VITURBO: A RECONFIGURABLE ARCHITECTURE FOR VITERBI AND TURBO DECODING

Joseph R. Cavallaro and Mani Vaya (cavallar, vaya@ece.rice.edu)

Department of Electrical and Computer Engineering, Rice University
Houston, Tx, USA

1. ABSTRACT

A runtime reconfigurable architecture for high speed Viterbi and Turbo decoding is designed and implemented on an FPGA. The architecture can be reconfigured to decode a range of convolutionally coded data with constraint lengths varying from 3 to 9, rates 1/2 and 1/3, and various generator polynomials. It can also be reconfigured to decode Turbo coded data with constraint length 4 and rate 1/3. Reconfiguration of the architecture requires a single clock cycle and does not require FPGA reprogramming. The proposed architecture can deliver data rates up to 60.5 Mbps for Viterbi decoding and 3.54 Mbps for Turbo decoding, making it suitable for a range of wireless communication standards like IEEE 802.11a, 3GPP, GSM, GPRS, and many others.

2. INTRODUCTION

There has been a growing need for devices that have the flexibility to support multiple communication standards. A reconfigurable architecture, which has the flexibility to operate in multiple standards has obvious advantage over conventional devices in terms of smaller area and seamless switching across standards. This has motivated the design of a high speed reconfigurable channel decoding architecture with Viterbi and Turbo decoding capability as proposed in this paper. Previous work on unified Viterbi/Turbo decoding [1] was limited to 3G codes and data rates (2 Mbps). A flexible Viterbi decoder [2] with data rate up to 2.5 Mbps was also proposed recently.

3. DECODER ARCHITECTURE

As the proposed architecture is highly flexible and caters to high data rate systems, numerous critical issues were addressed in order to realize it. First and foremost was the issue of support for Viterbi as well as Turbo decoding. SOVA and log-MAP are two competing algorithms for Turbo decoding. While the SOVA algorithm offers a small degradation in performance compared to log-MAP [3], the disadvantage is more than offset by the fact that the computational complexity of SOVA is nearly half that of log-MAP, and it is also very similar to Viterbi decoding, hence making SOVA the ideal candidate for our reconfigurable architecture.

Another important issue was the reconfiguration between different constraint lengths(K) and rates(R), on top of the two different decoding techniques. We know that for a constraint length K convolutional code there are 2^{K-1} states and 2^{K-2} butterflies. Each ACS unit in the proposed architecture handles the ACS computation for a single butterfly (Figure 1), and hence 2^{K-2} ACS units are required for fully parallel high-speed operation. The highest constraint length handled by the circuit is 9 and hence 128 ($2^{(9-2)}$) ACS units are required (Figure 2). The lower constraint length decoding operations use parts of the complete circuit. However, the routing of path metrics between ACS units becomes an important issue, as it is different for different constraint lengths. Configurable Data Routers were designed in order to provide this flexibility. Also important for this flexibility are the Branch Metric units and Survivor Management units.

The third important issue was to limit power consumption. Architectural power control schemes were designed so as to power down parts of the circuit that may not be required for a particular decoding type. Figure 3 shows the complete architecture with the reconfigurable/flexible units shaded, each of which we shall now discuss. A detailed study of the architecture can be found in the work by Vaya [4].

3.1. Branch Metric Unit (BMU)

In order to provide enhanced flexibility for all different decoding configurations, the BMU has been divided into three major units: Branch Metric Computation Unit ($BMcompute$), Codeword Look-Up Table ($Codeword\ LUT$), and Branch Metric Multiplexers ($BMmux$).

This work was supported in part by Nokia Corporation, Texas Instruments Inc., the Texas Advanced Technology Program under grant 1999-003604-080, and by NSF under grant ANI-9979465.

BMcompute : As shown in Figure 2, the $BMcompute$ unit computes all the possible branch metrics for a given decoder type with inputs being: the received data, decoder type(Turbo/Viterbi), and the rate. It may be noted here that the number of possible branch metrics is equal to 2^n, where k/n is the rate of the constituent encoder. Each ACS unit needs a specific pair of branch metrics and these are provided to it using the *Codeword LUT* and $BMmux$ multiplexer.
Codeword LUT and BMmux: The *Codeword LUT* uses the constraint length, rate, decoder type, and the index of the ACS unit to provide the relevant $BMmux$ with a codeword, as shown in Figure 2. This codeword is used as a control signal by the $BMmux$ to select the correct branch metric. Also, new codewords can be programmed into the $Codeword\ LUT$, hence providing support for any generator polynomial for constraint length 3-9 Viterbi decoding and constraint length 4 Turbo decoding.

