

An approach for FPGA implementation of LST-SW algorithm

Assaad El Makhloufi, Samir El Adib , Naoufal Raissouni

assaadmakhloufi@gmail.com, adibsamir@gmail.com, naoufal.raissouni.ensa@gmail.com }

Remote Sensing & Geographic Information System Unit (RS&GIS)
University Abdelmalek Essaadi, ENSA Tetouan, Morocco

Abstract. Satellite remote sensing (SRS) needs to make use of up-to-date technology. The trend in SRS missions has always been towards using hardware devices with smaller size, lower cost, more flexibility, and higher computational power. Therefore, Field Programmable Gate Array (FPGA) technology is highly used by the scientific community and hardware developers to implement different SRS algorithms. Thus, FPGAs offer well-suited architectural features; large number of programmable logic elements, distributed and block RAMs, DSP and register slices and look up tables. This paper describes an approach to the implementation of the Land Surface Temperature Split-Window (LST-SW) algorithm structure based on FPGA technology.

Keywords: FPGA, Remote sensing, Virtex, VHDL, LST-SW.

1 Introduction

Recently, Field Programmable Gate Array (FPGA) technology has become a viable target for the implementation of different algorithms in different fields. These developments are offering new opportunities especially in satellite remote sensing. Digital sensors mounted onboard the remote sensing satellites scan vast areas of the earth's surface during day and night and beam this data to the satellite ground stations for further processing and usage in different applications: Digital Signal Processing (DSP), video processing, control systems engineering, bioinformatics, aerospace and defense systems. With their continuous development and improvement, remote sensing satellites [1] are increasingly being utilized and use of the latest available technological components. The emergence of peripherals such as Field Programmable Gate Arrays (FPGAs) [2], which can bridge the gap in on-board and real-time analysis of remote sensing data. Some FPGAs incorporate a large number of arithmetic blocks that can be low-complexity blocks such as simple multipliers or can be relatively more complex like the Digital Signal Processing (DSP) units which consist of combinations of various components like multipliers, adders, accumulators, shift registers. A DSP unit significantly accelerates the FPGA's performance and allows achieving greater productivity and flexibility, while decreasing cost and power consumption [3].

FPGAs are fully reconfigurable, a technological feature that combines the flexibility of traditional microprocessors with the performance of Application Specific Integrated Circuits (ASIC) devices [4]. Also, the reconfigurable hardware was certified by international remote sensing agencies, which was frequently used in remote sensing missions and space-borne earth observation missions [5]. FPGA also offers exact results with compact size, and low cost which makes the reconfigurable system interesting for onboard data processing.

The FPGA architecture makes it possible to implement any combinational and sequential circuit, which can range from a simple logic function to a high-end soft-processor [6]. In the past, FPGAs have been mainly used in signal processing and network packet analysis. However, thanks to the high-speed embedded resources included in the FPGA such as DSP slices and fast memories, they are now also utilized for algorithm acceleration either as coprocessors or standalone systems, System-On-Chips (SoCs). Moreover, several vendors reserve space in the FPGAs to custom soft hard-IP (HIPs) processors.

Recently, FPGAs have become a viable target technology for implementation different algorithm in remote sensing such us: A Real-Time Marker-Based Visual Sensor Based on a FPGA and a Soft-Core Processor [7], FPGA Implementation of an Algorithm for Automatically Detecting Targets [8], FPGA Implementation of the N-FINDR Algorithm [9], On-Board Ortho-Rectification for images Based on an FPGA [10], FPGA Implementation of JPEG-LS Remote Sensing Image Coding Algorithm [11]. These computing systems combine the flexibility of general-purpose processors with the speed of application-specific processors. The objective of this work is to develop an FPGA-based hardware version of the Land Surface Temperature Split-Window (LST-SW) algorithm structure [12] on FPGA.

As image sizes and bit depths grow larger, software has become less useful in the images processing realm. Recently developed hybrid FPGAs, such as the Xilinx Virtex-5 [13], offer the versatility of running diverse software applications on embedded processors while at the same time taking advantage of reconfigurable hardware resources, all on the same chip package. In fact, radiation hardened FPGAs are in great demand for military and space applications [14] [15]. Xilinx FPGAs have been used in more than 50 missions. In this work, we use a Xilinx Virtex-5 XC5VLX50T FPGA as a baseline architecture since it is similar to the existing FPGAs [8] that have been certified by several international agencies for remote sensing applications. They are based on the same architecture so we could immediately implement our design on them.

