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

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

School of Electronics and Computer Science, University of Southampton, Southampton, UK
prw@ecs.soton.ac.uk & rw3@ecs.soton.ac.uk

Abstract- Recent work has shown that the use of switched current methods can provide an effective route to implementation of analog IC functionality using a standard digital CMOS process. Further recent advances indicate that adopting a switched current architecture can lead to equivalent performance but with a significantly reduced area compared to switched capacitor structures. While this is useful in itself, the use of behavioural modeling and simulation at a structural and building block level has allowed architectural exploration and evaluation to be carried out on novel topologies based on this approach. The result is an integrated design flow that uses behavioural models to test the performance of the circuit, leading directly to a synthesized structural model that can be verified using a common design platform. This has the obvious benefit of reducing the full custom analog design effort required when developing topologies and building blocks for new processes.

In this paper we describe the design approach of a Phase Locked Loop (PLL) based on a novel Switched-Current (SI) architecture using behavioural models written in MAST. Simulations demonstrate the performance of the design at a high level and are used to optimize the behaviour of the loop response with regard to design specifications. The modeling approach is explained, from the basic building blocks used in the SI methodology through to more abstract models of the system architecture. The advantage of using behavioural models is highlighted with the ability to carry out parametric analysis and statistical analysis to investigate tolerances, adherence to specifications and parameter variations. The resulting simulations are consistent with transistor level simulation results, but several orders of magnitude faster. The resulting design achieves a performance that is comparable with designs using current techniques but with significantly reduced area.

1 Introduction

"Switched currents" is a current-mode sampled analog data technique that produces circuits that consist entirely of transistors and hence are compatible with a standard digital CMOS technology [1]. It has been predicted in [2] that SI circuits have the potential to surpass the performance of their switched-capacitor (SC) equivalents. This indicates that SI will play an important role in the design of future mixed-signal ICs, and therefore merits further research in SI based analogue IC design. PLLs are fundamental building blocks, employed in numerous applications ranging from clock recovery, frequency shift keying, to demodulation and frequency synthesis. Despite the potential of SI and the importance of PLLs in mixed-signal ICs, there has been no reported work in the area of SI based PLLs, apart from [3] which proposes a first order, 1MHz all-digital PLL architecture. The main contribution of the work in [3] is that it shows the possibility to design simple PLL circuits using transistors alone and hence without the need for linear capacitors. However, most current and future practical PLL applications require 2nd order architectures with higher frequencies of operation than the PLL reported in [3]. The details of the implementation in this approach are given in the paper to be published at ISCAS 2005 [8] and previously in [5-7].

In general, much of the work into PLL modeling has been at an abstract level that does not explicitly model the lower level IC functions, but rather the macroscopic behaviour of the design elements at a high level [9-11]. While this is useful and valid, in *this* context it was important for the IC designer to be able to not only simulate the overall response of the circuit, but particularly to evaluate the cycle by cycle response of the design, without necessarily modeling all the detail of each individual transistor. As a result the modeling approach taken is structural, without using detailed and complex MOS models of individual devices. The consequence of this is a detailed, switch level simulation, but using behavioural switch models and simplified model structures instead of characterized transistors implemented exactly as in the IC layout.

0-7803-9352-X/05/$20.00 © 2005 IEEE.

2 SI PLL modeling

The proposed 2^{nd} order SI PLL has a similar architecture to that of a conventional PLL, but without the need for a phase detector (PD). Due to the sampled nature of the SI technique, it is possible to perform phase detection as part of the loop filter (LF) operation, and we refer to this as ‘implicit phase detection’. This has a number of advantages over a conventional PLL, not least the ability to control the behaviour of the PLL as a whole by modulating the external switch control signals and also the simplified architecture of the PLL design itself. With most of the other main blocks of the PLL well understood, the implicit phase detector will be elaborated further in this paper.

2.1 Implicit phase detector

The principle of implicit phase detection is best illustrated through a voltage mode example of two cascaded voltage mode sample and hold cells shown in Figure 1. The clocks ϕ_1 and ϕ_2 are non-overlapping, and the waveforms of the circuit are shown in Figure 2(a) and (b) for the case where V_{in} is a square wave voltage signal, with the same frequency as ϕ_1 and ϕ_2, but with a positive and negative phase error, θ_e, respectively.

