

# An accelerator architecture for programmable multi-standard baseband processors

Anders Nilsson  
Department of Electrical Engineering  
Linköping University  
Linköping, Sweden  
email: andni@isy.liu.se

Eric Tell  
Department of Electrical Engineering  
Linköping University  
Linköping, Sweden  
email: erite@isy.liu.se

Dake Liu  
Department of Electrical Engineering  
Linköping University  
Linköping, Sweden  
email: dake@isy.liu.se

## ABSTRACT

Programmability will be increasingly important in future multi-standard radio systems. We are proposing an architecture for fully programmable baseband processing, based on a programmable DSP processor and a number of configurable accelerators which communicate via a configurable network. Acceleration of common cycle-consuming DSP jobs is necessary in order to manage wide-band modulation schemes. In this paper we investigate which jobs are suitable for acceleration in a programmable baseband processor supporting a number of common Wireless LAN and 3G standards. Simulations show that with the proposed set of accelerators, our architecture can support the discussed standards, including IEEE 802.11a 54 Mbit/s wireless LAN reception, at a clock frequency not exceeding 120 MHz.

## KEY WORDS

CDMA, OFDM, DSP, SDR

## 1 Introduction

Programmable baseband processors are necessary to support multiple radio standards, since a pure ASIC solution will not be flexible enough. ASIC solutions for multi-standard baseband processors are less area efficient than their programmable counterparts since processing resources cannot be shared between different operations.

In this paper we present an approach to combining baseband processing of multiple radio standards into an area efficient and versatile baseband processor. The key to increase processing capacity and still maintain flexibility is to introduce *accelerators* in the processor. An accelerator is extra hardware added to a programmable processor which performs a certain pre-configured task while the processor is free to perform other operations. However, every extra accelerated function will increase the hardware cost, so selecting the right accelerators to cover most processing needs over multiple standards is essential. In this paper we

identify key accelerators and present an architecture for a programmable processor which is aimed at the following radio standards:

Table 1.

Standard	Modulation	Type
WCDMA	CDMA	3G
TD-SCDMA	CDMA	3G
IEEE 802.11b	DSSS/CCK	Wireless LAN
IEEE 802.11a	OFDM	Wireless LAN

We prove our accelerator concept by using the most demanding algorithm in every accelerator. By using 54 Mbps IEEE 802.11a [1] and 3.84 Mchip/s WCDMA [6] in our calculations we ensure architectural support for less demanding communication standards.

The paper is organized as follows. In section 2, we survey four different communication standards and provide an overview of the operations associated with each standard. In section 3, we discuss the proposed accelerators. In section 4 an accelerator interconnect network proposal is presented. In section 5 our results are presented. Finally conclusions are drawn in section 6.

## 2 Survey of communication standards

To be able to decide which functions to accelerate we survey the different communication standards listed in Table 1. The communication standards have been analyzed focusing on sample rates, required functionality of a baseband processor, and cycle cost/latency requirements. We have chosen to restrict ourselves to only focus on reception in this paper, since the receiver is more computation demanding than the transmitter where all data are known in advance. The maximum data rate of each standard has been used in our calculations. The analysis is divided into several processing tasks:

- Radio front-end processing.

- Symbol processing.
- Data recovery.
- Forward error correction and channel coding.

In order to compare the computational load of a certain algorithm, we define “MIPS cost” which corresponds to how many million instructions-per-second a regular processor would require in order to perform the specified function. The MIPS “cost” is calculated as follows:

$$cost_i = \frac{OP_i \cdot N_i}{t_i}$$

where  $cost_i$  is the associated MIPS cost,  $OP_i$  the number of clock cycles required by a standard DSP processor to perform the operation,  $N_i$  the number of samples or bits to process and  $t_i$  is the maximum time allowed for the operation to complete. For symbol related operations the time to perform the operation is considered to be one symbol time. Latency requirements imposed by the standards [1],[2],[6],[7] has also been taken into account when calculating the required MIPS cost.

