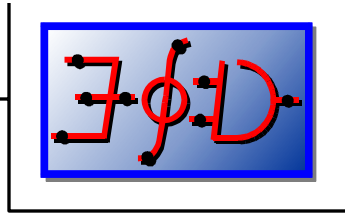
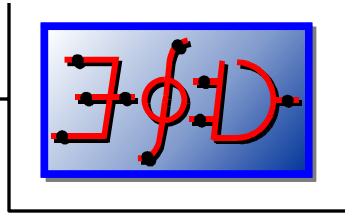


Behavioural Modeling and Simulation of a Switched-Current Phase Locked Loop

Peter R. Wilson, Reuben Wilcock, Bashir Al-Hashimi & Andrew D. Brown

Electronic Systems Design Group
School of Electronics and Computer Science
University of Southampton, UK
prw@ecs.soton.ac.uk



Introduction

As part of our current IC design efforts we need to design some mid range (low frequency?) Phase Locked Loops.

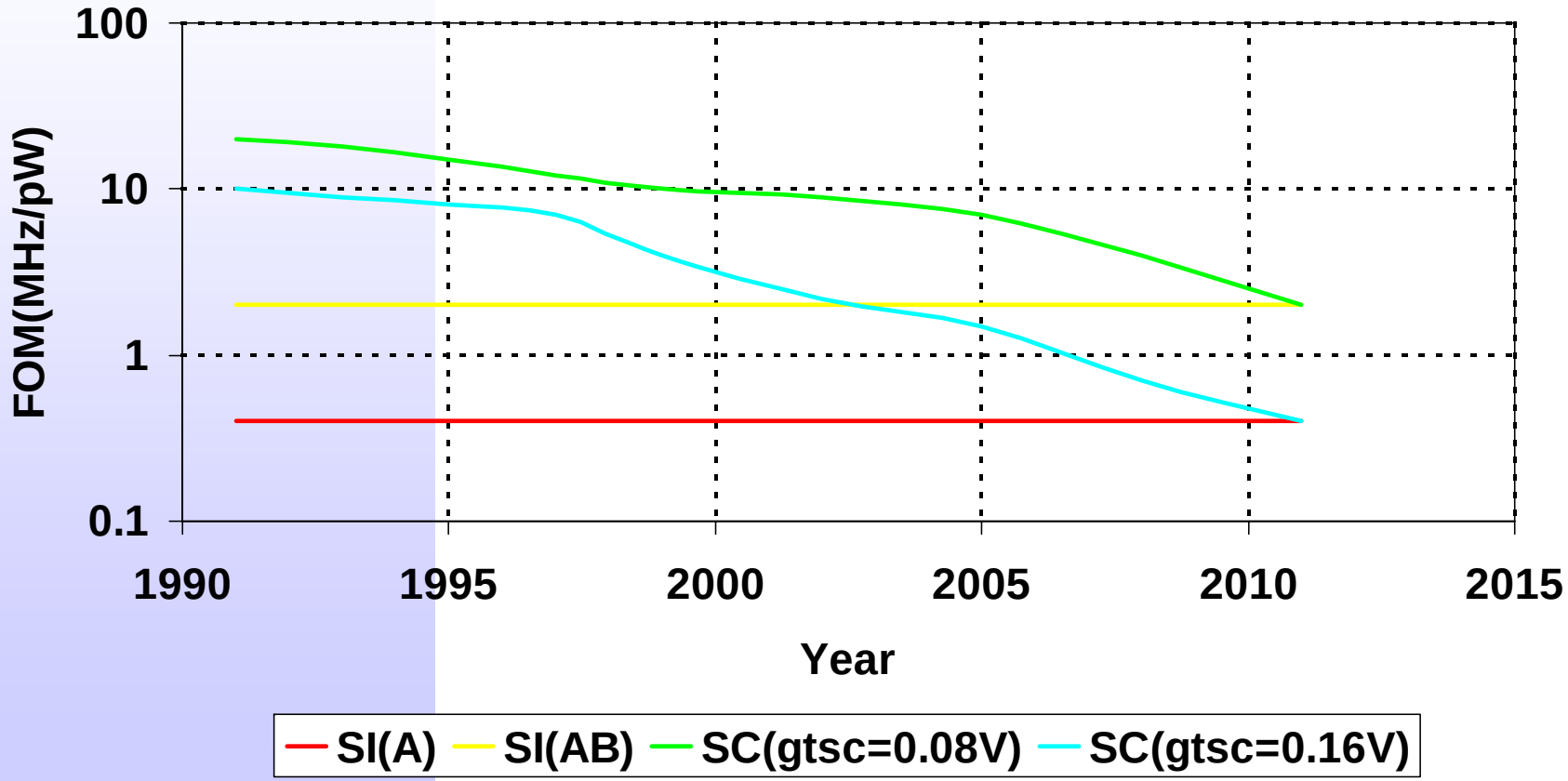
We are looking at 100MHz-500MHz FSK applications.

