VIDEO TRANSMISSION THROUGH DOMAIN SPECIFIC RECONFIGURABLE ARCHITECURTES OVER SHORT DISTANCE WIRELESS MEDIUM UTILIZING BLUETOOTH IEEE 802.15.1™ STANDARD

Imran Ahmed[1], Tughrul Arslan[2], Sami Khawam[3]

[1,2,3] *School of Electronics and Engineering. University of Edinburgh, King's Buildings Mayfield Road, Edinburgh, EH9 3JL, UK*

Abstract

This paper describes efficient video transmission over Bluetooth using Reconfigurable architectures. Currently most of the hardware solutions in research utilize MPEG-4 compression algorithms to compensate for the limited bandwidth available on Bluetooth channel. The reconfigurable MPEG-4 and Error Correction Encoders and Decoder blocks presented in this paper not only provide flexibility but also means where dynamic or static mapping of different algorithms can be done to meet various performance constraints vis-à-vis reduced silicon area, reduced power, improved speed and error correction.

Keywords: Reconfigurable, turbo codes, array, programmable, Bluetooth, Motion Estimation, DCT, embedded, Domain Specific, low-power.

1. Introduction

The authors recently introduced a Reconfigurable SoC platform [1] where computational intensive parts of MPEG4 Compression (DCT and Motion Estimation ME) were implemented in hardware. The flexibility of the DCT and ME arrays have also been shown [2]. The Reconfigurable MPEG-4 design was implemented using LEON processor in [3] as shown in Fig 1 below:

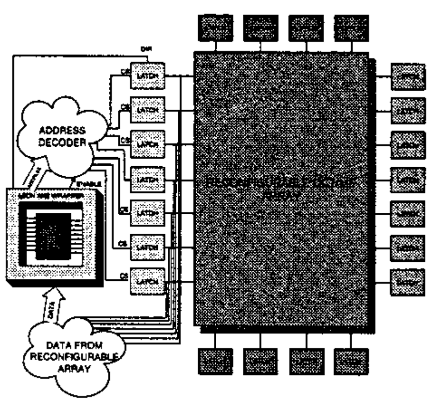

Fig 1. Reconfigurable MPEG-4 array controlled by LEON Processor.

The aspects of reconfigurability in the above mentioned design is out of scope of this paper and are not discussed here. The authors have now built a Bluetooth platform with the aim of demonstrating the effective video transmission utilizing the advantages of the dynamically-reconfigurable system investigated in [3]. The design utilizes the same DCT and ME arrays [1]-[4] but the platform is changed utilizing ARM7TDMI [5]-[8] processor instead of Open Source Leon processor [26].

The processor performs the operations that cannot be done efficiently in the reconfigurable logic, such as data-dependent control and program flow control while the computational cores are mapped to the reconfigurable hardware.

As shown in figure 2 below the overall architecture consists of five major modules. MPEG4 Block with DCT and ME blocks mapped on Reconfigurable arrays and remaining MPEG4 algorithm flow control and Bluetooth software stack running on ARM7 Processor. There are also separate Baseband unit and RF analog Radio module (Ericsson chipset)[9].

The paper is organized as follows. In section 2 a general description of Bluetooth is given. In section 3 Reconfigurable hardware requirement and developed blocks are described. In section 4 Software portion of the Bluetooth is discussed and implementation using embedded L2CAP commands is described. In Section 5 Results from the research are presented. The conclusion and future research areas are discussed at the end.

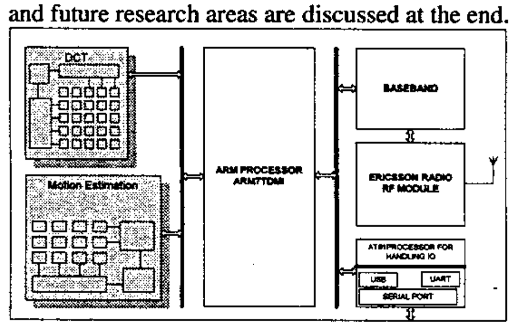

Fig 2. System Block Diagram

2. Overview of Bluetooth System

As per Bluetooth specification [10] Bluetooth Radio operates in the ISM band of 2.4 GHz. Information is modulated using frequency hoping GFSK (Gaussian Frequency Shift Keying) at the rate of 1Msymbols per second and transmitted in one of 79 1-MHz channels. Any Bluetooth design motivation factors will be low complexity, power consumption and low cost targets [11]. The Bluetooth operates on a master slave concept. The device that initiates the connection is Master. Slaves respond to Master which can control data transmission of up to 7 Slaves [14].

3. Reconfigurable Hardware.

The design utilizes static reconfigurability by mapping different ME and DCT algorithms on the reconfigurable blocks providing flexibility for SOC design. The mapping of different algorithms on these arrays and their effect on area, power and speed has been shown previously [2].