2 Role of FPGA in satellite remote sensing applications

The trend in remote sensing missions has always been towards using hardware devices with smaller size, lower cost, more flexibility, and higher computational power [16] [17]. Reconfigurable Hardware (RH) provides a flexible medium to implement hardware circuits.

The RH resources are reconfigurable post-fabrication, allowing a single-based hardware design to implement a variety of circuits. FPGAs can be reconfigured to avoid hardware faults [18], whether they result from fabrication or the environment.

Reconfigurable hardware offers the necessary flexibility and performance with reduced energy consumption compared to other high-performance processors. By mapping functionality to FPGAs, the computer designer can optimize the hardware for a specific application resulting in acceleration rates of several orders of magnitude over general purpose computers. Furthermore, these devices are characterized by lower form/wrap factors compared to parallel platforms and by higher flexibility than ASIC solutions. Moreover, satellite-based remote sensing instruments can only include chips that had been certified for space conditions. This is because space-based systems must operate in an environment in which radiation effects

have an adverse impact on integrated circuit operation [19]. Ionizing radiation can cause soft-errors in the static cells used to hold the configuration data. This will affect the circuit functionality and can cause system failure. Therefore, it requires special FPGAs that provide on-chip reconfiguration error-detection and/or correction circuitry. High-speed, radiation-hardened FPGA chips with million gate densities have recently emerged can support the high throughput requirements for the remote sensing applications. Radiation-hardened FPGAs are in great demand for military and space applications. For instance, industrial partners such as Microsemi Corporation or Xilinx (www.xilinx.com) have been producing radiation-tolerant antifuse FPGAs for several years for high-reliability space-flight systems. Microsemi FPGAs have been on board more than 100 launches and Xilinx FPGAs have been used in more than 50 missions.

3 Description of LST Split-Window algorithm

Land Surface Temperature (LST) has a significant role in the land surface characters on local and global scale and his one of most key parameters in the biophysics of land surface processes [20]. LST plays an important role in the cycle of the natural ecosystem evolution and global change. LST determination from satellite data is mainly influenced by the atmosphere and surface emissivity.

LST is defined as the surface radiometric temperature corresponding to the instantaneous field-of-view of the sensor or more precisely [21], as the `ensemble directional radiometric surface temperature [22]. The term "ensemble" depicts the bulk contribution of an inhomogeneous pixel. For a given sensor viewing direction, LST depends on the distribution of temperature and emissivity within a pixel and the spectral channel of measurement [23].

In our case, the LST Split Window (LST-SW) algorithm proposed by (Sobrino J. & Raissouni N) has been used for implementation purpose on FPGA (see eq. 1):

$$LST = T_4 + 1.40 (T_4 - T_5) + 0.32 (T_4 - T_5)^2 + 0.83 + (57 - 5W) (1 - \epsilon) - (161 - 30W) \Delta \epsilon . \quad (1)$$

where T_4 and T_5 in (K) are the brightness temperatures measured in the Advanced Very High-Resolution Radiometer (AVHRR) Channels 4 and 5 on board the National Oceanic and Atmospheric Administration (NOAA) satellite series, W is the total amount of atmospheric water vapor in ($g\ cm^{-2}$). ϵ and $\Delta \epsilon$ are, respectively, the average effective emissivity in both channels and the spectral variation of emissivity. Also, T_4 , T_5 , W , ϵ are binary images contains pixels of images coded in 16 bits.