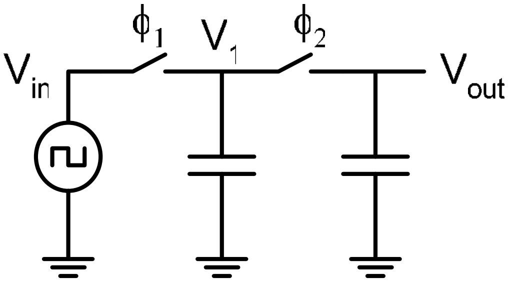

Figure 1. Voltage mode sample and hold

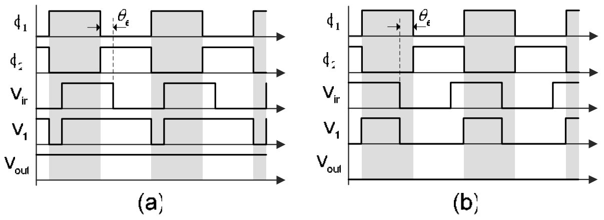

Figure 2. Sample and hold waveforms

2.2 Basic Switched Current Cell Behavioural Modeling

A SI memory cell is the current mode equivalent of Figure 1 and its operation as an implicit PD is now considered. There are a variety of techniques to model this kind of cell with the idea of incorporating non-ideal effects that improve the accuracy of the simulation. Behavioural models were developed of the basic memory cell using fast behavioural switch models to replace the transistor models allowing much faster simulations, but retaining some of the intrinsic switching behaviour of the design. The schematic of this type of memory cell model is shown in figure 3 and the basic memory cell model (without sampled output) written in MAST is shown in figure 4. The majority of the model infrastructure has been ignored at this stage, to highlight the simplicity of the modeling approach in this context.

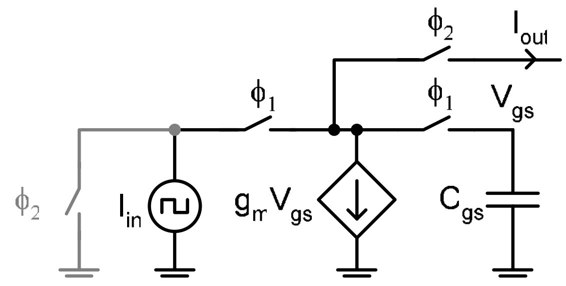

Figure 3. SI memory cell circuit model

```
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
}
```

Figure 4. MAST behavioural switch model

The advantage of using MAST is that we can sample the input voltage (modeling the effect of the input switch). This can then be used in the characteristic equation as shown to compactly model the output switches. The three sections of the model illustrate the function of the switch. The **values** section shows the **if** statement that selects whether the current is to be taken from the input and charge up the capacitor, or when the switch is opened, then the value of **vgs** that was sampled is held across the capacitor. The **when** section is used to monitor the state of the phase control (**phi**), and also to sample the **vgs** voltage. This works reasonably well, but there can be convergence issues with the change in state and current sources that can actually make the simulations longer than necessary when used in a complex configuration (on its own this circuit simulates extraordinarily fast). To combat the possibility of convergence difficulties when used in a larger circuit, a semi-structural behavioural model can be used that has “soft” switches using a continuous transition between states to ensure a smooth transient simulation. We could obviously add this in the equations for the ideal model, but hierarchy is a powerful way of implementing these advanced functions once, and not having to repeat for multiple instances. The resulting model is shown in figure 5. Notice that we are still using digital events to control the switches in the design at this stage. This has the obvious benefit of reducing the analogue simulation complexity and hence providing greater efficiency and speed overall, compared with using a “spice-like” purely analogue switch. This can work well in this context where the threshold voltage is assumed to not rely on, or be affected by hysteresis (i.e. different transition threshold voltages for rising and falling edges).

```
# input gm stage
vccs.1 vgs 0 inp inm=gm

# switch cap stage
sw_l4.1 inp vgs phi=ron=ron,roff=roff
c.c1 vgs 0 = c

# output gain stage
vccs.2 vgs 0 outp outm = a*g
```