## 2.1 Radio front-end processing

The first task for a baseband processor is to filter and decimate the input signal, in order to reduce interference from adjacent channels and to relax requirements on the analog-to-digital converter.

The occupied bandwidth, the over-sampling rate (OSR) and the required sampling rate are presented in the following table:

Standard	Bandwidth	OSR	Sample rate
IEEE 802.11a	20 MHz	2	40 MHz
IEEE 802.11b	11 MHz	2	22 MHz
W-CDMA	3.84 (5) MHz	4	15.36 MHz
TD-SCDMA	3.84 (5) MHz	4	15.36 MHz

As stated in the table above, the maximum sample rate the processor needs to process is 40 MHz.

As decimation filter, we use a raised-cosine FIR filter, which maintains the phase of the received signal. Raised-cosine filters have a symmetrical impulse response with  $N/2-1$  zero taps for a filter length of  $N$ . [13] This property will reduce the computation load significantly since the processor does not need to calculate those taps.

The number of taps in the FIR filter is determined by the smallest roll-off factor. The roll-off factor determines the transition band of the filter. A lower roll-off factor yields better filter performance since the transition band is small, however the length of the filter increases accordingly. IEEE 802.11a [1] requires a roll-off factor of  $r = 3/64 = 0.0468$ . Well known FIR filter design formulae [13] give the required number of taps from the roll-off factor. An  $r = 3/64$  yields a filter of length 21 taps.

For comparison, we have asserted a cycle cost of 30 operations per processed input sample. This cycle cost

also includes control flow instructions. Normally all 21 complex filter taps are processed. However since at most every second output sample is used, savings can be made. 30 operations per sample is a moderate estimation for such a large filter. The resulting required MIPS cost is:

Standard	Required MIPS
IEEE 802.11a	1200
IEEE 802.11b	440
W-CDMA	600
TD-SCDMA	600

## 2.2 Symbol processing

Since synchronization and channel estimation schemes in all four communication standards are diverse, algorithms and functions cannot easily be shared between the different standards. Initial channel estimation and synchronization cost has been estimated to:

Standard	Required MIPS
IEEE 802.11a	108
IEEE 802.11b	12
W-CDMA	25
TD-SCDMA	25

In Direct Sequence Spread Spectrum (DSSS) and CDMA systems, channel estimation is performed by using a matched filter (correlator) to estimate the channel impulse response. The channel estimate is only calculated for as many points as RAKE (see below) fingers used.

Since synchronization and channel estimation is only performed at the beginning of IEEE 802.11a packets and the processing power requirement is acceptable, the synchronization tasks are run completely in software. However for CDMA and DSSS modulation schemes, the matched filter correlator is run continuously. The associated MIPS cost of continuous channel estimation is included in the MIPS cost for the RAKE unit.

For DSSS and CDMA systems, channel compensation and de-spread is performed by a RAKE receiver [8]. The RAKE unit has a certain number of taps (often referred to as “fingers”), which correspond to taps in the channel impulse response. By using four “fingers”, up to 90% of the received signal energy can be used to recreate the transmitted signal in an office environment. [9] The maximum delay in the delay element  $\tau$  is 127 chips to accommodate a delay of  $33 \mu s$  at 3.84 Mchip/s. A Rake finger is shown in Figure 1.

However in OFDM systems, the symbols are recreated in the frequency domain. The conversion from time domain to frequency domain is performed by an FFT. The cost of a 64 point Radix-4 FFT is approximately 430 clock cycles in a regular processor. The associated processing cost for reconstructing the transmitted symbols is presented in the following table:

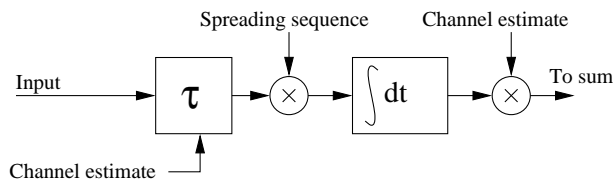


Figure 1. Rake finger.