We need to integrate onto a process of 120nm.

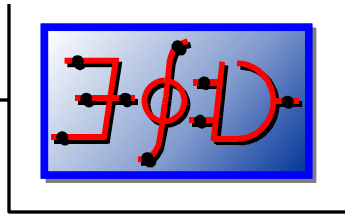
Key Analogue Design Issues…

- Main technique to tackle poor output conductance of short channel devices is cascoding.
- With DSM, threshold voltage does not scale linearly with supply voltage which reduces the possibility of cascoding techniques.
- Lower supply voltages directly impact on the dynamic range of voltage mode circuits.
- DSM processes often either do not support passive component integration or their sizes/costs are prohibitive.

Switched Cap or Switched Current?



FOM=f(area,power,SNR)



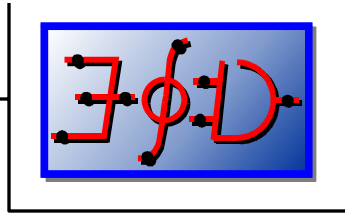
Suitability of SI for DSM

Addressing the DSM problems with SI:

- **Recent SI memory cells use neutralisation techniques to improve output conductance hence no cascoding necessary**
- **Current mode ensures suitability for low supply voltages**
- **Analogue functionality with NO passive components**
  - Smaller area
  - Suitability for any digital process

Main practical issues with SI:

- **The operation of SI circuits depends on the parasitic themselves hence have no choice but to use transistor level simulators**
- **Hand design of even simple blocks can take time**



Background: Switched Currents

"Switched Currents" (SI) is a current domain sampled data signal processing technique.

- **Drain current maintained, despite the gate opened, through the gate-oxide capacitance of the device.**
- **The most basic building block of SI is a memory cell**
- **SI circuits require only a basic digital process**

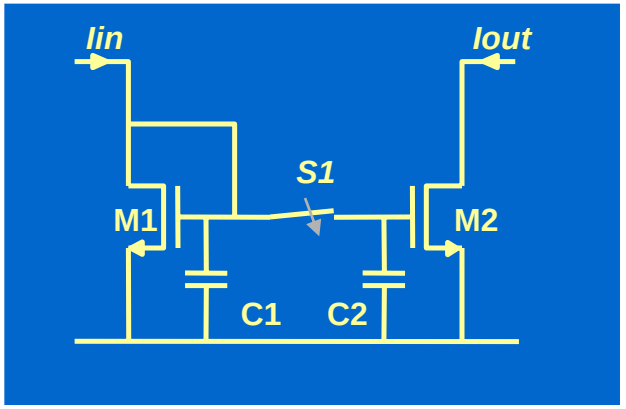
Switched Currents have been around for a while – but it is now that Digital processes have stretched away from their Analogue counterparts that it has become more attractive as a means of closing the gap.

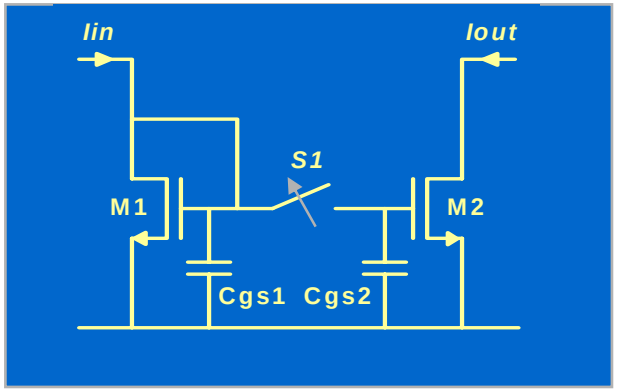
Switched Currents in a nutshell

If the switch is closed, and assuming ideal conditions, the circuit operates as a current mirror.

On opening the switch, the circuit operates as a current mode track and hold – as C2 keeps the drain current in M2 flowing.

Capacitors C1 and C2 can be the MOS transistor's effective gate-source capacitances, Cgs1 and Cgs2 respectively.





So what's the problem?

We have developed design methods and tools (to a certain extent) to design using SI ✓

We have successfully designed example circuits, produced silicon and it works ✓

BUT:

- **Each design is handcrafted**
- **The parameters need tuning**
- **Simulations take days (higher level behavioural modeling is *still* hardly used by IC designers)**

