

# Design, mapping, and simulations of a 3G WCDMA/FDD basestation using network on chip

Daniel Wiklund and Dake Liu\*  
Dept. of Electrical Engineering  
Linköping University  
S-581 83 Linköping, Sweden  
{danwi,dake}@isy.liu.se

## Abstract

*This paper presents a case study of a single-chip 3G WCDMA/FDD basestation implementation based on a circuit-switched network on chip. As the amount of transistors on a chip continues to increase, so does the possibility to integrate more functionality onto every chip. By combining general-purpose and application-specific hardware, it is possible to integrate the complete baseband part of a 3G basestation on a single chip. Such a single-chip basestation has been modeled from a communication perspective without full implementations of the processing elements. The system has been scheduled and implemented as a traffic model for a network on chip simulator. Simulation results show perfect adherence to the schedule already at a network clock frequency of 75 MHz. The overall network usage is relatively low except for the area closest to the radio interfaces. This will allow for other messages, e.g. control related, to be transported over the network during the gaps in the communication schedule.*

*Keywords: Network on chip, 3G, WCDMA, basestation, scheduling*

## 1 Introduction

Currently, the third generation of mobile telephone systems is deployed throughout the world. The standard used in Europe is the WCDMA/FDD mode of the 3GPP specification [1]. WCDMA (wideband code-division multiple access) means that different users share a relatively wide spectral band using coding instead of time slots. Each user is allocated a spreading sequence used to smear its narrow-band data signal over the broader spectral band. Each user is differentiated from other users by the chosen spreading sequence which preferably should be orthogonal to the other

spreading sequences in use. The narrow-band signal can then be recovered at the receiver by the same mechanism. Hobson et al. have a good, short introduction to the processing necessary for 3G reception [2].

The amount of computation that has to be performed in the basestation will require the use of special-purpose hardware for several of the stages. Communication between the subsystems is not trivial and is the focus of this paper.

Lately there has been a move in usage of networks from communication between systems to communication within systems. Extending this trend further there is an obvious possibility to use networks on chip. The work presented in this paper is a case study of a basestation that is based on the SoCBUS network on chip for the internal communication. The design goal for the SoCBUS network has been to give good performance at a low cost for this type of regular signal-processing systems.

Section 2 describes the high-level design specification. The SoCBUS network on chip is described in section 3. Section 4 introduces the basestation processing flow. Sections 5 and 6 describes the mapping and subsystems, respectively. Section 7 discusses system scheduling. Section 8 shows some simulation results and section 9 concludes the paper.

## 2 High-level design specification

The case study presented in this paper concerns the baseband part of a 128-channel radio basestation for WCDMA/FDD. The case study is limited to a specific traffic case in the basestation with the basic assumption that 128 data channels each transporting 384 kbit/s is used in both uplink and downlink. No regard is given to changes in this situation that would occur if a new user connects or a channel changes transfer parameters.

The implementation is based on a SoCBUS network of appropriate size. This network is used for all communication between subsystems.

---

\*This work is supported by the Swedish Foundation for Strategic Research (SSF) through the Stringent electronics research center at Linköping University.

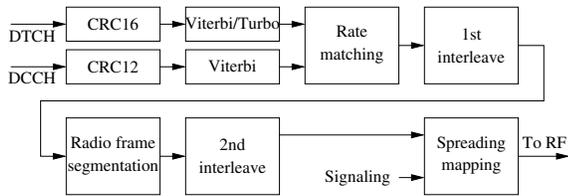


Figure 1. Downlink transmission flow

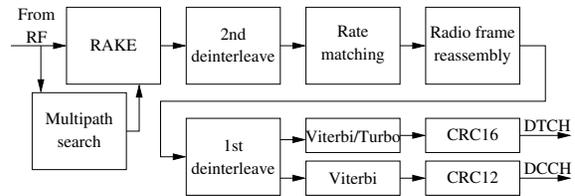


Figure 2. Uplink reception flow

### 3 SoCBUS network

The network on chip considered in this paper is the result of the SoCBUS project at Linköping University. The network uses a hybrid packet/circuit switching method known as packet connected circuit (PCC) [3]. This approach uses a small routing packet that traverses the network and sets up a circuit switched route for the bulk payload data to follow. By using the PCC concept, both the need for buffers and the resulting latency penalty are eliminated. The drawback is that a failed routing will cause a retry which may lead to increased congestion.