Figure 5. MAST semi-structural sampled switch model

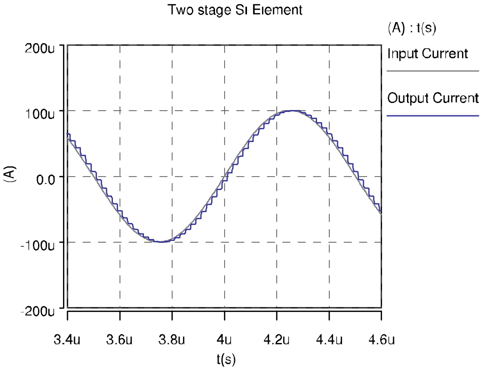

Figure 6: Sampling Switched Current memory cell

Using this approach the basic switched current cells can be tested and used to validate the basic operation against transistor level models – for example testing the basic memory cell takes seconds to produce the results shown in figure 6.

2.3 Basic Integrator Model

At the heart of the new PLL design is the use of a bilinear integrator element. We can build integrator elements easily using the new Switched Current (SI) building block just devised. If we connect two switch elements in series, and feedback the output current into the input stage, then we get a basic integrator. This is shown in figure 7.

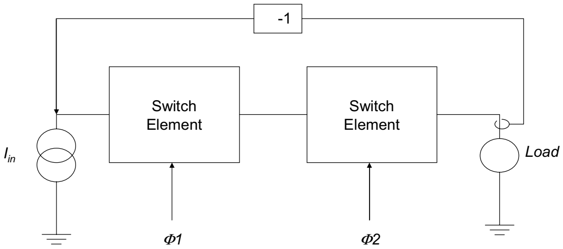

Figure 7: Basic Integrator Model

If we simulate this using a steady state input current of 5µA, and ensure that the two clocks *Φ1* and *Φ2* are non-overlapping, we will get an integration function that will step up the output current in 5µA steps at each stage. The resulting output current is shown in figure 8, and you can see both the overall integration function, and also the detail switching behaviour at each stage, this is dependent on the capacitance and switch characteristics within the switching element model, and allows detailed specification of the physical layout on the eventual IC design. This integrator is switching at 50MHz frequency (20ns period).