3.2. Add Compare Select Unit

The ACS unit takes in as inputs: the concerned path metrics, the concerned branch metrics, and outputs the survivor path metrics for Viterbi and Turbo decoding, and the decision bits for Viterbi and Turbo decoding. For the case of SOVA based Turbo decoding, the difference between the path metrics [3] is also computed. It may be noted here that, since $K = 4$ for constituent Turbo encoders, four ACS Units, specifically with indices from 0 to 3 have been programmed to do the ACS computations for both Viterbi and Turbo decoding, while the remaining ACS units(4 -127) are only activated when computing path metrics and decisions for Viterbi decoding.

3.3. Configurable Data Router

Because we have designed a completely parallel high speed architecture, the intermediate path metrics are not stored, but are routed back to the relevant ACS units for use in the next clock cycle. However, this is a complex problem since the routing of the path metrics varies with the constraint length in question. Configurable Data Routers (Figure 2)were designed [4] to solve this problem. Configurable Data Routers consist of banks of multiplexers, each multiplexer receiving inputs from the outputs of different ACS units, and feeding a particular ACS unit, as shown in Figure 1. The Multiplexers have inbuilt logic to route the path metrics according to the constraint length, decoding type, and the index of the Multiplexer. While 2 input multiplexers are sufficient for Viterbi decoding, 4 input multiplexers are needed for Viterbi/Turbo decoding. As we can see, Configurable Data Routers are critical to the reconfigurability of the architecture, as they provide the flexibility to migrate between different constraint lengths and decoding types.

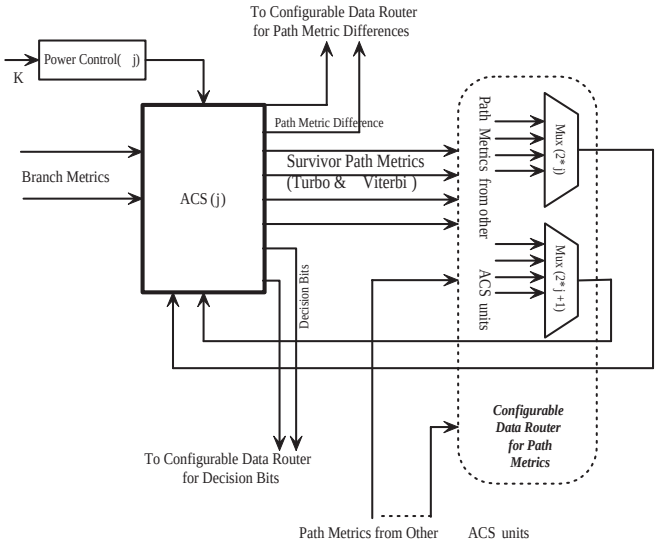

Fig. 1. j^{th} ACS unit and Configurable Data Router interconnections

For writing the decision bits and path metric differences to memory it is important that the data be written in order. With varying constraint lengths, the order of the data changes and hence Configurable Data Routers (as shown in Figures 2, 3) have been employed to route the data according to the constraint length.

3.4. Survivor management Unit (SMU)

The flexible traceback units, which form the core of the SMU, use the decoder type, constraint length, current state, and the decision bit stored at a certain state to evaluate the previous state and the decoded bit. The SMU also contains additional hardware for soft decision computation for Turbo decoding, which is powered down when Viterbi decoding is in progress.

3.5. Interleaving

The focus of this work was on the reconfigurability between different decoding techniques, and since interleaving is only used for Turbo decoding, a simple block based interleaver was implemented. Data is written in a matrix format, and the transpose of the matrix is output as the interleaved data.

4. ARCHITECTURAL POWER CONTROL

As explained earlier, for a high speed decoding architecture, a fully parallel architecture has been developed. For the case of constraint length K, 2^{K-2} $BMmuxes$ and ACS units are required for a fully parallel computation. While 4 ACS