The network components are connected through dual unidirectional links based on a 16 bit data bus. The two links form a bidirectional connection between the components.

### 4 Basestation processing flow

#### 4.1 Downlink

The downlink processing is relatively straight forward. The basestation will transmit a set of downlink physical channels, typically one for each terminal, plus a set of common channels. The common channels are three synchronization channels, a broadcast channel, a paging channel, and so on. All these common channels are transmitted to all users within the reach of the basestation. Of special interest among these are the synchronization channels that are used for initial cell search where a mobile terminal can find and connect to a basestation [4].

Figure 1 shows the data flow for downlink user data processing. Incoming data from the media access control (MAC) layer consists of two different streams. One is the dedicated transport channel (DTCH) for data and the other is the dedicated control channel (DCCH). First, CRC and forward error correction coding is added to the streams. These are then sent through a rate matcher that will make sure that the data rate of the stream is correct for the physical layer. The stream is then interleaved, segmented into slots, and interleaved again. Finally, the stream will be mapped and spread to the chip rate and output to the radio frontend.

Table 1. Block types

Abbr	Subsystem function
RI	Radio interface
SM	Spreading and modulation
Rake	Rake receiver and despreader
IRM	Interleaver and rate matcher
RFRI	Radio frame reassembly and interleaver
VTC	Viterbi/Turbo encoder
VTD	Viterbi/Turbo decoder
CRC	CRC calculator and checker
Mac	Mac layer interface

#### 4.2 Uplink

The uplink processing flow is similar to the downlink flow but involves much more computation. Figure 2 shows the processing flow for the uplink reception. First in the flow is multipath combination, which is based on a multipath search filter and a Rake receiver in cooperation. Since each terminal will experience different multipath propagation conditions, this will require the use of one multipath searcher and combiner per terminal in the basestation. The Rake receiver will sum the multipaths and despread the incoming signal.

The recovered stream is then sent through deinterleaving followed by reverse rate matching. It is further sent through radio frame reassembly and a second deinterleaver before the forward error correction coding is used to restore the received data, which is then sent to the MAC layer.

### 5 Function mapping

The basestation data flow implementation is based on a set of functional subsystems that are listed in table 1. Each subsystem performs a part of the processing flow. Some functionality has been grouped, e.g. the rate matcher and an interleaver. This is possible since these functions are each very simple and a net gain in cost can be found when the functions share memory. The subsystem selection is based on an analysis of the processing and memory requirements of the basestation and is beyond the scope of this paper.

The architectural mapping is found in figure 3. The ar-

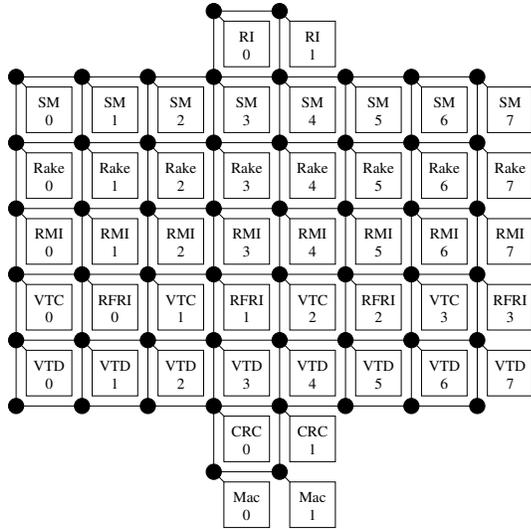


Figure 3. Architectural mapping of the basestation

chitecture is based on a mesh network with 8x8 routers. The network has been optimized somewhat to get a lower overall cost with only 52 routers. Further optimizations are possible since the top and bottom rows (with two routers each) are superfluous for the functionality since the blocks can be attached to the nearest full row. Also, some of the links in other parts of the network are unused and can be removed from the design with the current allocation and schedule. An optimal network is not the main design target and this is therefore ignored.

