Reply letter

“Hardware Reconfiguration Scheme Based on the Digital TV Signal”

June 13, 2015

Dear Editor,

We really appreciate the comments of the TCSVT reviewers, which were highly insightful and enabled us to greatly improve the quality of our manuscript. The following pages list our responses to each of the comments.

Attached to this letter, please find a revised version of our manuscript. We hope that the revisions in the manuscript and our responses will be sufficient to make our manuscript suitable for publication.

Best Regards,

Rodrigo de Oliveira, Lucas Cordeiro, Eddie de Lima Filho, and Vicente de Lucena Júnior

Federal University of Amazonas (UFAM), Brazil,

E-mail: rodrigo@dcc.ufam.edu.br

**Associate editor’s comments to the author**

The paper presents a way to reconfigure the DTV receiver. This might be good to compare the approach with what is done in the MPEG reconfigurable video coding standard whose has merely the same goal.

**Response:** Two paragraphs were added at the end of section II. They had not been previously included because the proposed approach has a broader application than only video coding. However, they indeed present the same goal.

**Key references that must be included:**

From reviewers' comments (please check this section):

The related work section is rather short. I think more work around agile software update can be added which can include some watermarking based approaches supporting both legacy systems and new systems.

**Response:** The related work section was completely reviewed and extended, while including a discussion regarding the suggested references.

The motivations of this work are more or less the same that those of the MPEG Reconfigurable Video Coding. Authors should mention it in the background of their work, e.g.:

• S. S. Bhattacharyya, J. Eker, J. W. Janneck, C. Lucarz, M. Mattavelli, M. Raulet, Overview of the mpeg reconfigurable video coding framework, Jrnl. of Signal Proc. Systems 63 (2).

Considering also that, in that field, reconfiguration has already been tackled either at a finegrain (FPGA only):

• E. Bezati, S. CasaleBrunet, M. Mattavelli, J. Janneck, Synthesis and optimization of highlevel

Stream programs, in: Electronic System Level Synthesis Conf., 2013, pp. 16.

• J. Janneck, I. Miller, D. Parlour, G. Roquier, M. Wipliez, M. Raulet, Synthesizing Hardware from Dataflow Programs: An MPEG4 Simple Profile Decoder Case Study, Jrnl. of Signal Proc. Systems 63 (2) 241249.

and at coarsegrain (including, but not limited to, FPGA implementations):

• Nezan J, Siret N, Wipliez M, Palumbo F., Raffo L (2012). MultiPurpose

Systems: A Novel DataflowBased Generation and Mapping Strategy. Symp on Circuits and Systems (ISCAS), pp. 30733076

• C. Sau, L. Raffo, F. Palumbo, E. Bezati, S. CasaleBrunet, M. Mattavelli, Automated design flow for coarsegrained reconfigurable platforms: An rvccal multistandard decoder usecase, in: 2014 International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation, pp 5966.

**Response:** All mentioned references were included and used in the text (see references [17] to [24]), along with others related to the same area.

Would suggest including a reference to the Transport Stream standard.

ISO/IEC 138181, ITUT Recommendation H.222.0

**Response:** It was already there. Now it is reference [25].

**Responses to the comments of Reviewer #1**

This paper covers very up to date topics: how to keep pace with the rapid evolution of the video standards without the need of completely changing the provided technologies and how to manage intelligent/adaptable systems.

According to my humble opinion the paper is too much pedantic. Section IV.A for example reports basic concepts for hardware designers.

**Response:** That was not our intention. Maybe, the broad applicability of the proposed technique may have caused it. Section IV was created in order to provide a better understanding of the framework, for readers of different areas (*e.g.*, video coding).

Keeping pace with the rapid evolution of the video standards is one of the claims of the paper. Point is that reconfiguration help maintaining the same reference technology by means of reprogrammability.

Nevertheless, it does not help in passing from one standard to other in a flexible way. As far as I understood from the paper the designers would need to write the hardware description from scratch. Please comment on this aspect considering also the considerations made below with respect to Reconfigurable Video Coding.

**Response:** Yes, the proposed methodology is not as flexible as the RVC framework. Indeed, a new hardware description (regarding the current approach, not RVC, it will developed anyway) is need. A discussion was added to the last paragraph of section II.

The motivations of this work are more or less the same that those of the MPEG Reconfigurable Video Coding. Authors should mention it in the background of their work, e.g.:

• S. S. Bhattacharyya, J. Eker, J. W. Janneck, C. Lucarz, M. Mattavelli, M. Raulet, Overview of the mpeg reconfigurable video coding framework, Jrnl. of Signal Proc. Systems 63 (2).