The clusters that were identified as reusable blocks for Motion Estimation [2] include Register-Multiplexer (with optional register at the output), Absolute Difference Calculator, Adders and Subtractors and Accumulator as shown in figure 3.

Similarly the clusters for DCT [2], [4] supporting Distributed arithmetic implementations are: Add-Shift and Memory elements. These clusters are shown in fig 4.

Figure 3: Array for Motion Estimation

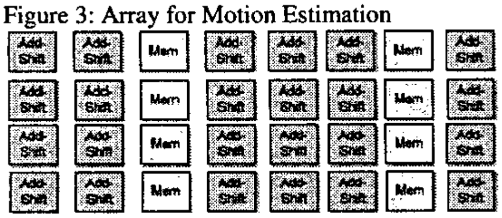

Figure 4: Array for Distributed Arithmetic DCT

Bluetooth error correction was kept simple in the specification [10] due to short distance line of sight communication. However Quality of Service can vary significantly due to interference, objects blocking the links or devices moving around. There has been significant interference widely reported [18]-[21] due to Wireless LAN 802.11, Wifi or Home RF which operates at the same frequency as Bluetooth. This interference can cause spatial and temporal error propagation as well as synchronization errors effecting video quality [18].

Video quality can further be affected by drop in transmission data rate. Different ACL packets for Asynchronous Bluetooth transmission are shown in fig 5.

Packet Type	Payload (bytes)	Data Rate kbps
DM1	0-17	108.8
DH1	0-27	172.8
DM3	0-121	387.2
DH3	0-183	585.6
DM5	0-224	477.8
DH5	0-339	723.2
AUX1	0-29	185.6

Figure 5: Bluetooth packet types and Data rates.

The rates of data transmission as shown in the figure are at the Base band level. Each layer adds a header and the rate available to application will be less than the rates as in figure 5.

ARQ Retransmission for payloads in case of FEC and CRC enabled packets, can further reduce the data rate. The acceptable picture quality for MPEG video is 384Kb/s [17], which can be seriously affected by the above mentioned factors. These results indicate that improved error correction techniques can improve the quality of video especially at low SNR.

Turbo Code based error control has shown to improve QOS [22] of Bluetooth at low bit rates and low SNR. Dynamic reconfigurable architectures have already been utilized [25] for implementing Max-Log-Map Algorithms [24] for Turbo codes for 3G networks. Similarly Parallel Convolution Coding scheme for turbo codes is also proposed for 3GPP standard [23]. The same concepts can be used in Bluetooth to have a very efficient error correction scheme when BER is very high due to low SNR.

The proposed block diagram of Base band employing Dynamically Reconfigurable Turbo Coder as in [25] is described in figure 6 below.

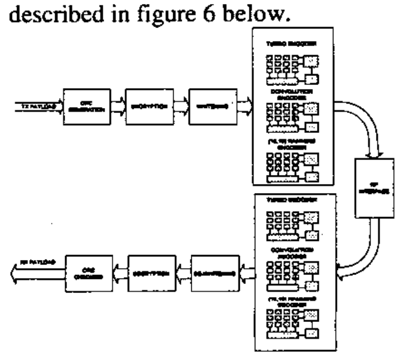

Figure 6: Reconfigurable encoding/decoding scheme for Bluetooth

The type of error encoding selected by the upper layers depends on SNR. When SNR is high the usual Bluetooth hamming (15, 10) encoding is employed. Using the received signal strength indicator RSSI which is indication of SNR of any particular packet the quality of channel can be estimated. This indicator can be used during decoding phase. For BER of 10^{-6} Turbo Coders will be used and for BER of 10^{-3} convolution coding is used.

For Turbo decoder, addition, Max operation calculation and Memory are the main functional blocks.

Bluetooth also has a feature that enables devices to change the packet types depending on the quality of the link. In this simple technique Device A sends LMP_auto_rate PDU to notify device B to enable this feature. Once enabled device B can change the packet size device A sends by sending LMP_preferred_rate PDU. Since the implementation of this messaging is application specific hence can be utilized for changing the encoder and decoders for error correction.

Processor and the Reconfigurable system are loosely coupled where reconfigurable hardware acting as a coprocessor to host ARM7TDMI processor.

5. Bluetooth Software.

The software development is done in three phases. In the first phase MPEG4 application developed over the Bluetooth stack in full PC simulation environment as shown in figure 8 below.

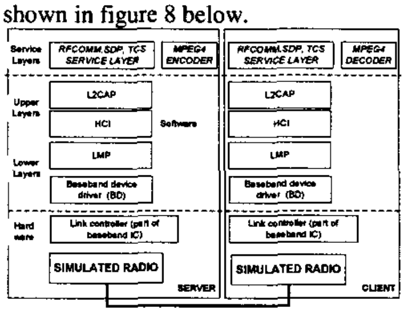

Figure 8: PC based Simulation of Bluetooth Stack.