Standard	Type	Req. MIPS
IEEE 802.11a	FFT	108
IEEE 802.11b	RAKE	550
W-CDMA	RAKE	384
TD-SCDMA	RAKE	384

### 2.3 Data recovery

When the data-symbols have been recreated the de-mapper extracts binary information from the received symbol. De-mapping is tedious work. For each 64 QAM symbol (6 bits worth of data in IEEE 802.11a) 6 comparisons and 6 loads must be made. This yields a total of two operations per received bit. In higher data rate modes of IEEE 802.11b the data are partly processed by using a Modified Walsh Transform. (MWT) [4] The computation of a MWT is similar to a FFT. The cost of de-mapping and data recovery is summarized in the following table:

Standard	Required MIPS
IEEE 802.11a	108 MHz
IEEE 802.11b	160 MHz
W-CDMA	8 MHz
TD-SCDMA	8 MHz

### 2.4 Forward error correction

Forward error correction and channel coding are the most computation demanding functions performed in a baseband processor and are necessary in order to improve data rate over noisy channels.

Several different encoding schemes are often combined to further reduce the bit error rate (BER).

- **Interleaving.** Data are reordered to spread neighboring data bits in time and in the OFDM case among different frequencies.
- **Convolutional codes.** This class of error correcting codes includes regular convolutional codes as well as turbo codes. Turbo codes and regular convolutional codes is very similar. Decoding of convolutional codes is performed by the Viterbi algorithm [11], whereas Turbo codes are decoded by utilizing the Soft output Viterbi algorithm. [12].

- **Scrambling.** In several standards data are scrambled with pseudo-random data, to ensure an even distribution of ones and zeros in the transmitted data-stream.

Since all these schemes operate on bits, and since bit operations and irregular execution is very inefficient in a signal processor, the required MIPS cost is very high. Interleaving for IEEE 802.11a costs 5 op/bit whereas interleaving for WCDMA and TD-SCDMA costs 32 op/bit. The costs of interleaving for the different standards are:

Standard	Required MIPS
IEEE 802.11a	270
IEEE 802.11b	–
W-CDMA	122
TD-SCDMA	122

Viterbi and Turbo codes are considered large research areas by themselves. We acknowledge the MIPS cost for Viterbi and Turbo decoding according to the following table.[11][12]

Standard	Required MIPS
IEEE 802.11a	4000
IEEE 802.11b	–
W-CDMA	1964
TD-SCDMA	1964

Scrambling is also a very cycle consuming task for a programmable DSP since it requires many bit-operations. We have asserted 3 op/bit for scrambling in IEEE 802.11a/b. This yields the following MIPS costs:

Standard	Required MIPS
IEEE 802.11a	162
IEEE 802.11b	33
W-CDMA	–
TD-SCDMA	–

TD-SCDMA uses Reed-Solomon codes as extra data protection in packet transfer mode. An estimated MIPS cost of 20 MIPS is required for Reed-Solomon (RS) decoding. However, since RS is only used in TD-SCDMA and the cost is low, it will not be considered for acceleration.

## 3 Proposed accelerators

In Figure 2 a summary of all MIPS costs are presented. The method of selecting functionality to accelerate must consider:

1. **MIPS cost.** A function with a very high MIPS cost must be accelerated since the operation cannot be performed by a regular processor.
2. **Reuse.** A function that is performed regularly and is used by several radio standards is a good candidate for acceleration.

3. **Circuit area.** Acceleration of special functions is only justified if there can be considerable reduction of clock frequency or power compared to the extra area added by the accelerator.

	IEEE 802.11a	IEEE 802.11b	WCDMA	TD-SCDMA
Filter and decimation	1200	440	600	600
Channel estimation	108	12	25	25
RAKE	—	550	384	384
FFT/MWT	108	160	—	—
Tracking	12	—	—	—
Demap	108	22	8	8
Interleaver	270	—	122	122
Viterbi/Turbo	4000	—	1964	1964
Scrambler	162	33	—	—
CRC	—	0.2	0.2	0.2
Reed-solomon	—	—	20	—

Figure 2. Breakdown of operations and MIPS cost for various standards.