4 Proposed Hardware FPGA architecture for the LST-SW algorithm

The architecture proposed of the hardware used to implement the LST algorithm (see **Figure 1**), in our implementation we have devices our algorithm in two parts (see equations (2) and equations (3)):

$$\text{Part 1(LST1)} = T_4 + 1.40 (T_4 - T_5) + 0.32 (T_4 - T_5)^2 + 0.83 . \quad (2)$$

$$\text{Part 2(LST2)} = (57-5W) (1- \epsilon) -(161-30\epsilon) \Delta\epsilon . \quad (3)$$

$$\text{LST} = \text{LST1} + \text{LST2}. \quad (4)$$

For data input, we use a hard disk (PC) to store our binary data. In addition, our architecture contains two FIFO: the first FIFO to transmit the binary images T_4 , W and also the result of the first part. The second FIFO to transfer the binary images T_5 , ϵ and the result of the second part. Finally, we transmit each part in the two FIFOs: FIFO 1 (part1) and FIFO 2 (part2). The LST module is used to implement our version of the LST algorithm.

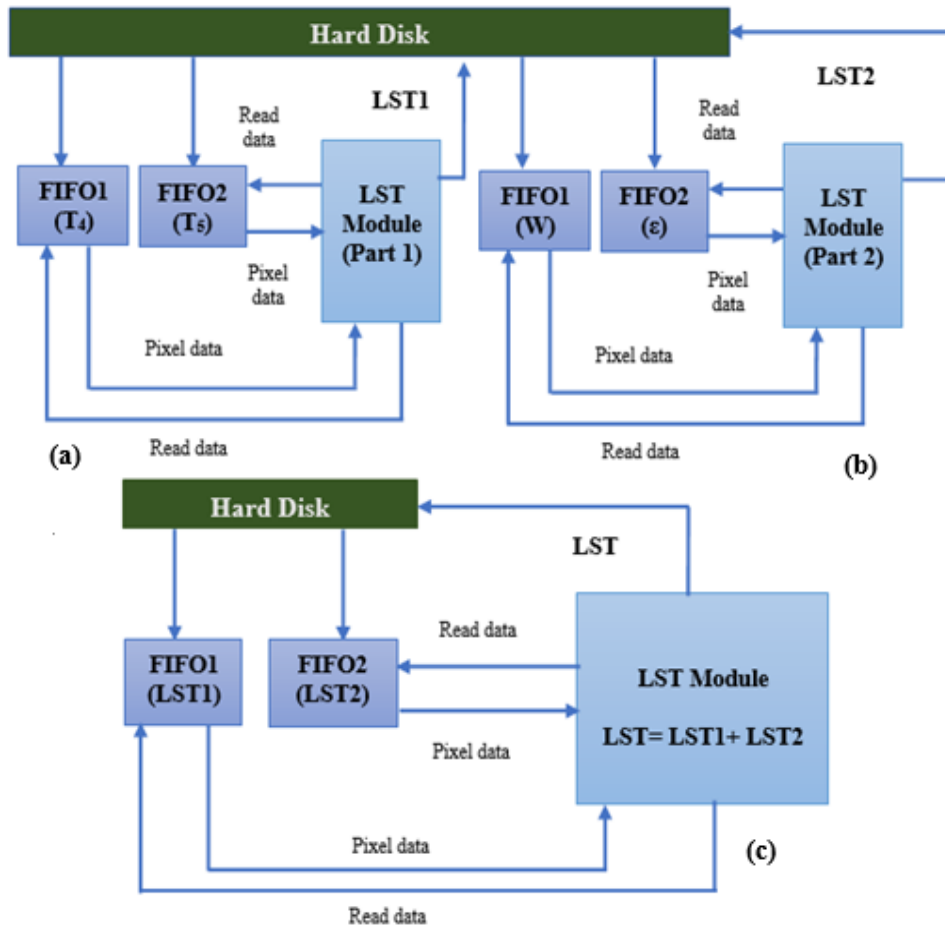


Fig. 1. Hardware architecture proposed for implementation of the LST S-W algorithm

To calculate LST1 we send two images: T_4 through FIFO1 and T_5 into FIFO 2 then the calculation happened in the LST module (see **Figure 1. [a]**). The last result of part1 stored in the hard disk. For LST2 we send two images: W through FIFO1 and ϵ into FIFO 2 then the calculation happened in the LST module (see **Figure 1. [b]**). The last result of part2 stored in the hard disk. Finally, to calculate LST we send both parts that stored in memory through the

two FIFOs: LST1 through FIFO1 and LST2 into FIFO2 (see **Figure 1. [c]**) then we add this result and send it to the hard disk.