In the second phase the application is developed in a Split Stack fashion where Ericsson EBDK [9] boards are utilized. These are commercially developed boards that have Bluetooth network Stack layers upto and including the host controller interface (HCI) layer fixed as firmware. The higher Bluetooth protocol layers can be run on a standard PC, Controlling the EBDK over a serial link in a so called split stack set up as shown in fig 9 below.

In the last phase fully embedded application is run on development boards utilizing ARM7TDMI and later part of research will interface Micro Emissive Displays with the embedded solution.

The MPEG4 Software layer is build on top of L2CAP layer and utilizes services provided by L2CAP layer. The communication between Master and Slave (Two Peer Devices) is ACL (Asynchronous Connectionless) link. The ACL link is a point to multipoint link between the master and all the slaves participating on the Pico net [10]. The communication between the Bluetooth devices is via packets. Software system utilizes an efficient Kernel. MPEG4Server and MPEG4 Client peer tasks defined in Kernel communicate as following

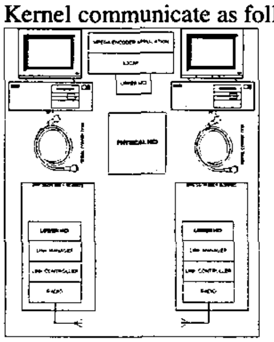

Figure 9: Split Stack Simulation utilizing Ericsson Boards.

MPEG4ServerApplication

- Set Major and Minor Device Class.
- Make the Server Device connectable.
- Make an Inquiry in to available Bluetooth devices in the vicinity. At this point Slave device will respond with its BT Address.
- Send L2CAP Connect Request.
- Page-HCI Connect Request.
- Set Authentication and Encryption parameters.
- Send L2CAP Configure Request.
- Set HCI QOS Setup (Link Manager sends QOS Request to slave).
- Accept/Reject the Configuration Request from Client (2^{nd} peer).
- Send L2CAP Data Request. The data packet in turn is segmented by L2CAP to HCI MTU Size, as per agreed QOS. HCI Data Request is send by HCI Layer.
- Base band may in turn further Segment HCI Data to fit in currently allowed packet types [12].
- ACL Data Packet is send.

MPEG4ClientApplication

- Set Major and Minor Device Class.
- Make the Client Discoverable- Enable Page Scan
- Register PSM for MPEG4 Server Application
- Give Response to Connect Request from Server
- Check the Configuration parameters requested from Server and Give Response.
- Request for Configuration parameters to Server.
- If Server accepts the configuration parameters requested, start receiving MPEG4 Data.

6. Results

Area usage of DCT implementation on reconfigurable fabric is described below.

Computations done on reconfigurable fabric had demonstrated 75% reduction in power consumption when compared to generic FPGA architecture, which makes them the ideal candidate to be implemented in Bluetooth devices for transmission of Video information.

Power Consumed by ARM7 is 5.85 mW, Reconfigurable arrays is 1.08mW, which shows considerable power improvements if compared with Xilinx Virtex –E (4.37mW). Power Consumption for 0.18um ASIC is 0.68 um.

Similarly Area consumed by arrays is 21189(um^2) as compared to Xilink Virtex E (38340 um^2) it is again a significant improvement.

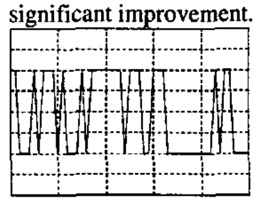

Figure 10: Input data to turbo encoder

Figure10 Shows Matlab Simulation of Turbo Encoder, when input data is supplied to turbo encoder. Input data rate is R_x.

As shown in figure 11 output data is three times the rate of input data. The algorithm used in the simulation is

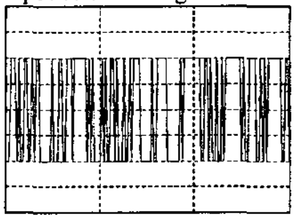

Figure 11: Output data from encoder

Max-Log-Map. White Gaussian noise is added to the data (simulating channel). Output of data is shown is figure 12. The data output is after three iterations.

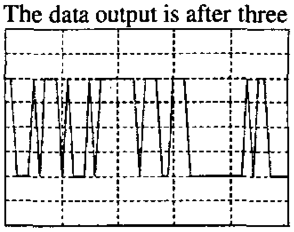

Figure 12: Output data from turbo decoder.

7. Conclusion.

Paper has presented the concept of Reconfigurable System for Bluetooth platform. Reconfigurable DCT and ME arrays presented in [1]-[4] are utilized again for this research. The requirement of improved error correction has been demonstrated and reconfigurable design using Turbo coding shown. Processing elements for reconfigurable ME, DCT, Turbo Decoders/Encoders are also identified. The direction of future work is to integrate the embedded platform with Micro Emissive Displays.