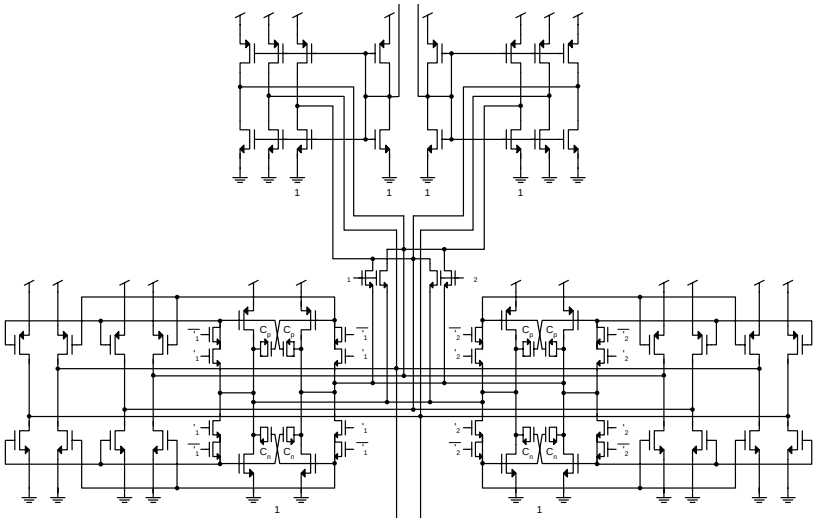
If we could only run many simulations *early* in the design phase using an architectural model, then we could establish the key design parameters *quickly*.

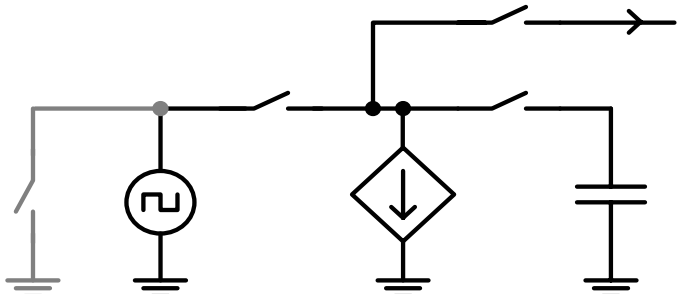
Model of the SI Memory Cell

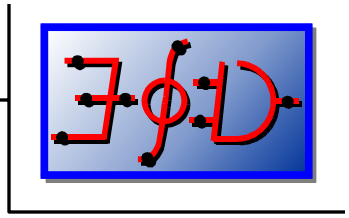
SI circuits can appear very complex but they all rely on simple building blocks.

We have developed behavioural models of the basic memory cell using fast behavioural switch models to replace the transistor models

- **allows much faster simulations, but retains some of the intrinsic switching behaviour of the design.**
- **Essential to model non-ideal behaviour since this is fundamental to SI operation (i.e. C_{gs})**





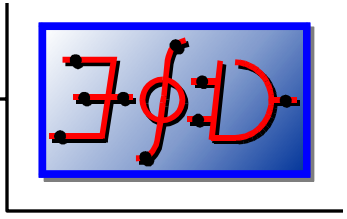


MAST memory cell model

Using MAST allows the model to be created using mixed analog and digital functions, with direct control of the interfaces and switching elements.

The sections of the model allow for a structured approach to the implementation of the key functions required.

- **Note only a subset of the model is shown for clarity and simplicity**



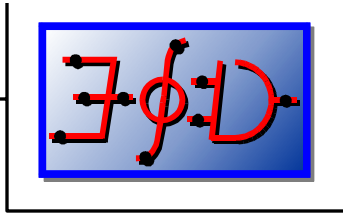
MAST memory cell model

The when(event_on()) function waits for a digital even on the clock input that drives the basic switching element in the memory cell

The model remains in the digital domain to create a sample of the internal voltage vgs, called vgs_d.

This is extremely fast and does not add ANY overhead to the analog model/simulation

```
when(event_on(phi)) {
  if (phi == 14_1) {
    schedule_event(time,sw,1)
  }
  else {
    schedule_event(time,vgs_d,vin)
    schedule_event(time,sw,0)
  }
}
values {
  vin = v(inp) - v(inm)
  if(sw==1) {
    iin = ic + vin*gm
    vgs = vin
  }
  else {
    iin = vgs*gm
    vgs = vgs_d
  }
}
equations {
  ic = d_by_dt(c*vin)
  i(outp->outm) += iin/gm
  i(inp->inm) += iin
}
```

MAST memory cell model

The values section is handling the basic analog functionality of the memory cell and calculating the scaled current during the tracking phase (sw = 1)

Note that in the else part of the if statement the value of vgs is set to the stored digital value – this is the memory aspect of the cell

```
when(event_on(phi)) {
  if (phi == 14_1) {
    schedule_event(time,sw,1)
  }
  else {
    schedule_event(time,vgs_d,vin)
    schedule_event(time,sw,0)
  }
}
values {
  vin = v(inp) - v(inm)
  if(sw==1) {
    iin = ic + vin*gm
    vgs = vin
  }
  else {
    iin = vgs*gm
    vgs = vgs_d
  }
}
equations {
  ic = d_by_dt(c*vin)
  i(outp->outm) += iin/gm
  i(inp->inm) += iin
}
```