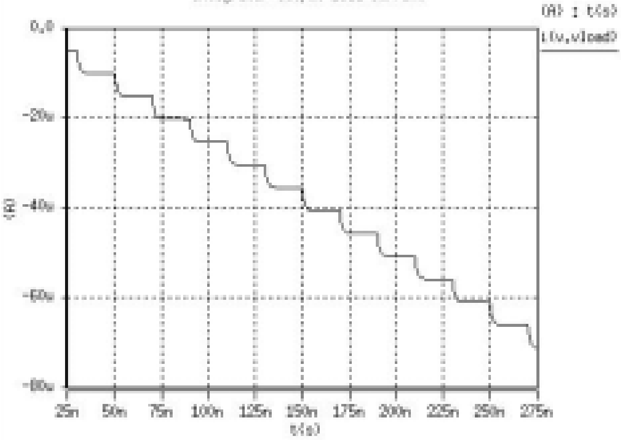

Figure 8: SI Integrator Output Current

2.4 Bilinear Integrator

With the basic integrator structure we can create arbitrary filter functions, but the direct transformation from Laplace to Discrete functions still leaves a significant effort to map from one domain to another. One way of mapping directly from the S domain is to use a bilinear integrator structure – this allows a data flow approach to be used that can implement filter functions directly. In order to facilitate this, a bilinear integrator structure was developed that can be implemented directly in the switched current technology, and the resulting MAST model is given in figure 9.

```
template  bilin_samp  inp  inm  outp  outm
outsampled phi1 phi2 =c,gm,ron,roff,a
electrical inp,inm,outp,outm,outsampled
state logic_4 phi1,phi2
number a = 1
number c = 100f
number gm = 100u
number ron=1k
number roff=100meg
{
electrical vgs2,vgssampled
val v vgs_samp, vgs_sampout
state nu isamp= 0
when(event_on(phi2)) {

schedule_event(time,isamp,vgs_samp)
}

values{
  vgs_samp = v(vgs2)
  #v(vgssampled) = isamp
}

# input gm stage
vccs.1 vgs 0 inp inm=gm

# switch cap stage
sw_l4.1 inp vgs phi1=ron=ron,roff=roff
c.c1 vgs 0 = c

# output gain stage
vccs.2 vgs 0 outp outm = a*gm

# sampled buffer
vcvs.1 vgs 0 vgs2 0=1

# switch cap stage
sw_l4.2 vgs2  vgssampled
phi2=ron=ron,roff=roff
c.c2 vgssampled 0 = c

# final output gain stage
vccs.3 vgssampled 0 outsampled 0= a*gm
#r.fr vgssampled 0=1k
}
```

Figure 9: Bilinear Integrator Switched Current MAST Model

One feature of this approach is that as well as the switching stages highlighted previously in the basic switching element models, there is also a sampling of the output gain stage as well.

3 Switched Current PLL Model

3.1 Digital Filter Model

It is beyond the scope of this paper to discuss the details of the PLL implementation in detail, but the basic switched current building blocks can be combined easily to create a sampled, semi-structural behavioural model in Saber allowing extremely fast simulations investigating the key parameters of the design (PD Gain K_d) and their effect on the overall system response. The PLL LF can be designed using well documented SI techniques [4], but that is beyond the scope of this paper. The core of the new PLL architecture [8] uses the basic switched current bilinear integrator building block shown previously and applied in the configuration shown in figures 10 & 11. If the reader is interested in the fundamental design concepts then they are referred to [8] for further details.

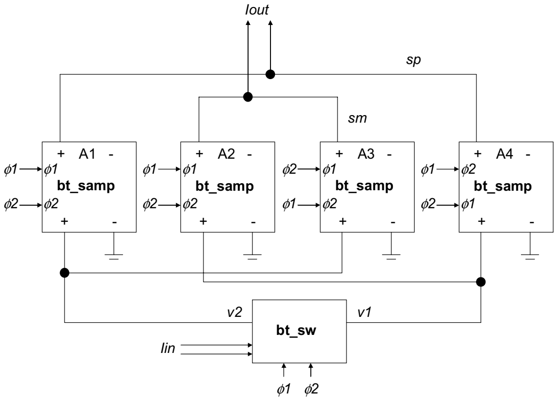

Figure 10: Block Structure of the proposed behavioural model

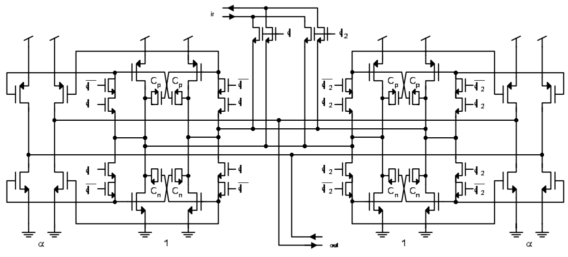

Figure 11: Transistor Level implementation of two-phase sampled filter

3.2 Basic PLL Architecture

The key in the new architecture is implicit phase detection. The resulting PLL structure therefore looks somewhat unusual to the normal approach and a simplified version is shown in figure 12.

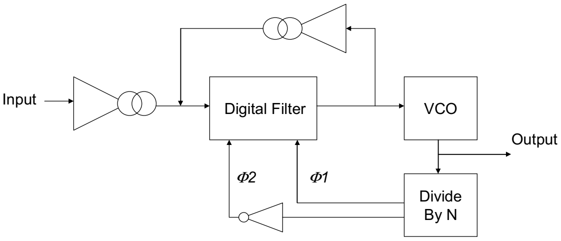

Figure 12: SI PLL Architecture

3.3 SI PLL Basic Simulation

If we use an FSK input and switch between two different frequencies, we can test the behaviour of the proposed PLL architecture easily. Carrying this out, we can also ensure that the behavioural model exhibits consistent behaviour to that of a transistor level model. The reason for doing this is calibration of the model. While we do not intend to carry out exhaustive simulations using the transistor model, it is important at all stages to ensure the integrity of the models so that we can rely on the simulations and the decisions we take regarding design parameters. The difference with the conventional simulation approach is clearly that we can carry out many more simulations using the behavioural model in contrast to the transistor level model. Simulating a basic transistor level model in Spectre we get the following output behaviour on the filter, as shown in figure 13.