8. Acknowledgements

(Not shown for confidentiality)

9. References

[1] Khawam S, Arslan T, Westall F, "Embedded Reconfigurable array targeting motion estimation applications", 2003 IEEE International Symposium on Circuits and Systems (ISCAS 2003)

[2] Ahmed I, Arslan T, Baloch S, Khawam S, " Efficient implementation of Mobile Video Computations on Domain Specific Reconfigurable Arrays." Design, Automation and Test in Europe Conference and Exhibition, 2004. Proceedings ,Volume: 2 , 16-20 Feb. 2004

[3] Ahmed I, MSc Project Report "System Level Integration of Low Power High Performance Reconfigurable MPEG 4 architectures"., www.sli-institute.ac.uk

[4] Khawam S, Arslan T, Westall F, "Domain Specific reconfigurable array for Distributed Arithmatic".13th International Conference on Field Programmable Logic and Applications (FPL2003)

[5] ARM Ltd. ARM architecture. [5] ARM Ltd. ARM architecture. http://www.arm.com/armtech/

[6] ARM Ltd. An introduction to Thumb, March 1995

[7] ARM Ltd. Application note #31: Using EmbeddedICE, Feb 1999. http://www.arm.com/support/567G46/$File/ATPCS.pdf?OpenElement

[8] ARM Ltd. Product overview: ARM7TDMI (rev.3) core processor, 2001. http://www.arm.com/techdocs.nsf/html/ARM7Docs.

[9] Bluetooth: Connect Without Cables, Jennifer Bray, Charles F. Sturman, Prentice Hall, ISBN 0130898406

[10] Bluetooth SIG, Specification of the Bluetooth System, version 1.2, www.bluetooth.com

[11] J.C. Haarsten, and S. Mattisson, "Bluetooth- A New Low Power Radio Interface Providing Short-Range Connectivity", Proceedings of the IEEE, vol. 88, pp. 1651-1661, Oct 2000.

[12] "MPEG-4 Video Transmission over Bluetooth links" by Chong Hooi Chia, M Salim Beg , Personal Wireless Communications, 2002 IEEE International Conference on , 15-17 Dec. 2002 Pages:280 - 284

[13] Corina Scheiter, Rainer Steffan, Benno Stebernack, "A System for QOS enabled MPEG-4 Video Transmission over Bluetooth for Mobile Applications".

[14] Ko Sung-Yuan,"The Embedded Bluetooth CCD Camera," Electrical and Electronic Technology, 2001. TENCON. Proceedings of IEEE Region 10 International Conference on , Volume: 1 , 19-22 Aug. 2001 .

[15] Ericsson. Bluetooth development kit (EBDK). http://www.ericsson.com/bluetooth/

[16] Hirotsugu Okura, Masami Kato and Shuji Tasaka, "The Effect of Segmentation Mismatch on Quality of continuous Media Transmission by Bluetooth", Personal, Indoor and Mobile Radio Communications, 2002. The 13th IEEE International Symposium

[17] "Performance of the Bluetooth-Based WPAN for Multimedia Communication".

[18] Shaou-Gang Miaou, Chia-Yuan Huang, Kuang-Jui Ho, Ming_Chou Tu,"Quality Degradation and Improvement of H.263 Video Transmitted in Bluetooth Packets under the Interference of Wireless LAN"

[19] M. B. Shomake, Wi-Fi and Bluetooth Coexistence Issues and Solutions for the 2.4GHz ISM Band, T.I White Paper V 1.1, Feb, 2001

[20] Mobilian Corporation Wi-Fi (802.11b) and Bluetoooth: An Examination of Coexistence Approaches Problem, 2001.

[21] IEEE P 802.15 Working Group for Wireless Personal Area Networks Procedure for Selection of WLAN-WPAN Coexistence Mechanisms, IEEE802.15-00/353, Nov 2000.

[22] Valenti, M.C.; Robert, M. "Improving the QOS of Bluetooth Through Turbo Coding"MILCOM 2002. Proceedings , Volume: 2 , 7-10 Oct. 2002

[23] 3GPP TG 25.212 http://www.3gpp.org/.

[24] A. Chass and A. Gubeskys, "Efficient Software Implementation of Max-Log-Map Turbo Decoding Algorithm on StarCore SC140 DSP", Motorola Semiconductor Israel, Ltd., ICSPAT 2000 Israel.

[25] Designing and Mapping of a Turbo Decoder for 3G Mobile Systems Using Dynamically Reconfigurable Architecture". By Ahmad Alsolaim and Mingwei Ding.

[26] Leon processor, www.gaisler.com