We propose acceleration of operations circled together in Figure 2. As shown in the figure operations common to most standards are grouped together. By accelerating the selected functions we ensure support for the communication standards listed in the figure. The accelerators we suggest are:

- A configurable decimator and filter.
- A four “finger” RAKE accelerator for use in CDMA and DSSS modulation schemes.
- A Radix-4 FFT/Modified Walsh transform accelerator. This accelerator is used in OFDM modulation schemes and in IEEE 802.11b which uses MWT.
- A Turbo/Viterbi decoder.
- A configurable block interleaver.
- A configurable scrambler.

A brief description of the properties required by the proposed accelerator follows:

### 3.1 Decimator

As described earlier IEEE 802.11a requires a FIR filter with a roll-off of  $\sim 0.05$ . This implies a FIR filter length of 21

taps. By allowing the filter to be reconfigured, decimation and filtering can be accommodated for all other communication standards covered by this processor. Since the required MIPS for filtering and decimation is very high and the operations are performed on every input sample this function is accelerated.

### 3.2 RAKE unit

By using four rake fingers up to 90% [8] of the received signal energy can be used to recreate the signal in an pedestrian/office environment. We propose a four finger rake accelerator utilizing an accumulator unit and a simple complex multiplier capable of multiplying samples with  $\pm 1 \pm i$ . The accelerator also contains 512 words of complex memory for delay path storage, de-spread code generators and a matched filter which performs multipath search and channel estimation. Since the input signal is oversampled 4 times, a fractional delay stage is not necessary to provide sub-chip resolution. An overview of the RAKE unit is presented in figure 3.

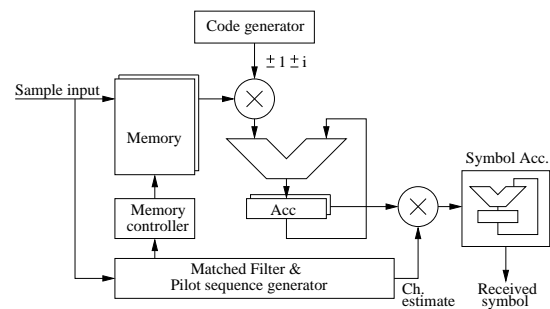


Figure 3. RAKE Accelerator

A more detailed description of a flexible RAKE unit is presented in [10].

### 3.3 Radix-4 FFT/MWT

By using a Radix-4 butterfly and flexible address generators a configurable FFT/Walsh transform accelerator can be built. The structure of such an accelerator is described in [4]. The structure can perform a 64 point FFT in 54 clock cycles and a modified walsh transform for IEEE 802.11b in 18 clock cycles.

### 3.4 Viterbi/Turbo decoder

The Viterbi/Turbo decoder is the most demanding block to implement without acceleration. Since Viterbi and Turbo decoders are frequently used, several implementations exist in the research community.

In [12] a reconfigurable 54 Mbps Viterbi decoder and a 2 Mbps Turbo decoder are presented. The required clock

frequency is 60.5 MHz for 2 Mbps Turbo decoding.

### 3.5 Interleaver

In [3] a multi-mode block interleaver accelerator for IEEE 802.11a has been implemented. Following results have been achieved: Interleaving of 288 bits of data consume 34 clock cycles. The estimated area for the complete block interleaver including memories is  $0.0270 \text{ mm}^2$  in a  $0.18 \mu\text{m}$  process. Since block interleaving in TD-SCDMA and WCDMA is similar, the same structure with a memory block that is written column-wise and read row-wise can be used. Convolutional interleaving for WCDMA/TD-SCDMA is performed by using flexible address generators and regular memory.

