Summary of the Ph.D. Thesis
"Research and Realisation of Turbo-Block-Codes"

Dipl.-Ing. Kalyan Koora
koora@ifn.et.tu-dresden.de

A new class of codes, turbo codes, have been introduced for channel coding in communication systems by French scientists in 1993. Due to the iterative decoding scheme of these codes, it is possible to achieve results near to the Shannon limit. In order to decode these codes, implementation of a soft input and soft output decoder is essential. The realisation of such type of decoders can be done either by using a maximum a posteriori (MAP) symbol estimation or by using a soft output Viterbi algorithm (SOVA). The main objective of this Ph.D. thesis was to carry out a research study of different algorithms and design a suitable hardware architecture of the turbo decoder.

Since turbo codes consists of at least two recursive systematic convolutional codes (RSC), an interleaver, puncture patterns, at first a study of each component has been carried out. In many systems mostly block codes are preferred to encode, transmit and decode the frames independently from one and other. Due to the application of the RSC codes the difficulties in terminating the trellis of the decoders were also pointed out. By using tail bits and the polynomial division property of RSC a new approach to terminate both the decoders is presented, *Modified Turbo Block Codes (MTBC)*. Further, MTBC allows to use *variable* input block lengths, whereas the maximum block length is equal to the depth of the interleaver. This type of coding scheme also conveys an extra information to the decoder, decoding starts and ends in zero state. With this additional information the decoder is able to enhance its correction performance by suppressing the *flattening effect* which results by keeping the trellis of the decoders open. In channel coding interleavers are mainly used to split bundle or periodic errors into single errors. In addition to this, interleavers help the turbo codes to increase the hamming distance of the overall code, to ensure one redundancy bit for every systematic bit, to keep the decoder outputs with inputs uncorrelated. Since it has been shown in the literature that random interleavers are suitable for long block lengths, here two methods are presented:

1. making use of selected input frames of weight two
2. selection of all suitable input frames for the design of the interleavers (*binary tree*)

In the initial stage of the research it has been found that the implementation of SOVA decoders leads to the fast processing decoders and offer a straight forward solution with less control architecture. Thus, the implementation of turbo decoders using SOVA as SISO component has been considered.

Simulation tool aided algorithm test was the first step taken to implement the above described turbo decoder. C program realising a SOVA was written for the simulation tools SPW and COSSAP. After successful implementation and testing of the floating point software code, a

bittrue model of the decoder was programmed. There all calculations were restricted to integer numbers keeping the domain of numbers variable. This made it easy to test the influence of the input, output and internal word lengths on the correction capacity of the decoder. An extra care has been taken as soon as the internal results crossed upper and lower bounds of the set domain. A simple programming type has been introduced which makes it possible to switch between ideal and bittrue codes. This helps to select suitable word lengths of the algorithm at different processing stages without any great performance loss. In a parallel approach the code was fed into Cadence and Altera software tools. Concentrating on the flexibility of the codec for further tests, e.g. use of different random interleavers, influence of the word lengths in real systems etc., and taking the cost point of view into consideration it was finally decided to test the decoder using field programmable gate arrays (FPGA) from Altera. The control of the hardware circuit was taken up by a personal computer. The communication between the PC and the circuit was achieved via parallel port. Since the circuit performs only decoding, rest of the functions like data generation, encoding, puncturing, distortion and quantisation were also performed by the PC. Presently two modes are supported by the circuit. In first and simple mode the circuit takes up the received and extrinsic information and performs one complete iteration and gives back the new extrinsic information. This allows to observe the development of the results after every iteration. As the communication via parallel port is considered to be one of the bottle neck, this mode is very slow compared to the second and fast mode. In the second mode the iteration number is set as a parameter. After the reception of the information block along with the redundancy information, the desired number of iterations are performed by the hardware circuit and final results are given back to PC.

After the success of hardware implementation and software programming measurements were carried out to test the efficiency of the designed test system. Many measurements were carried out in AWGN and MEDIAN 60 GHz channel environment for BPSK and OFDM-DQPSK modulation schemes. The flexibility of the test system has been proved using the VHDL code by changing the word lengths and the length of the soft update unit of the decoder core. For further applications the use of error histogram and the use of adaptive iteration scheme were discussed.