MAST memory cell model

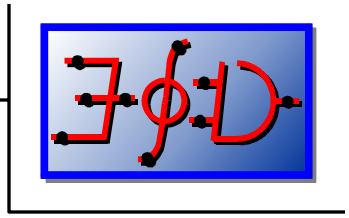
The equations section of the model calculates the correct input and output currents

it also models the effect of the input capacitance on the response of the cell.

```
when(event_on(phi)) {
  if (phi == l4_1) {
    schedule_event(time,sw,1)
  }
  else {
    schedule_event(time,vgs_d,vin)
    schedule_event(time,sw,0)
  }
}
values {
  vin = v(inp) - v(inm)
  if(sw==1) {
    iin = ic + vin*gm
    vgs = vin
  }
  else {
    iin = vgs*gm
    vgs = vgs_d
  }
}
equations {
  ic = d_by_dt(c*vin)
  i(outp->outm) += iin/gm
  i(inp->inm) += iin
}
```

School of Electronics and Computer Science

University of Southampton, UK



An obvious question here?

Why not use V*-AMS?

We did - we developed models in VHDL-AMS, Verilog-AMS, Spectre etc etc.

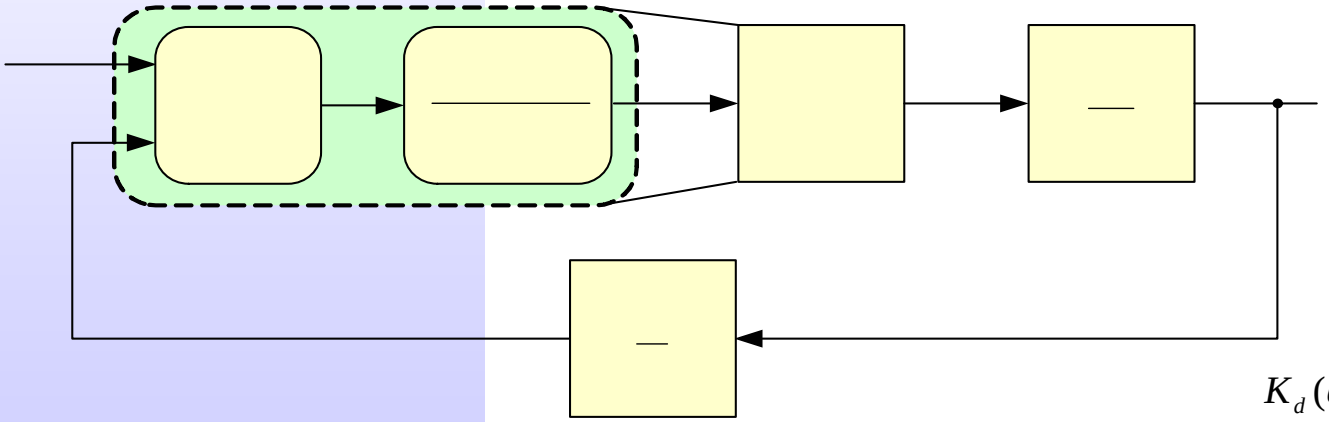
Mast was the quickest from a design perspective

- Simple and quick to create models
- Easy to debug (!)
- Excellent analysis tools

caveat.. BUT….. A relatively poor Cadence integration

SI PLL Core

A novel 2nd order PLL architecture was used that does not require a phase detector and offers benefits including small area, low power and compatibility with a standard digital CMOS process.



$$H(s) = \frac{g_m}{g_m + 2sC_{gs}}$$

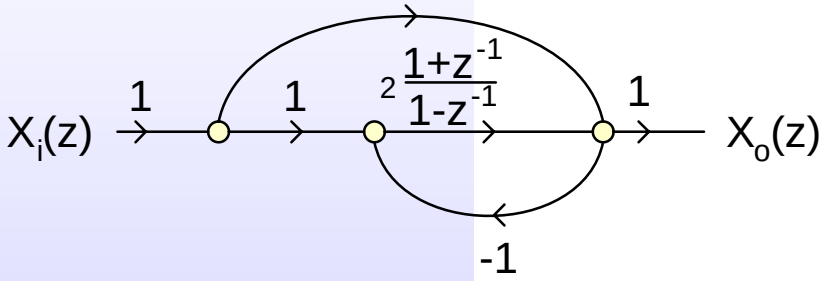
$$K_d = 0.66A$$

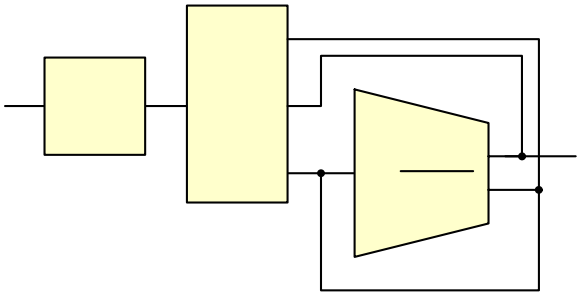
$$K_d(\theta_e) = \frac{g_m A \exp\left(\frac{-g_m(\pi - \theta_e)}{2\pi F_c C_{gs}}\right)}{\pi F_c C_{gs}\left(1 - \exp\left(\frac{-g_m \pi}{2\pi F_c C_{gs}}\right)\right)}$$

Perfect example of the application of SI to a fundamental block

SI PLL Core

Main SI content of SI PLL is the loop filter





Loop filter operation is key to achieving the correct PLL loop characteristics. It is essential to model the filter to determine correct operation considering non-ideal effects.