## 4 Network

The network is a kind of bus connecting accelerators to the baseband core and to each other. The accelerator network is essential in the baseband processor. We are using a passive network since the network configuration is static during operation. The actual network consists of two sub-networks, one used for sample based transfers and a serial network for bit based transfers. The division of the two networks is essential to improve the throughput of the networks since bit based transfers require tedious framing and de-framing of data chunks not equal to the data width of the network.

Each sub-network consists of a crossbar switch which is configured from the processor. The switch allows related accelerators to be connected with each other and system memories. This crossbar approach enables the data to flow seamlessly between accelerator units without the intervention of the DSP core. The processor is only involved during creation and destruction of network connections.

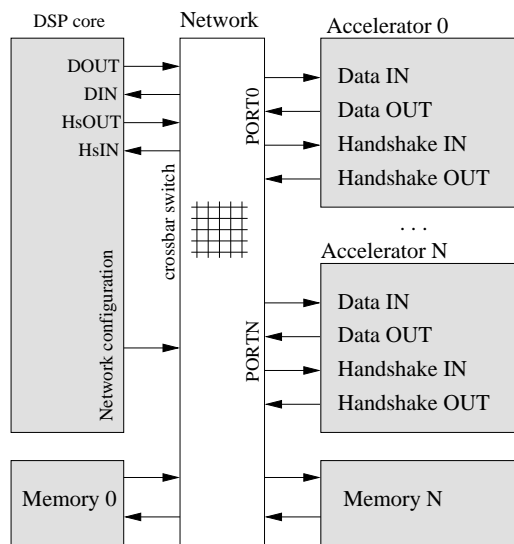


Figure 4. Accelerator network.

By allowing several accelerators to communicate with each other, several accelerated functions can run in parallel and thus lower the computation time further. The functionality of the network is illustrated in Figure 5.

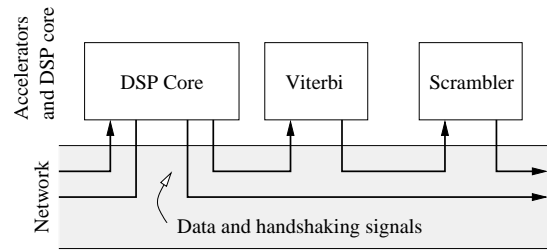


Figure 5. Example of logical network connections.

## 5 Results

The main focus of this paper has been to identify accelerators needed for a programmable baseband processor capable of receiving WCDMA, TD-SCDMA and IEEE 802.11a/b. The different communication standards have been analyzed in terms of required functionality of the baseband processor, cycle cost and the added area of acceleration. We have identified the following accelerators:

- Filter and decimator.
- RAKE accelerator.
- FFT/MWT accelerator.
- Demapper.
- Viterbi/Turbo decoder.
- Interleaver.
- Scrambler.

The proposed accelerators provide functional coverage for the most demanding job in a wide range of radio standards. Combined with the flexibility of a programmable DSP core, this architecture is well equipped for future standard updates.

A complete programmable baseband processor architecture is presented in Figure 6. Simulation of a processor with an accelerator network as shown in Figure 4 has shown that the processor can be run at a clock frequency not exceeding 120 MHz while receiving 54 Mbit/s IEEE 802.11a data. The clock frequency constraint is derived from latency requirements found in IEEE 802.11a. Since the memories are connected to the configurable network, memories can easily be “moved” between different accelerators and the core, without the need of costly memory move operations by the processor.

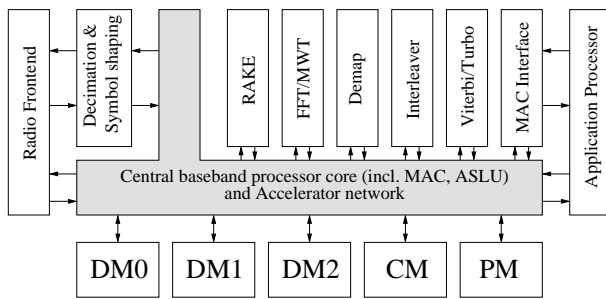


Figure 6. Top architecture of the programmable baseband processor.