LST hardware module is the heart of our system, and we took special care in its definition using VHDL as the hardware description Language. This module has set as top module of our system, because he contains different hardware part such us, registers, arithmetic-logic units. Also, all the calculations of our algorithm are done in this module: addition, subtraction, multiply. In our implementation, we multiply 100 with the coefficient of the algorithm (see equation (4) and (5)) to avoid the floating-point and their difficulty in their implementation, and when we got the result, we divided the coefficients by 100 to get back to our equations.

$$\text{Part 1(LST1)} = 100 \times T_4 + 140 (T_4 - T_5) + 32 (T_4 - T_5)^2 + 83 . \quad (5)$$

$$\text{Part 2(LST2)} = 100 \times (57 - 5W) (1 - \epsilon) - 100 \times (161 - 30\epsilon) \Delta \epsilon . \quad (6)$$

The LST-SW algorithm has been described using VHDL (VHSIC Hardware Description Language), and the design has been synthesized and implemented using Xilinx ISE 9.2i tool. As we said, T_4 , T_5 , W , ϵ are binary images generated by ENVI software in form of. Png files, but our Xilinx ISE cannot support this type of files. A program was written using MATLAB to support design verification through file manipulation. Therefore, we have written a program to convert this png files to text file format which was read by the VHDL.

5. Implementation results

5.1 Input Data

In our proposed implementation we have used 4 images (format: 512 columns, 512 rows). we chose to work with images similar to the real satellite images (T_4 , T_5 , W , ϵ) to check the operation of our system Then we have converted this image to text file contains pixels of images coded in 16 bits.

5.2 Test bench generation

The most time-consuming part of a system design of this nature is verifying that the VHDL code operates as intended. This is done by executing the device code in a simulator. The simulation of the device is only meaningful if the inputs to the system are manipulated in a way that causes the device to operate in its intended manner. This is easily done using VHDL to create a test bench for the entity under test. One of the major improvements made as test bench development progressed was the adoption of file I/O for loading processing parameters and image data into the system.

We have creating a test bench to circulate satellite data into system, which are the latter text format files contains pixel values that what we called the file Input/Output.

5.3 Register Transfer Level (RTL) schematic

Figure 2 show the Register Transfer Level of the first part and second part and the result of our algorithm, as we can see this figures shows the input signal which are the outlar

(T₄), outlar 2(T₅), outlar 3(W),outlar 4(ε), Also the LST1 and LST2 and they are data satellite as text file contain pixel coded in 16 bits, and the output that contain the result of the first part LST 1 and second part LST 2 and the LST result.