Main advantages resulting from application of SI to PLL:

- **Full integration of loop filter**
- **Area reduction**
- **Low power, and operation at low supply voltage**

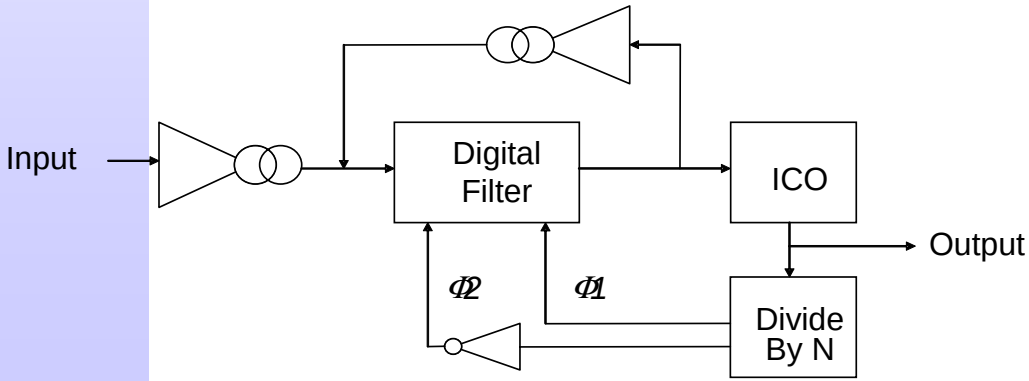
SI PLL

We have used this concept to design a complete PLL

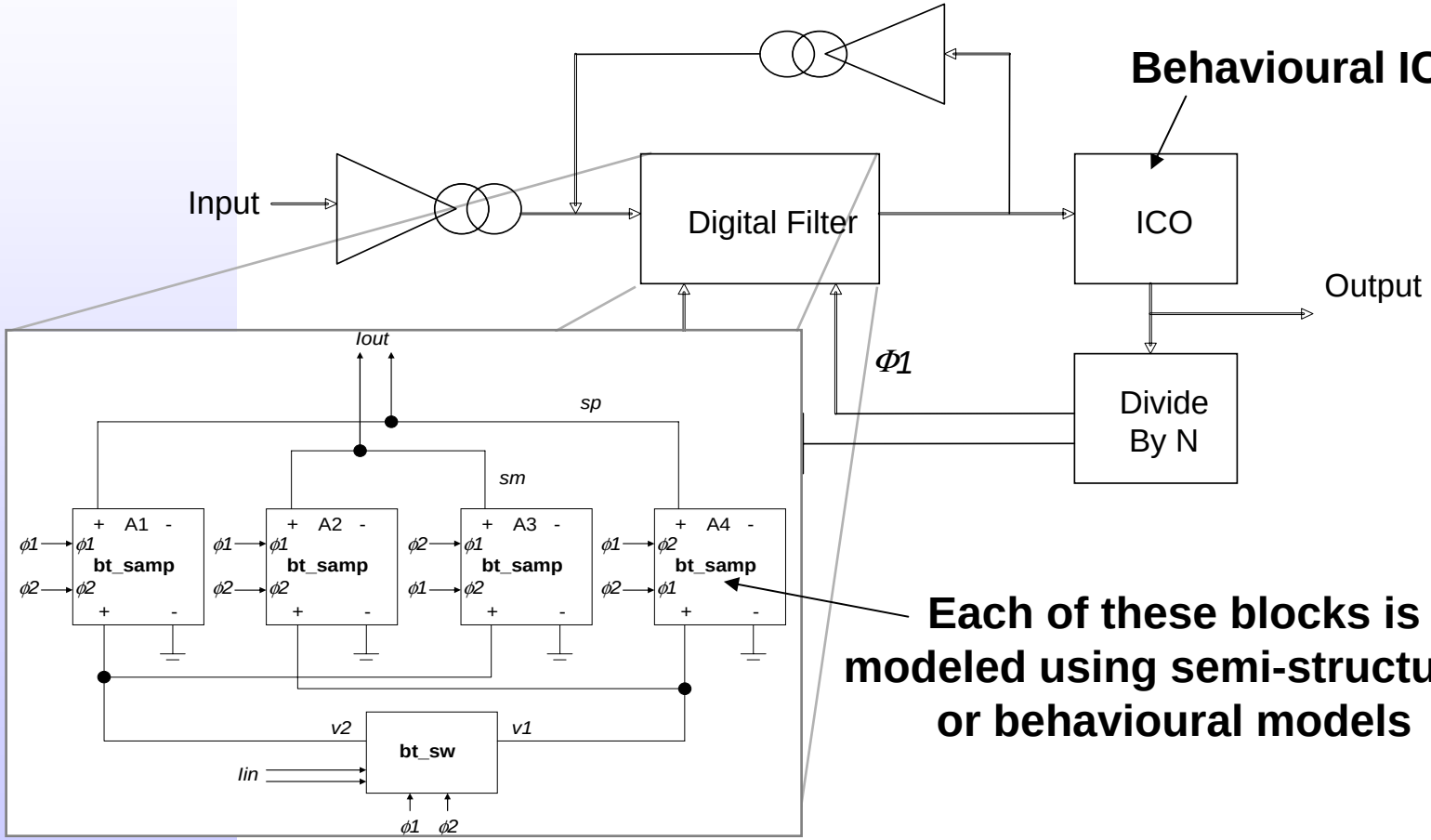
- Details of the PLL design are given in recent papers – Wilcock, Wilson and Al-Hashimi, ISCAS 05 & Wilson and Wilcock, ISCAS 05
- Novel, implicit phase detection – hence no PD block