To estimate the area used by accelerators, we have synthesized all but the Viterbi/Turbo decoder using Synopsys Design Compiler for a UMC 0.18  $\mu\text{m}$  library. The synthesis result is presented in the following table:

Operation	Without acceleration MIPS	Req. acc. frequency MHz	Area $\text{mm}^2$
Decimator	1200	40	0.17
RAKE	384	16	0.190
FFT/MWT	160	13.5	0.28
Demapper	108	12	0.016
Viterbi	4000	60	1.1
Interleaver	270	8.5	0.047
Scrambler	162	54	0.011
Network	-	-	0.126

All MIPS requirements except the RAKE and FFT unit correspond to IEEE 802.11a. The maximum MIPS requirement on the RAKE unit originate from WCDMA, and the requirement on the FFT/MWT unit originate from IEEE 802.11b. The area of the Viterbi/Turbo decoder is estimated from gate count given in [12].

A complete accelerator architecture except Viterbi/Turbo decoder but including the accelerator network occupies  $0.84 \text{ mm}^2$  in a 0.18  $\mu\text{m}$  process. The total area including core, accelerators, network and memories occupies approximately  $2.5 \text{ mm}^2$ .

## 6 Conclusion

Programmability is essential for multi-standard baseband processors. In order to be able to process high bandwidth communication standards in a programmable processor, acceleration is necessary. As a response to this, we have presented an accelerator architecture for programmable baseband processors targeted at software defined radio applications. We have also presented a selection of functions to accelerate for a CDMA/OFDM baseband processor. Our architecture is versatile and area efficient since we ensure maximum utilization of each accelerator.

## References

- [1] IEEE 802.11a, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications High-speed Physical Layer in the 5 GHz Band, 1999.
- [2] IEEE 802.11b, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band, 1999.
- [3] Eric Tell, Dake Liu; A Hardware Architecture for a Multi Mode Block Interleaver; Submitted to *International Conference on Circuits and Systems for Communications (ICCS)*, Moscow, Russia, June 2004
- [4] Eric Tell, Olle Seger och Dake Liu; A Converged Hardware Solution for FFT, DCT and Walsh Transform; Proc. of the *International Symposium on Signal Processing and its Applications (ISSPA)*, Paris, France, Vol. I, pp. 609 - 612, July 2003
- [5] H Heiskala, J T Terry, *OFDM Wireless LANs: A Theoretical and practical guide*, Sams Publishing, 2002
- [6] 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Physical channels and mapping of transport channels onto physical channels (FDD) 3GPP TS 25.211 V3.5.0
- [7] 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Spreading and modulation (TDD) 3GPP TS 25.223 V5.3.0
- [8] 3rd Generation Partnership Project; User equipment radio reception and transmission; 3GPP TS 25.102 V6.0.0
- [9] H. Holma and A. Toskala; *WCDMA for UMTS Radio Access For Third Generation Mobile Communications*, John Wiley and Sons, Inc., 2001.
- [10] L. Harju, M. Kuulusa, J. Nurmi; Flexible implementation of WCDMA RAKE receiver; *Signal Processing Systems, 2002. (SIPS '02)*. IEEE Workshop on , 16-18 Oct. 2002 Pages:177 - 182
- [11] S. Lin, D. J. Costello Jr.; *Error control coding, Fundamentals and Applications*; Prentice Hall.,1983
- [12] Cavallaro, J.R.; Vaya, M.;Viturbo: a reconfigurable architecture for Viterbi and turbo decoding; Proceedings of *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. (ICASSP '03)*. Volume: 2, 6-10 April 2003 Pages:II - 497-500 vol.2
- [13] L. Wanhammar, H. Johansson; *Digital filters*; Dept. of EE, Linkoping University, Linkoping, Sweden, 2001