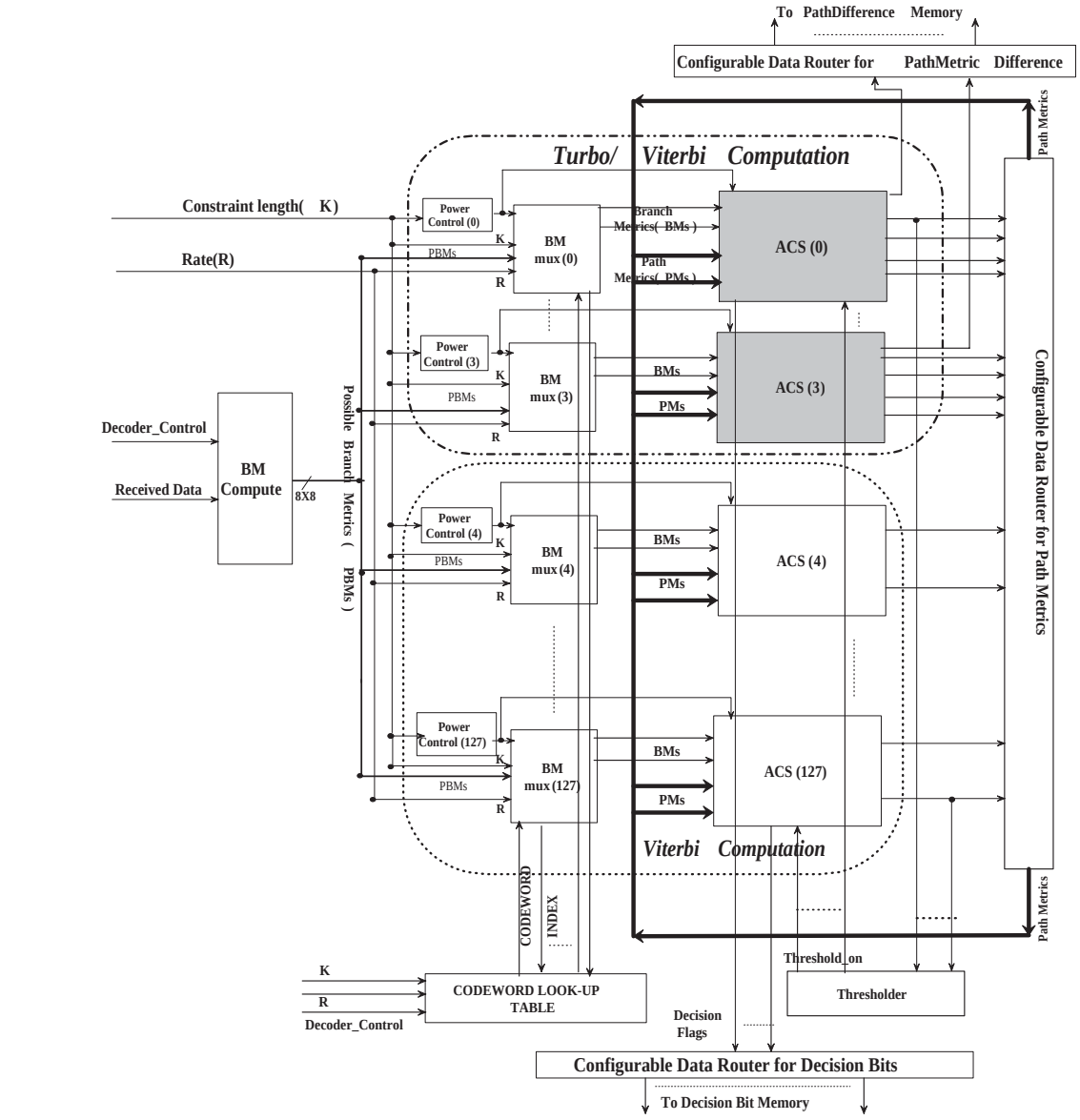

Fig. 2. ACS, BMU and Configurable Data Routers

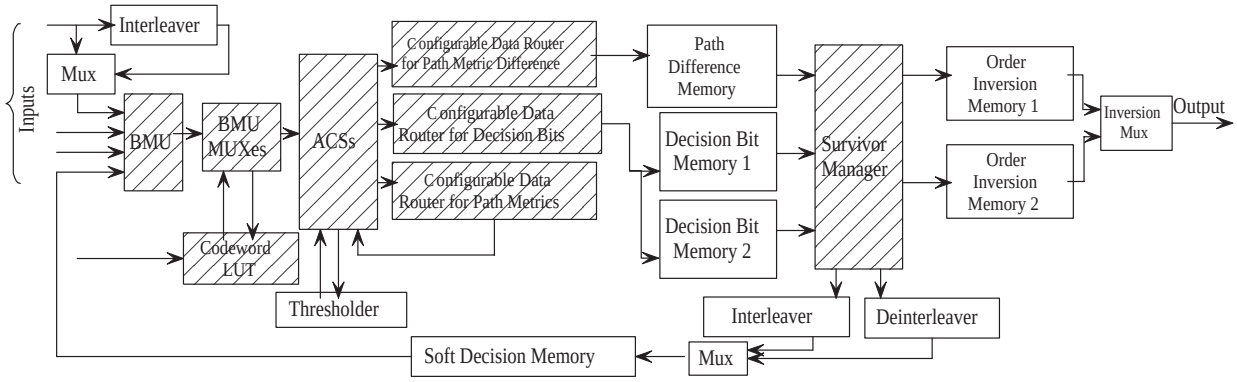

Fig. 3. VITURBO: The complete architecture

and $BMmux$ units are used for constraint length 4 decoding, 128 ACS and $BMmux$ units are used for constraint length 9 decoding. The power control mechanism [4] shuts down the clock inputs to the $BMmuxes$ and ACS units that are not pertinent to the ongoing decoding (Figure 2), hence saving power. Also, depending upon the decoding type (Viterbi/Turbo), parts of the SMU that are not being used are shut down.