The network considered uses only the simplest router version implemented within the SoCBUS project. This is simply because the extra functionality of the latter, more complicated routers are not necessary for the application. In the final network, some routers have less than five ports which will give a smaller contribution to the total area than  $0.06 \text{ mm}^2$  per router [5]. Thus, with the selected network topology, the routers will occupy a chip area of less than  $52 \cdot 0.06 = 3.12 \text{ mm}^2$  plus wiring.

The functional allocation follows the data flow with the radio interfaces on the top edge of the network to the MAC interfaces at the bottom edge. Uplink data goes from top to bottom, and vice versa for the downlink data. The allocation of subsystems to network ports is done to minimize the network distance between communicating subsystems.

A real basestation would need a system controller to be present. This is not included in the described allocation but can be connected more or less anywhere in the network.

## 6 Processing subsystems

Most steps in the processing flows have relatively low computing complexity. An example is (de-)interleaving of

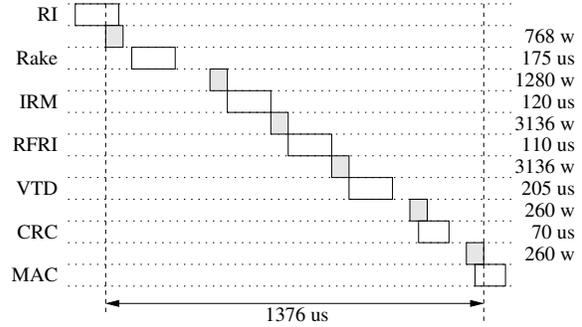


Figure 4. Worst-case uplink processing schedule (not to scale)

data, which is a trivial operation in the 3G standard. Data is simply written in order to a memory and then read from the memory using a permutation of row and column order.

The multipath search is by far the most processing intensive task in the flow with the longest path delay (from the ITU pedestrian B model) being  $3.7 \mu\text{s}$ , or about 60 samples. After the multipath search is complete, Rake combining can then be done using simple techniques, e.g. the FlexRake architecture [6].

The Turbo decoding is the second most processing-intensive task in the basestation. Data is encoded with rate 1/3 using two 8-state constituent encoders. The total incoming data rate to each of the eight Turbo decoder subsystems is about 6.2 Mbit/s. This target can be reached using ASIC-style hardware [7].

## 7 Processing and communication scheduling

The maximum latency for processing a 3G slot is on the order of a few slot times. A maximum latency of two slot times ( $1333 \mu\text{s}$ ) is assumed for the schedule in this case study. Therefore, the communication and processing schedule must be relatively tight to meet the deadline.

The downlink processing is practically trivial compared to the uplink processing and will not be discussed further here although it has been considered in the scheduling of the system. The uplink is more interesting since all heavy processing is done in that part. A nice property of the 3G system is that the received uplink channels must be (almost) in synchronization with each other, simplifying the task of handling several simultaneous channels.

There are four distinct subschedules for the receiver depending on which specific instances of the subsystems that are involved in the processing. Figure 4 shows the uplink processing schedule with the worst completion time. The SoCBUS transfers are shown as grey boxes and the processing as white boxes. The annotations are the transfer size in words (i.e. multiples of 16 bits) and maximum pro-

**Table 2. Longest possible processing times in the reception flow**

Abbr	Subsystem	Max time
RI	Radio interface	n/a
Rake	Rake receiver and despreader	175 $\mu$ s
IRM	Interleaver and rate matcher	120 $\mu$ s
RFRI	Radio frame reasm+interleave	110 $\mu$ s
VTD	Viterbi/Turbo decoder	205 $\mu$ s
CRC	CRC calculator and checker	70 $\mu$ s
Mac	Mac layer interface	n/a