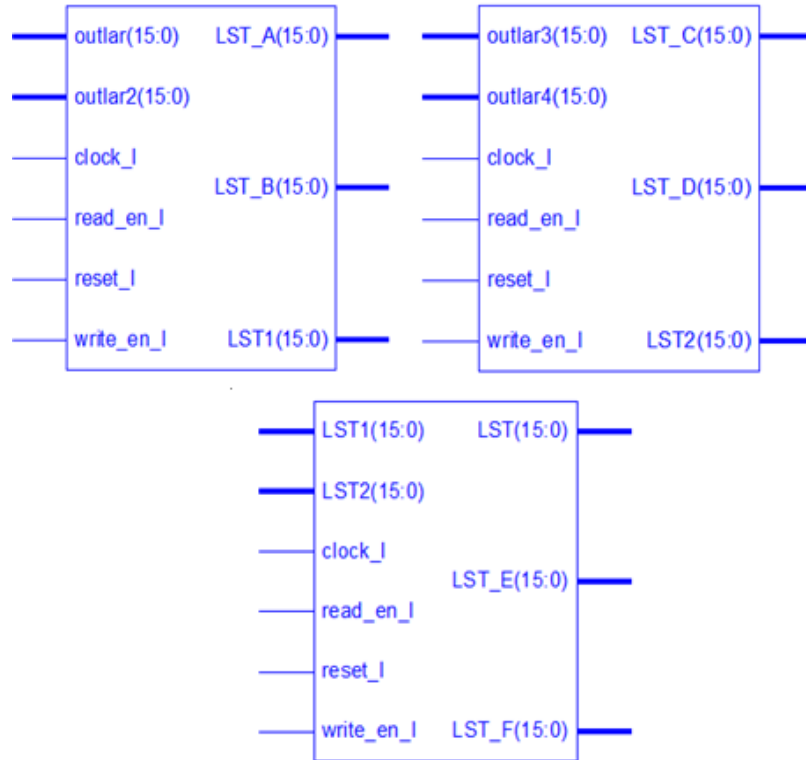


Fig. 2.Register Transfer Logic for the system

The **Figures 3,4** and **5** shows the simulation results of the proposed implementation of the LST-SW algorithm in each part. In this simulation we see all the data that be sending into system as binary number coded in 16 bits using the test bench for reads images file and it was written specifically to replace the acquisition of images. This data buffering by FIFO, it is a structure used in hardware application when we need to buffer a data and it's like a pipe where the first element entering into the pipe is the first element that output from the pipe. Therefore, for LST1 (see Figure 3), we send the first element of our image T₄ (outlar) into FIFO 1 and the first element of image T₅ (outlar2) into FIFO 2 and we calculated the first element it means the first pixel in each data together then we stored this result in the first line, this operation continue until the last element.

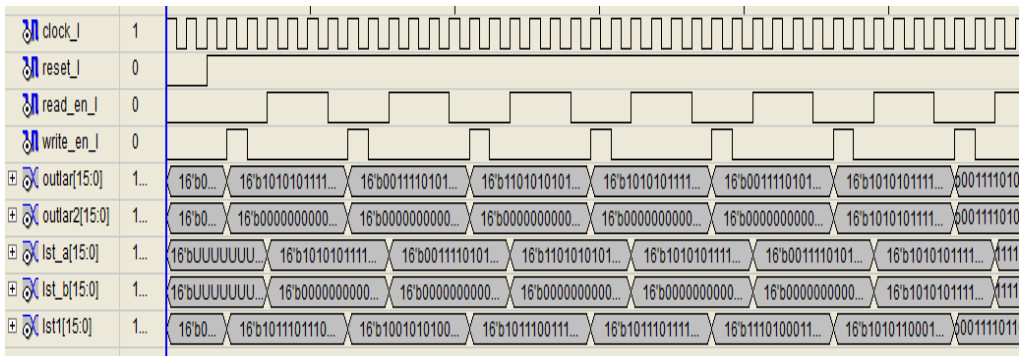


Fig. 3.Simulation Results for the LST1

The same way, to calculate LST2 (see Figure 4), we send the first element of our image W (outlar3) into FIFO 1 and the first element of ϵ (outlar4) into FIFO 2 and we calculate the first element it means the first pixel in each data together then we stored this result in the first line, this operation continues until the last element.

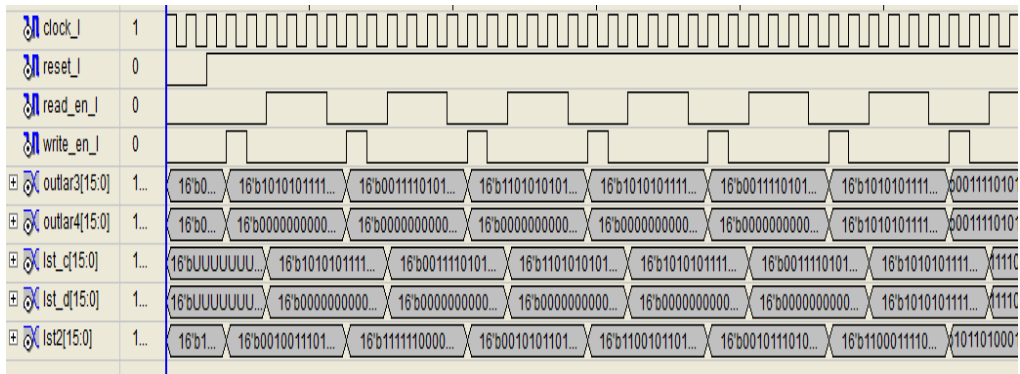


Fig. 4.Simulation Results for the LST2

Figure 5 shows the result of the addition of the two parts, with the same way, we transfer the result of the LST1 into FIFO 1 and the result of LST 2 into FIFO 2, this calculation is done like the earlier ones, we add the first element pixel in each FIFO and we memorized, this operation continues until the last pixel.

