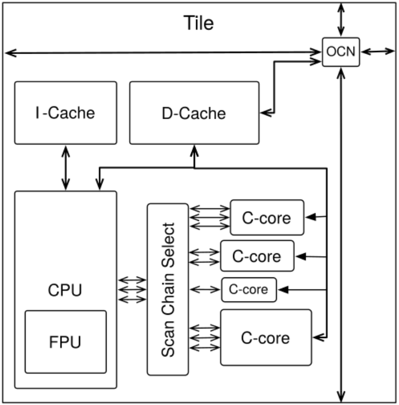
What is a C-Core?

- Specialized core which implements software functions in hardware
- Analyze most frequently used functions and translate to verilog code
- C-cores are coupled to a host CPU via L1 cache and scan chain

Tile layout



Chip layout



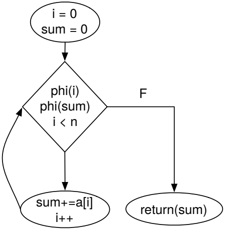
Tile layout

Venkatesh et al. *Conservation Cores: Reducing the Energy of Mature Computations*, SIGARCH Comput. Archit. News, March 2010

Generating the Cores I

- Characterize workload and identify regions of *hot* code
- Translate CFG to state machine
- Compile code
    - Compare c-core specs and code
    - Generate stubs that allow execution on c-core or CPU

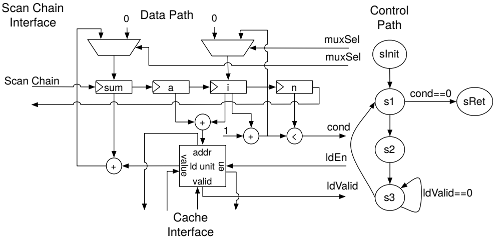
```
computeArraySum
{
   sum = 0;
   for(i = 0; i < n; i++)
   {
      sum += a[i];
   }
   return(sum);
}
```



Venkatesh et al. *Conservation Cores: Reducing the Energy of Mature Computations*, SIGARCH Comput. Archit. News, March 2010

Generating the Cores II

- Change code to allow patching
  - Constants
  - Operators
  - Control flow
- Insert scan chains and synthesise to hardware
- Add exception bit to each state transition

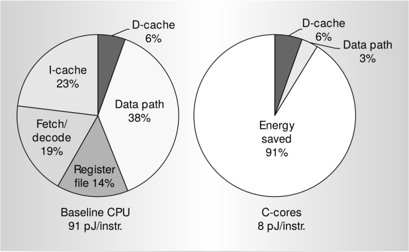


Venkatesh et al. *Conservation Cores: Reducing the Energy of Mature Computations*, SIGARCH Comput. Archit. News, March 2010

Execution

- Decide at runtime to execute code on CPU or c-core
- Pass function arguments via scan chain
- Start c-core execution with a single bit master scan chain
- Once complete the c-core throws an exception and transfers controll back to the CPU
    - In case of a patched core, pass control back and forth between CPU and c-core

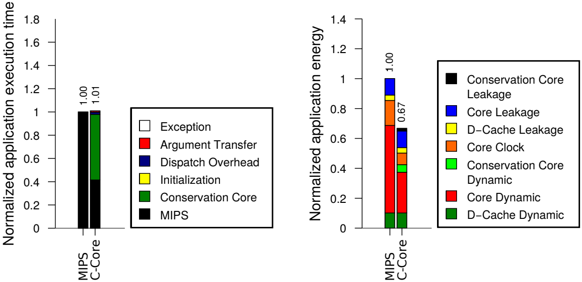
Results I



Goulding-Hotta et al. *The GreenDroid Mobile Application Processor: An Architecture for Silicon's Dark Future*, Micro IEEE, vol.31, no.2, 2011

- Reduce energy by removing instruction fetch and decode and simplifying the data path
- C-core executions in one example tile span ≈ 10% of code

Results II



Venkatesh et al. *Conservation Cores: Reducing the Energy of Mature Computations*, SIGARCH Comput. Archit. News, March 2010

Average energy and execution times of bzip2, cjpeg, djpeg, mcf and vpr in c-cores and the MIPS core.

Conclusion

- Dark silicon is the part of chip which cannot be operated within the power budget

Conclusion

- Dark silicon is the part of chip which cannot be operated within the power budget
- It is a result of scaling process technologies without reducing supply voltage accordingly

Conclusion

- Dark silicon is the part of chip which cannot be operated within the power budget
- It is a result of scaling process technologies without reducing supply voltage accordingly
- Introduce specialized energy efficient hardware to utilize dark silicon

Thank you for your attention.
Any questions?