The basic governing parameters of the PLL are

- switching element gain (a – default = 1.0)
- current feedback gain (b – default = 0.3).



SI PLL – using bilinear transform structure

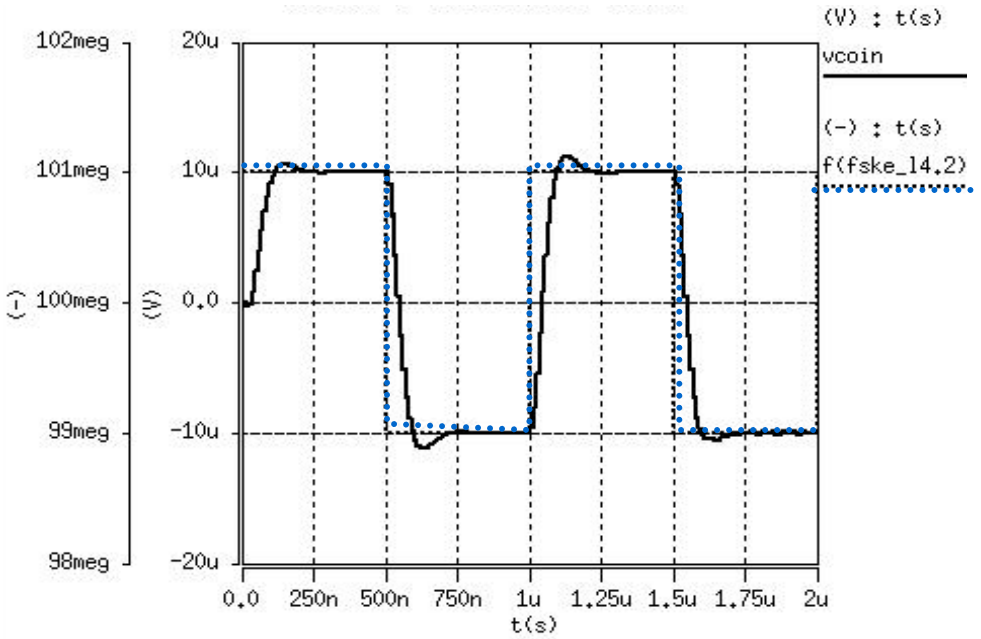


Each of these blocks is modeled using semi-structural or behavioural models

Basic Simulation

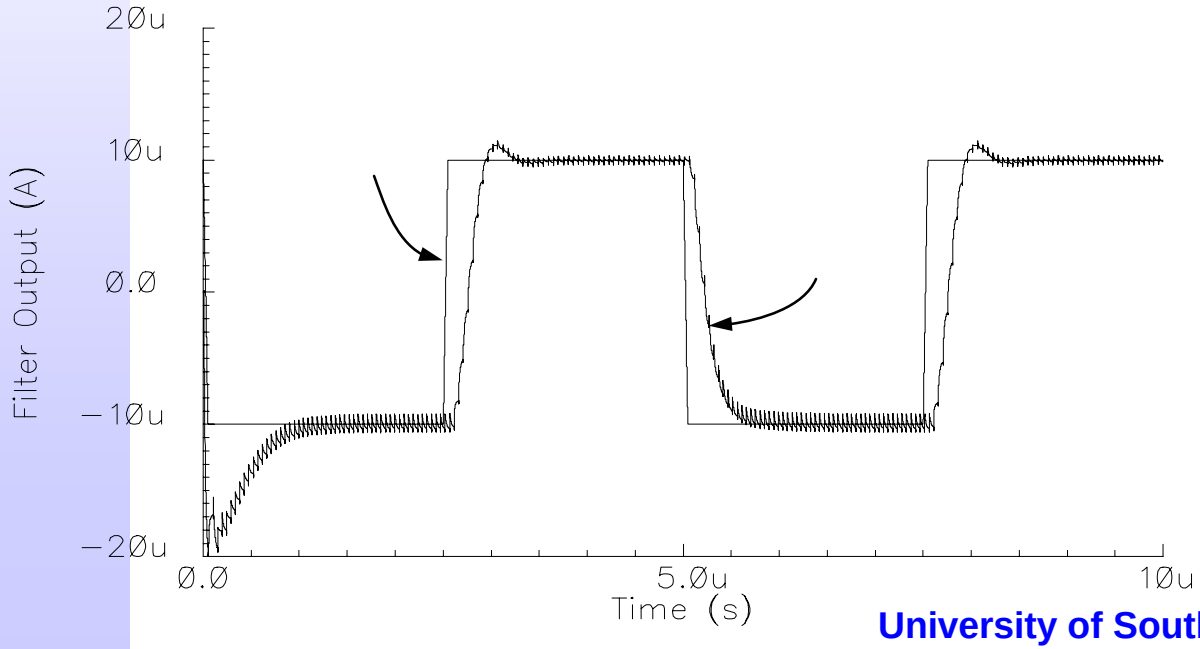
Using this behavioural approach, a simulation with the initial nominal parameters was carried out

The input was an FSK signal varying between 99MHz and 101MHz



Transistor Level Simulation

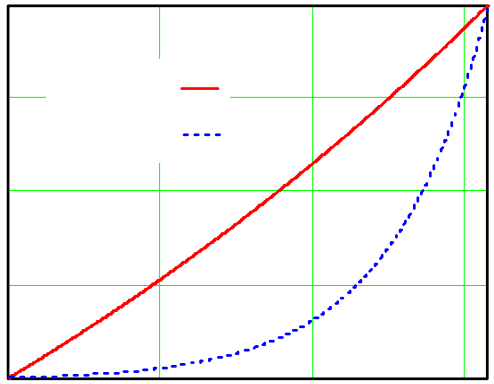
With the same basic architecture a transistor level simulation was carried out using transistor level models with the results shown below. The waveforms show the same basic loop response for the same parameters