5. SYSTEM IMPLEMENTATION AND RESULTS

	Decoder Type	Gates (Logic)	Gates(Memory) 128 bit frame	Max. Freq.
1.	Viterbi (7)	62,987	65536	71.3 MHz
2.	Viterbi(9)	166,348	262,146	68.4 MHz
3.	Turbo	37,812	65536	67.7 MHz
4.	Viterbi(3-5)	33,487	16,384	64.6 MHz
5.	Viterbi(3-5)+ Turbo(4)	42,385	81,812	63.3 MHz
6.	Viterbi(3-7)	67,042	65,536	62.8 MHz
7.	Viterbi(3-7)+ Turbo(4)	76,037	131,072	62.1 MHz
8.	Viterbi(3-9)	181,560	262,146	61.9 MHz
9.	***VITURBO : Viterbi(3-9)+Turbo(4)***	***190,288***	***327,680***	***60.5 MHz***

Table 1. Architectural Tradeoffs (Area/ Frequency)

Decoder Type	Clock Freq.	Data Rate	Power Consumption (sans Quiescent Power)
Viterbi (K=5)	60.5 MHz	60.5 Mbps	154.67 mW
Viterbi (K=7)	60.5 MHz	60.5 Mbps	561.64 mW
Viterbi (K=9)	60.5 MHz	60.5 Mbps	1.59 W
Turbo(K=4)	60.5 MHz	3.54 Mbps	184.88 mW

Table 2. Power Consumption for different configurations of VITURBO (Quiescent Power = 225 mW)

The proposed architecture was implemented on a Xilinx Virtex II FPGA and VHDL was used to describe the architecture. Table 1 shows the various tradeoffs involved in the proposed reconfigurable design. The gate counts for logic and memory have been separated in order to give an in-depth analysis of the architecture. Under the column 'Decoder Type', the numbers in brackets represent the constraint length of decoder. Comparing the logic gate count for a standalone constraint length 9 Viterbi decoder (2^{nd} row), with that for a reconfigurable constraint length 3 to 9 Viterbi decoder (8^{th} row), we see that the gate overhead for reconfiguration is only 9%. Now comparing the reconfigurable constraint length 3 to 9 Viterbi decoder with our VITURBO architecture (constraint length 3 to 9 Viterbi and Turbo), we see that the gate overhead for Turbo decoding is only 5%.

Table 2 shows the achievable data rates and the power consumption for different configurations of VITURBO. For the case of Viterbi decoding, the throughput is one output per clock cycle (with some initial latency). As shown in Table 1 the maximum clocking frequency for VITURBO is 60.5 Mhz, and hence data rates upto 60.5 Mbps are possible for Viterbi decoding. However, Turbo decoding throughput is lesser as numerous iterations are required for generating reliable results. For a clocking frequency of 60.5 Mhz, the throughput is 3.54 Mbps (for four iterations). From the table we can also compare the power consumption for different configurations of VITURBO. We see that for the same throughput, constraint length 5 Viterbi decoding requires much less power than constraint length 9 Viterbi decoding, as the computational complexity for constraint length 5 decoding is much smaller. It is clear from the data presented that while reconfigurable architectures provide enhanced flexibility at the cost of marginal increase in gate requirements, the power consumption is limited to the active gates in the selected configuration.

6. CONCLUSIONS

In this paper, we have reported a single reconfigurable architecture for Viterbi and Turbo decoding. This architecture can provide throughputs in the range of 60 Mbps for constraint length 3-9 Viterbi decoding and 3.54 Mbps for SOVA based Turbo decoding (4 iterations). It was demonstrated that with a 5% overhead in area (excluding memory), a constraint length 3-9 Viterbi decoder could support Turbo decoding. Such an architecture will find applications in devices which will support multiple standards for wireless communications. Power saving techniques ensure that the architecture is feasible for receiver structures, where power is a critical issue.

7. REFERENCES

[1] M. BickerStaff, D. Garrett et. al., "A unified Turbo/Viterbi Channel Decoder for 3GPP Mobile Wireless in 0.18 μm CMOS," *ISSCC*, pp. 124-451, 2002.

[2] D. Hocevar, A. Gatherer, "Achieving flexibility in a Viterbi decoder DSP Coprocessor," *IEEE VTC Fall*, pp. 2257-2264, 2000.

[3] J. Hagenauer and P. Robertson, L. Papke, "Iterative decoding of Systematic Convolutional Codes with the MAP and SOVA algorithms," *ITG Conference*, 1994.

[4] M. Vaya, "VITURBO: A Reconfigurable Architecture for Ubiquitous Wireless Networks," *M.S. Thesis, Rice University, Houston, Tx*, Aug 2002.