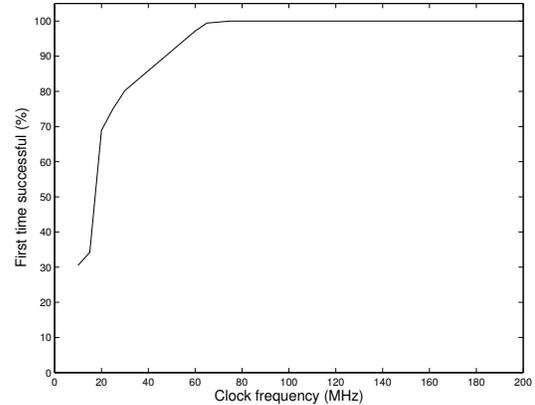
cessing time in microseconds. The gaps in the schedule are due to synchronization delays between the communicating subsystems and the different schedules for these including the downlink processing. This schedule shows a fourth of the processing for a single time slot and is closely related to the other three (shorter) schedules.

The critical chain of communication in total is 8840 words plus overhead, which gives less than 10000 network clock cycles for the communication. With a network running at 100 MHz this will give a total delay on the order of 100  $\mu$ s. Simulation results later showed that the minimum acceptable network frequency is 75 MHz and that the maximum overhead is 41 cycles per communication which yields about 122  $\mu$ s total communication delay. With maximum two time slot delays in the basestation, this will give about 1200  $\mu$ s for processing assuming no flow synchronization overhead.

## 7.1 Schedule analysis

The first step in the schedule is to send a set of samples from the radio interfaces to the Rake/despreader blocks every 50  $\mu$ s. With an oversampling factor of four, this will be 768 samples per period. The reason for selecting a period of 50  $\mu$ s is that the radio interface will require significantly less memory and that the Rake/despreader blocks can work continuously during reception of a slot without getting excessively large sets of samples at long intervals. Also the radio interfaces need to send the samples to four Rake blocks which would add about 400  $\mu$ s to the latency for the last receiving Rake block if a full slot would be buffered. Selecting a smaller time than 50  $\mu$ s would not give any significant savings.

The schedule continues with the rest of the processing subsystems and their associated communication. The maximum allowable processing time for each block in the schedule is given in table 2. The shortest of the four schedules is approx 907  $\mu$ s long while the longest schedule is 1376  $\mu$ s.



**Figure 5. First try successful routing**

## 7.2 Latency-induced storage

Extra memory requirements are added due to the additional latency incurred from the schedule. This latency is not part of the network latency but rather related to the processing schedule for the subsystems. Whenever a subsystem is occupied by computation, it is also considered unable to receive incoming data.

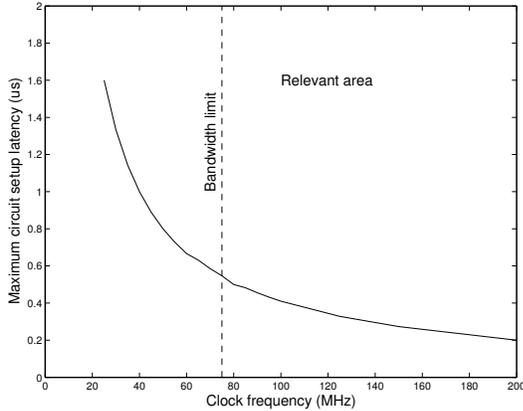
The maximum extra buffer memory requirements consist of a single buffer for each communication, in total about 100 kilobytes for all reception flows. This requirement can be relaxed somewhat by tweaking the schedule to get tighter communication. With the simulated schedule this has been lowered to about 25 kilobytes in total. The extra buffer memory can either be placed within the subsystems or in the wrappers.

## 8 Simulation results

### 8.1 Minimum network frequency

The basestation model has been implemented using the fixed scheduling for the communication described above. Simulation results show perfect adherence to the schedule with the SoCBUS running at 75 MHz. The number of first-try successful routings are shown in figure 5. The reason for the dropoff in first-try successful routing is that the bandwidth from the radio interface ports is too low for the lower frequencies.