Note on the response

You may have noticed that the acquisition is asymmetric – this is due to the non-linear phase detector gain, which is a trade off against stability hence the need for behavioural models to investigate this design space.

The PD gain is dependant on input frequency and hence the PLL locks under damped to higher frequency and vice versa



Simulation Comparison

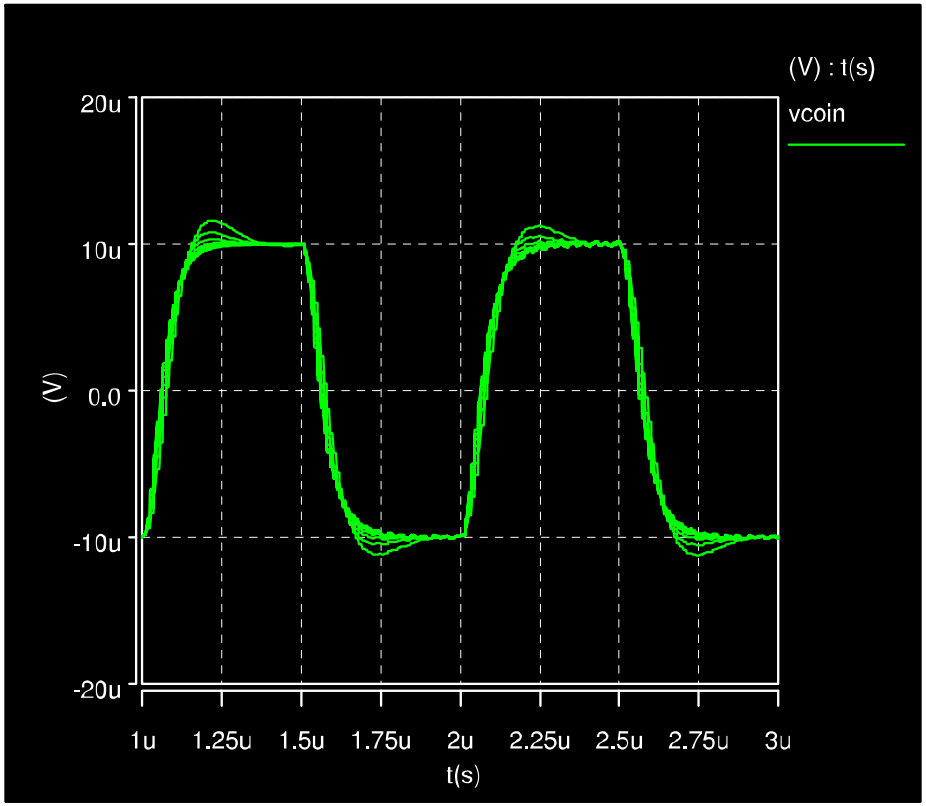
The Transistor level model took 1 hour to run a single cycle – at a frequency of 10 times less. For an equivalent frequency it would have taken around 10 hours for a single complete cycle. This was running on a Sparc Ultra 10. Even using a Linux simulation server we estimate simulation times of many minutes (perhaps a best of 20 minutes).