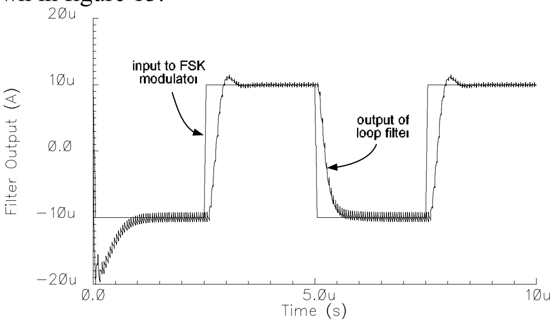

Figure 13: Transistor level simulation of filter response

An immediate thing to notice is the difficulty of obtaining information other than voltages and currents from transistor level simulations in Spectre. Post-processing and signal analysis is generally more difficult than in Scope, and we can take further advantage of these features in analyzing out design performance in Saber. This particular analysis took about an hour to execute a single simulation of the PLL using transistor level models. In addition this design we have used an extra division cycle to reduce the frequency from 100MHz (nominal) to 10MHz (nominal), to again reduce the simulation requirements. In the Saber model we can keep the true 100MHz output frequency and run a simulation to validate the basic loop performance. The resulting stimulus (input to FSK) and digital filter response can be seen in figure 14. The basic parameters to the PLL that govern this response are the switching element gain (a – default = 1.0) and the current feedback gain (b – default = 0.3). These parameters can be altered to provide the loop response required, giving a flexibility and controllability that is useful and not always available in a conventional PLL design. It is interesting to note that even including switching models, the corresponding simulation time was 1.57s. This was carried out using a basic low-cost PC (2.6GHz) running Linux – there being no special performance enhancements inherent in the hardware.

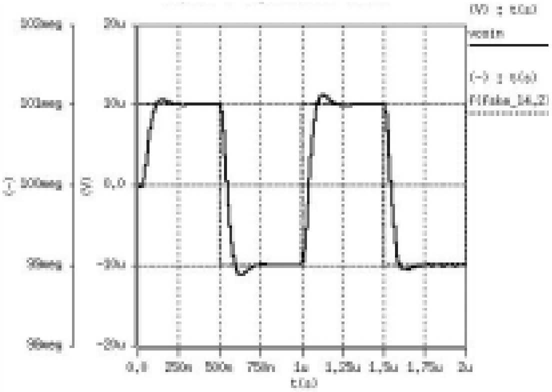

Figure 14: Basic PLL response (a=1 and b=0.3)

3.4 Design Exploration

What we can show is how after that initial design phase, multiple "what-if" style analyses can be carried out to investigate the behaviour of the design quickly using these models, with a performance that is just not possible with transistor level descriptions. As an example, the initial design provided a range of loop parameters, and simulations were carried out to evaluate the optimal loop response while varying the loop parameters. The resulting waveforms are shown in figure 15. This shows the measured VCO frequency while varying the loop parameters, with the same basic FSK stimulus.

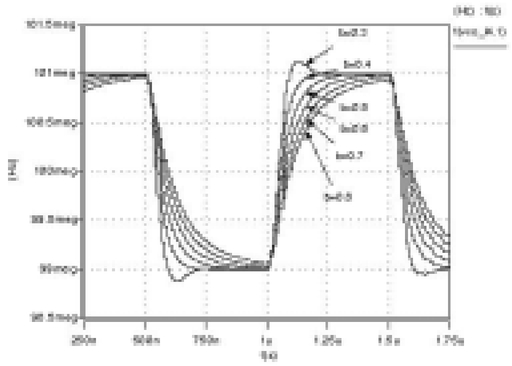

Figure 15: Response of Behavioural PLL to FSK input

To evaluate the effectiveness of the optimization (albeit manual at this stage), the resulting parameters chosen were fed back into a transistor level simulation to verify that the response of the detailed circuit model was consistent with the designed behaviour.