Figure 6 shows the maximum circuit setup latency as a function of clock frequency. The average setup latency is about 65% of this value. The full latency of a transmission is the setup latency plus data transmission time which is one cycle per word plus one cycle per hop. The longest path is five hops in this case. The maximum time for the longest



**Figure 6. Maximum circuit-setup latency**

transmission in the system is thus expressed by equation 1.

$$\begin{aligned}
 t &= t_{setup} + \frac{n_{data} + n_{hop}}{f_{clk}} \\
 &= 0.5 + \frac{3136 + 5}{75} \approx 42.4 \mu s
 \end{aligned} \quad (1)$$

This time is an overestimate since the longest setup time is not associated with the longest transmission in this particular application.

## 8.2 Network usage

The network usage on average is relatively low. The by far most used part is the radio interfaces where all samples have to be multicast to the Rake blocks giving a total transfer rate of 61.44 million words per second for these source ports.

The other parts of the network are significantly underused with a maximum transfer rate of slightly less than 12 million words per second. Changing the radio interfaces so that the requirements in that part are lowered below this will yield a significantly lower minimum network clock frequency. The estimated lower limit on clock frequency given from other parts of the network is about 15 MHz.

It must be considered that the data rates in the basestation are relatively low with only about 50 Mbit/s effective payload rate. The highest bandwidth requirement between stages is (of course) significantly higher, considering convolutional coding and soft decision bits. With rate 1/3 coding and four soft bits per data bit, the maximum rate between two stages after despreading is about 600 Mbit/s. Simulations have shown that the total transferred data is slightly less than 27 million words during 80 ms, corresponding to about 5.4 Gbits/s.

## 8.3 Control messages vs. transmission schedule

The overhead from the network is limited to the circuit setup time. Thus, most parts of the network have plenty of free time where control packets can be transferred to the different subsystems from a central controller. Even the radio interface is available for control messages since the path into the radio interface has a low utilization.

## 9 Conclusions

This paper has introduced a case study based on a single-chip WCDMA/FDD basestation which is feasible to implement. The case study has shown the appropriateness of the SoCBUS network for this kind of scheduled hard real-time application. The SoCBUS network is capable of delivering the necessary performance at 75 MHz, which is significantly lower than the maximum achievable speed for the network. Increasing the clock frequency somewhat will directly render gaps in the communication schedule.

The limiting factors for further channels in the basestation is rather the complexity of processing in the multipath search and Turbo decoding stages. The complicated control flow necessary for changes in the basestation data flow is also a complicating matter that may well challenge the system design.

## References

- [1] 3rd Generation Partnership Project, *3GPP web site*, <http://www.3gpp.org>, 2005.
- [2] Richard Hobson, Allan Dyck, and Keith Cheung, "Soc features for a multi-processor wcdma base-station modem," in *Proceedings of the IEEE International Workshop on System-on-Chip for Real-Time Applications (IWSOC)*, July 2004, pp. 318–321.
- [3] Daniel Wiklund and Dake Liu, "Design of a system-on-chip switched network and its design support," in *Proceedings of the International conference on communications, circuits and systems (ICCCAS)*, 2002.
- [4] 3rd Generation Partnership Project (3GPP), *TS25.214 – Physical layer procedures (FDD)*, <http://www.3gpp.org>, 2003.
- [5] Sumant Sathe, Daniel Wiklund, and Dake Liu, "Design of a switching node (router) for on-chip networks," in *Proceedings of the ASICON 2003 conference*, Oct. 2003.
- [6] Lasse Harju, Mika Kuulusa, and Jari Nurmi, "Flexible implementation of a WCDMA Rake receiver," in *Proceedings of the IEEE Workshop on Signal Processing Systems (SIPS)*, Oct. 2002.
- [7] M. A. Bickerstaff, D. Garrett, T. Prokop, C. Thomas, B. Widup, Gongyu Zhou, L. M. Davis, G. Woodward, C. Nicol, and Ran-Hong Yan, "A unified turbo/viterbi channel decoder for 3gpp mobile wireless in 0.18- $\mu$ m cmos," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1555–1564, Nov. 2002.