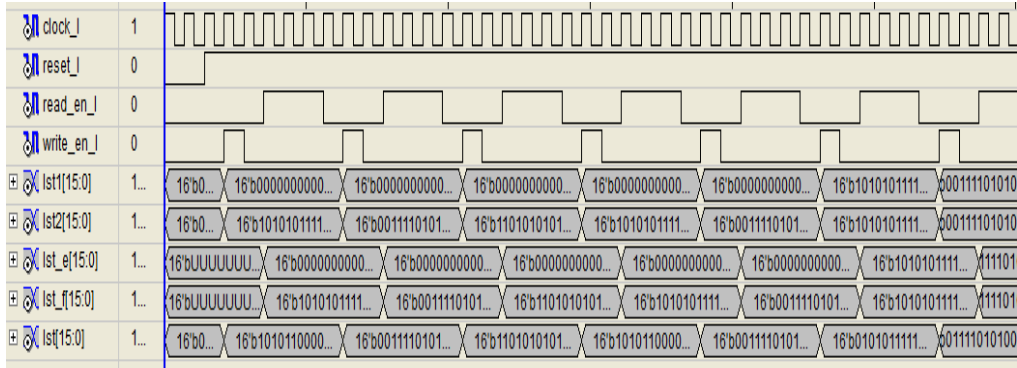


Fig. 5.Simulation Results for the LST

5.4 LST-SW FPGA implementation results

In this subsection, we conduct an experimental result of the computational performance of the proposed FPGA implementation. **Table 1** shows the resources used for our hardware implementation of the proposed LST algorithm design, conducted on the Virtex-5 LX50T FPGA. This FPGA has a total of 4706 slices, 28800 slice registers, and 28800 four input look-up tables available. In addition, the FPGA includes some heterogeneous resources, such as 48 DSP48s. In our implementation, we took advantage of these resources to optimize the design, The Block Slice Registers are used to implement the FIFOs without using block RAMs, with others slices are used for the implementation of the LST-SW algorithm with the DSP48s multiplier.

Table 1. Summary of resource utilization for the FPGA-based implementation of the LST algorithm.

Target FPGA	Virtex-5 LX50T		
	Used	Total	%
Device utilization			
Number of Slice Registers	3306	28800	11
Number of Slice LUTs	2231	28800	7
Number of Slices	222	7200	3
Number of bonded IOBs	84	480	17
Number of DSP48Es	27	48	56
Maximum Frequency	324.657 Mhz		

Logically, the number of DSP is higher because of the existence of the calculation in each part (LST1 and LST2) that consumes much of the resources of the FPGA.

6. Conclusion

The choice of an architecture for the embedded images processing is not trivial and that a compromise must certainly be found between flexibility, consumption, performance, cost and speed of design in terms of TTM (Time-To-Market). Also, the number of remote sensing applications requiring fast response of algorithm analysis has been growing exponentially in

recent years. In this paper, we have proposed an architecture for the FPGA implementation of the Land Surface Temperature Split Window (LST-SW) algorithm, one of the most well-known approaches for determination of Land Surface Temperature from satellite data in the remote sensing community. The system was designed using VHDL in a high level design method. All part of the design has been simulated and implementing using Xilinx tools with target FPGA Virtex-5 LX50T.