The switching behavioural model took 1.57 seconds for the same simulation at full 100MHz rate

"What If" analysis

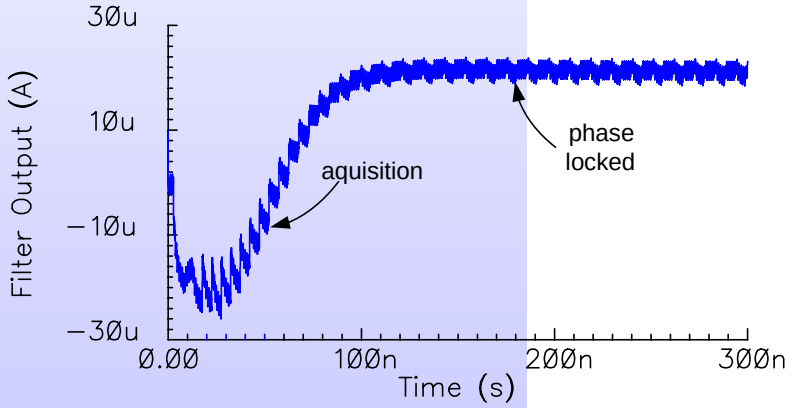
The vastly improved simulation times allow us to carry out variation in parameters to investigate the performance envelope of the design quickly.

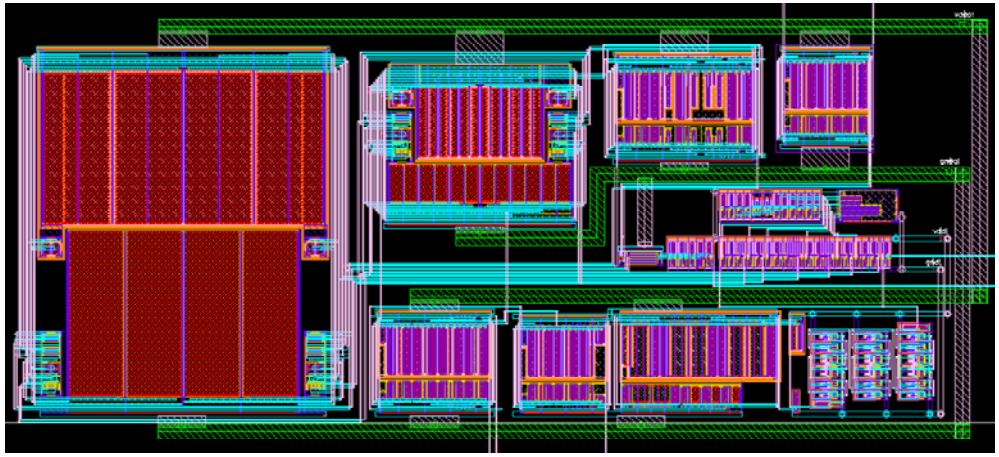
A critical parameter is the loop feedback gain, and by varying this we can obtain the optimal parameter value for our application



120nm Silicon so far…

A fully integrated 100MHz prototype chip is currently being fabricated on a 120nm digital process (ST). The chip has a supply voltage of 1V, power consumption of 5.9mW and an area of only 0.017mm^2 which is around 5% the size of a similar design based on a passive loop filter.





School of Electronics and Computer Science

University of Southampton, UK

Conclusions

We have used our design flow to successfully design a serious PLL using SI targeted at a standard Digital DSM process in less than 3 months.

Behavioural modeling was an integral part of the design process and vital to its success.

Design techniques to address low VDD issues are being developed as we learn about the process issues.

Next Silicon back in November 2005 at which time we can update the design community with results.

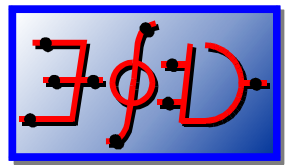
Final Remarks

A note by the design team is that it is often stated that languages and tools are effectively "for" a specific application - this is not always the case.

This example highlights the fact that sometimes tools can be applied extremely effectively in areas where perhaps the vendor does not envisage or encourage their use.

The use of behavioural modeling is becoming even more critical as a design tool for AMS IC designs.

The very simplicity and accessibility of MAST makes it an attractive choice for mixed signal and mixed level integrated circuit design – *for designers*.



Acknowledgments

EPSRC for funding the work (http://www.epsrc.ac.uk)

CMP for their excellent support (http://cmp.imag.fr)

My excellent design team! (http://www.esd.ecs.soton.ac.uk)