A key factor to notice in these simulations is that the behavioural simulations (even for a pseudo switch implementation) took a few seconds, whereas the transistor level simulations took roughly one hour per switching cycle. This illustrates graphically that this approach would not have been feasible without the use of behavioural models able to provide the designer with feedback in a short time, with a reasonable accuracy.

4 Conclusions

The paper has shown how behavioural modeling can be used at a variety of levels to model detailed behaviour of switched current blocks for use in chip architecture and the analysis of complex circuit elements such as phase locked loops.

The simulation results are consistent with those of a transistor level implementation, but within a fraction of the simulation time, even for a liberal setting. The benefits for IC designers of such as approach using MAST, VHDL-AMS or a similar modeling language such as Verilog-AMS are obvious and this paper provides further demonstration of the usefulness and effectiveness of behavioural modeling to support the IC design process.

The choice of MAST in this design was taken for several reasons, and in fact the work has been repeated using VHDL-AMS to enable a comparison of the approaches. The design team has noted that the time required to create the models and run simulations using MAST was significantly less than that required for VHDL-AMS. The conclusion was that MAST is simpler, faster to debug in Saber, and easier to relate directly to results than other approaches. While the design team accept that there are instances when it may well be more appropriate to use VHDL-AMS or Verilog-AMS – particularly where there is a larger digital component in the design for example – in designs with a significant analogue content, MAST is tailor made for mixed structural and behavioural simulation and modeling.

5 Final Note

Various implementations of the PLL described in this paper have been implemented in 0.12um CMOS and it is hoped that results of the fabricated chip will be available to be presented at the conference itself.

6 References

[1] C. Toumazou, J. B. Hughes, and N. C. Battersby, Switched-currents: an analogue technique for digital technology: *Peter Peregrinus Ltd.*, 1993.

[2] J. B. Hughes, A. Worapishet, and C. Toumazou, "Switched-Capacitors Versus Switched-Currents: A Theoretical Comparison," *presented at IEEE International Symposium on Circuits and Systems*, vol. 2, pp 409-412, May 2000.

[3] D. M. W. Leenaerts, G. G. Persoon, and B. M. Putter, "CMOS switched current phase-locked loop," *IEE Proceedings - Circuit Devices and Systems*, vol. 144, pp. 75-77, April 1997.

[4] J. B. Hughes, "Top-down Design of a Switched-Current Video Filter," *IEE Proceedings - Circuit Devices and Systems*, vol. 147, pp. 73, Feburary 2000.

[5] Y. Sun, Design of high frequency integrated analogue filters: *IEE Press*, 2002.

[6] D. J. Allstot, G. Liang, and H. C. Yang, "Current-mode logic techniques for CMOS mixed-mode ASICs," *presented at IEEE Custom Integrated Circuits Conference*, vol. 2, pp. 1-4, May 1991.

[7] Barcelona Design, http://www.barcelonadesign.com, 2004

[8] R. Wilcock, P.R. Wilson and B.M. Al-Hashimi, “A Novel Switched-Current Phase Locked Loop”, *ISCAS 2005*

[9] Kozak, M.; Friedman, E.G.,”Design and simulation of Fractional-N PLL frequency synthesizers”, *ISCAS '04*, 2004, pp:IV - 780-3 Vol.4

[10] Karray, M.; Seon, J.K.; Charlot, J.-J.; Nasmoudi, N., “VHDL-AMS modeling of a new PLL with an inverse sine phase detector (ISPD PLL)”, *BMAS 2002*, pp:80 – 83

[11] Abdennadher, S.,”Flow for phase locked loop mixed signal simulation and characterization using behavioral modeling”, *Southwest Symposium on Mixed-Signal Design*, 2003, pp:66 - 70