References

- [1] Roy,S., Behera,D.M.,Srivastav,K,S.: Satellite Remote Sensing: Sensors, Applications and Techniques. Vol. 87, pp 465–472. Sciences, INDIA (2017)
- [2] Yang,C. , Li,B., Chen ,L., Wei,C.: A Spaceborne Synthetic Aperture Radar Partial Fixed-Point Imaging System Using a Field- Programmable Gate Array—Application-Specific Integrated Circuit Hybrid Heterogeneous Parallel Acceleration Technique. Sensors (MDPI), (2017)
- [3] Yaseer,D.,Teresa,R.: Power estimation technique for DSP architectures. Digital Signal Processing. Vol 19, pp. 213-219 (2009)
- [4] Singh,G.: Reconfigurable Computing A review of the technology and its architecture. Journal of VLSI and Signal Processing. Vol 3, pp. 08-14 (2013)
- [5] Gonzalez,C., Resano,J., Mozos,D., Plaza,A., and Valencia,D.: FPGA Implementation of the Pixel Purity Index Algorithm for Remotely Sensed Hyperspectral Image Analysis. Journal on Advances in Signal Processing, Vol 10, pp. 13 (2010)
- [6] García,G., Jara,C., Pomares,J., Alabdo,A., Poggi,L., Torres,F. A Survey on FPGA-Based Sensor Systems: Towards Intelligent and Reconfigurable Low-Power Sensors for Computer Vision, Control and Signal Processing. Sensors (MDPI), (2014)
- [7] Tayara, H., Ham,W., Chong,K.: A Real-Time Marker-Based Visual Sensor Based on a FPGA and a Soft-Core Processor.Sensors(MDPI). Vol 16, (2016)
- [8] Gonzalez,C., Bernabé,S., Mozos,D., Plaza,A.: FPGA Implementation of an Algorithm for Automatically Detecting Targets. IEEE journal of selected topics in applied earth observations and remote sensing. Vol 9,pp .13 (2016)
- [9] Gonzalez,C., Resano,J., Mozos,D., Plaza,A.: FPGA Implementation of the N-FINDR Algorithm for Remotely Sensed Hyperspectral Image Analysis. IEEE transactions on geoscience and remote sensing.Vol 50, (2012)
- [10] Zhou,G., Zhang,R., Liu,N., Huang,J., Zhou,X.: On-Board Ortho-Rectification for Images Based on an FPGA. Remote Sensing (MDPI). Vol 9, (2017)
- [11] Wang,H.: FPGA Implementation of JPEG-LS Remote Sensing Image Coding Algorithm. Advances in Engineering Research.Vol 150,(2017)
- [12] Sobrino,J, Raissouni,N.: Toward remote sensing methods for land cover dynamic monitoring: Application to Morocco.International Journal of Remote Sensing. Vol 21,(2000)
- [13] Bulens,P.,Standaert,F.,Pellegrin,P.:Implementation of the AES-128 on Virtex-5 FPGAs.Progress in cryptology.pp. 16-26 (2008)
- [14] Kramer,H., Cracknell, A.: An overview of small satellites in remote sensing, International Journal of Remote Sensing, Vol 29, (2008)
- [15] Lysaght, P., Blodget, B., Mason, J., Young, J., Bridgford, B.: Enhanced architectures, design methodologies and CAD tools for dynamic reconfiguration of Xilinx FPGAs. Proc. Int. Conf. Field Programmable Logic Appl. pp. 1–6 (2006)
- [16] Heinz, D., Chang,C.:Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery. IEEE Trans. Geosci. Remote Sensing. Vol. 39, pp. 529–545 (2001)
- [17] Plaza, A., Martinez, P., Perez, R., Plaza, J.: A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data. IEEE Trans. Geosci. Remote Sensing. vol. 42, pp. 650–663 (2004)

- [18] Laplante, P.: Computing requirements for self-repairing space systems, *Journal of Aerospace Computing, Information and Communication*. vol 2, pp. 154–169 (2005)
- [19] Neil, B., Dawood, A.: Reconfigurable computers in space: Problems solutions and future directions. In *Proc. Military Aerosp. Appl. Programmable Logic Devices Conf.*, pp. 1–2 (1999)
- [20] Morrow, N., Friedl, MA.: Modelling biophysical controls on land surface temperature and reflectance in grasslands. *Agricultural and Forest Meteorology*. Pp. 147-161 (1998)
- [21] Prata, F., Sobrino, J.: Thermal Remote Sensing of Land Surface Temperature From Satellites: Current Status and Future Prospects. *Remote Sensing Reviews*. Pp. 175-224 (1995)
- [23] Norman, J., Becker, F.: Terminology in thermal infrared remote sensing of natural surfaces *Reviews*. pp. 159-173 (1995)
- [23] Becker, F., Li, Z.: Surface temperature and emissivity at various scales: Definition, measurement and related problems. *Remote Sensing Reviews*. vol 10, pp. 225-253 (1995)