Considering also that, in that field, reconfiguration has already been tackled either at a finegrain (FPGA only):

• E. Bezati, S. CasaleBrunet, M. Mattavelli, J. Janneck, Synthesis and optimization of highlevel

Stream programs, in: Electronic System Level Synthesis Conf., 2013, pp. 16.

• J. Janneck, I. Miller, D. Parlour, G. Roquier, M. Wipliez, M. Raulet, Synthesizing Hardware from Dataflow Programs: An MPEG4 Simple Profile Decoder Case Study, Jrnl. of Signal Proc. Systems 63 (2) 241249.

and at coarsegrain (including, but not limited to, FPGA implementations):

• Nezan J, Siret N, Wipliez M, Palumbo F., Raffo L (2012). MultiPurpose

Systems: A Novel DataflowBased Generation and Mapping Strategy. Symp on Circuits and Systems (ISCAS), pp. 30733076

• C. Sau, L. Raffo, F. Palumbo, E. Bezati, S. CasaleBrunet, M. Mattavelli, Automated design flow for coarsegrained reconfigurable platforms: An rvccal multistandard decoder usecase, in: 2014 International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation, pp 5966.

**Response:** All mentioned references were included and used in the text (see references [17] to [24]), along with others related to the same area.

Methodology:

1. Is the proposed methodology limited to a specific vendor?

**Response:** No, it is not. The use of SVF and the information provided in the proposed SI information allows the use in a wide variety of platforms. In order to clarify that, a new section was included: V.D.

2. What happen, from the functional point of view, if the bitstream I receive is not compatible with the available hardware?

**Response:** If there is no match between the hardware reconfiguration bit-stream and the working FPGA, the update content is rejected and the receiver will then wait for a suitable file. Some text regarding that was added to section V.D.

3. How long the reconfiguration process take? Are there any timing limitation?

**Response:** The time period used by reconfiguration process is given by RCT, in table VIII. The remount time (RCT), that is, the time needed for acquiring the reconfiguration bit-stream, is also informed, in tables VII and VIII.

Results:

1. Which boards did you used for prototyping activities?

**Response:** The test environment was described in the second paragraph of section V.E.

2. Did you get any result on complete decoders?

**Response:** Unfortunately, we did not. There is no restriction regarding that, but the proposed example applications are suitable to give a good idea of complexity and the validity of the proposed methodology. This was mentioned in the third paragraph of the conclusions.

3. What is the reconfiguration overhead?

**Response:** There is some overhead related to bá blá blá. It was discussed in section V.D.

**Responses to the comments of Reviewer #4**

Comments to the Author

This work presents a methodology to change the behavior of hardware components of TV receivers using FPGA dynamic revonfiguration based on digital TV signal content. Several hardware behaviors are already synthesized into configuration bitstreams. These bitstreams are sent using the same packeting method as other broadcast information (video, audio etc ).

Remarks:

\*abstract

reassemblies >

reassembles

**Response:** It has been corrected.

\*Figure 3:

Video

"DENC" never explained before!

**Response:** DENC was defined in the paragraph above Figure 3.

Unify the use of demultiplexer and demultiplexer

**Response:** Now we use only demultuplexer.

\*5.D

"The above rates this range have similar values ..." sentence not clear

**Response:** It has been corrected and extended, with a richer explanation.

The FPGA used for implementation tests is not mentioned

**Response:** The test environment was described in the second paragraph of section V.E.

The FPGA frequency that gave the timing results is not mentioned

**Response:** The test environment was described in the second paragraph of section V.E.

\*Tables 7 and 8 have the same title !

**Response:** It has been corrected.

\*comparison with litterature

In related works, the transmitted signal is not changed and the use of local predefined hardware configurations in the receiver is applied. In comparison, does the presented approach need to modify in a certain way the broadcast standard? In that case, what is the impact on existing materials using such broadcast standard and is it simpler than continiously update the receiver with new configurations?

**Response:** Yes. Indeed, the proposal itself, as shown in section V.A, is based on an extension of the existing SI standards. In addition, it can also be done as a proprietary solution, given that it has no impact on current structures. However, the complete framework proposed here is elegant and leaves no loose ends regarding feature updates towards FPGA devices, in digital TV environments, while a myriad of proprietary solutions would require more bandwidth (in order to update all different platforms, even using the same FPGA devices) and are more time consuming to implement. A discussion regarding such matter was included in section V.D.