

1-1-2011

# Reconfigurable platform for 3D-panoramic telepresence system for mobile applications

Artur Saakov  
*Ryerson University*

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>

 Part of the [Electrical and Computer Engineering Commons](#)

---

## Recommended Citation

Saakov, Artur, "Reconfigurable platform for 3D-panoramic telepresence system for mobile applications" (2011). *Theses and dissertations*. Paper 758.

# **RECONFIGURABLE PLATFORM FOR 3D-PANORAMIC TELEPRESENCE SYSTEM FOR MOBILE APPLICATIONS**

By

**Artur Saakov**

B.A.Sc., Moscow State Institute of Radio Engineering, Electronics and Automation  
(Technical University), Russian Federation, 2007

A Thesis

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Applied Science

in the Program of

Electrical and Computer Engineering

Toronto, Ontario, Canada, 2011

© Artur Saakov, 2011

**Declaration of Authorship**

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

# RECONFIGURABLE PLATFORM FOR 3D-PANORAMIC TELEPRESENCE SYSTEM FOR MOBILE APPLICATIONS

Artur Saakov  
Master of Applied Science  
Electrical and Computer Engineering  
Ryerson University 2011

## **Abstract**

The concept of telepresence allows human beings to interact with hazardous environments and situations without facing any actual risks. Examples include the nuclear industry, outer space and underwater operations, mining, bomb disposal and firefighting. Recent progress in digital system technology, especially in technology of reconfigurable logic devices (e.g. FPGA), allows the effective implementation of advanced embedded systems characterized by high-performance data processing and high-bandwidth communication. However, most of the existing telepresence systems do not benefit from these advancements. Therefore, the goal of this work was to develop a concept and architecture of the platform for the 3D-Panoramic Telepresence System for mobile robotic applications based on reconfigurable logic devices. During the development process, two versions of the system were implemented. The first system focused on feasibility testing of major components of the proposed architecture. Based on the experimental results obtained on the first prototype of the system and their analyses, a set of recommendations were derived for an updated version of the system. These recommendations were incorporated into the implementation of the second and final version of the system.

**Keywords:** 3D Panoramic Telepresence, FPGA, multi-stream processing, reconfigurable computing, stereo vision.

## **Acknowledgments**

There are many people I would like to thank for making this thesis possible. Firstly, I would like to thank my supervising professor Dr. Lev Kirischian for providing his knowledge, immeasurable experience, support and guidance in all aspects of my M.A.Sc studies.

I would also like to express personal gratitude to Valeri Kirischian for his influence since the beginning of my studies here at Ryerson and for sharing his engineering experience which proved to be very practical and supportive throughout.

Thirdly, I would like to thank the Department of Electrical and Computer Engineering at Ryerson for their great facilities and numerous resources needed for my research, as well as the review committee for their participation.

Fourthly, appreciation goes out to Victor Dumitriu for providing advice and opinions at times needed the most.

Lastly, and most importantly, I would like to express profound gratification to my family for all of the unconditional support they gave me every step of the way; throughout all of my goals and achievements.

## Table of Contents

1. INTRODUCTION .....	1
1.1. Motivation .....	1
1.2. Objectives .....	3
1.3. Contributions .....	3
1.4. Thesis organization.....	5
2. COMPREHENSIVE SURVEY OF TELEPRESENCE SYSTEMS FOR 3D-PANORAMIC MOBILE APPLICATIONS.....	6
2.1. Introduction .....	6
2.2. Human Vision System Organization and Operation .....	10
2.3. Telepresence Systems in Hazardous Environments .....	13
2.4. Machine Vision Systems .....	14
2.5. Stereoscopic Display Systems .....	20
2.6. Video Compression/Decompression Mechanisms and Standards .....	26
2.7. Real-Time Video Processors .....	35
2.8. Conclusion.....	39
3. ARCHITECTURE DEVELOPMENT OF RECONFIGURABLE PLATFORM FOR 3D-PANORAMIC TELEPRESENCE SYSTEM.....	42
3.1. Introduction .....	42
3.2. Analysis of Operation Modes.....	43

3.3.	System Architecture Organization .....	45
3.3.1.	3D-P Camera.....	47
3.3.2.	Video Compressor/Decompressor .....	49
3.3.3.	Transceiver.....	55
3.3.4.	Display .....	57
3.3.5.	Transport Stream Combiner/Dispatcher .....	59
3.3.6.	Video Pre-Processor/Video Processor .....	61
3.4.	Conclusion.....	69
4.	IMPLEMENTATION OF THE PLATFOM FOR 3D-PANORAMIC TELEPRESENCE SYSTEMS.....	71
4.1.	Introduction .....	71
4.2.	System Design Implementation (stage 1).....	72
4.2.1.	4-Vision Subsystem .....	73
4.2.2.	MARS Subsystem .....	100
4.3.	System Design Implementation (stage 2).....	114
4.3.1.	Selection of the Proposed Elements for the System Implementation .....	115
4.3.2.	System Hardware Architecture .....	120
4.4.	Conclusion.....	123
5.	EXPERIMENTS AND RESULTS.....	124
5.1.	Introduction .....	124

5.2.	Experimental Setup .....	124
5.3.	Experimental Results Analysis.....	131
5.3.1.	Startup Time Analysis.....	131
5.3.2.	Video Acquisition and Packet Transfer Time Analysis .....	134
5.3.3.	Packet Reception and Frame Buffering Time Analysis .....	136
5.3.4.	Analysis of Bayer Color Decoding and RGB Pixels Output Timing.....	137
5.4.	Compression Rate Analysis.....	140
5.5.	Power Consumption Analysis .....	145
5.6.	Utilization of Experimental Results in 4-Vision2 Subsystem Design .....	147
5.7.	Conclusion.....	149
6.	CONCLUSION AND FUTURE WORK .....	151
	BIBLIOGRAPHY.....	155



## List of Figures

Figure 2.1: Human Eye Organization .....	11
Figure 2.2: Human Visual Fields .....	12
Figure 2.3: Stereo Vision Head System .....	15
Figure 2.4: Stereo Vision Systems .....	16
Figure 2.5: Pseudo-Stereo Vision System.....	17
Figure 2.6: Line-Scan Sensor Rotation System .....	18
Figure 2.7: 2D Panoramic System Based on Panning Technique.....	18
Figure 2.8: Multiple Camera Panoramic System .....	19
Figure 2.9: CCD Camera and Curved Mirror Panoramic System .....	20
Figure 2.10: Rotating Mirror Architecture for Panoramic Vision .....	20
Figure 2.11: Linear Polarization Scheme.....	22
Figure 2.12: Circular Polarization Scheme .....	23
Figure 2.13: Autostereoscopic Displaying Schemes.....	25
Figure 2.14: MPEG GOP Sequence.....	29
Figure 2.15: JPEG2000 Encoder Flow .....	34
Figure 3.1: Telepresence System Architecture .....	45
Figure 3.2: Video-Acquisition and Pre-Processing Subsystem Block-Diagram .....	46
Figure 3.3: Video-Processing Subsystem Block-Diagram.....	46
Figure 3.4: Camera Modules Organization.....	48
Figure 3.5: Image Sensor Configuration Controller Block-Diagram.....	49
Figure 3.6: Resistance to Bit Errors of Various Compression Methods.....	52
Figure 3.7: Image Compressor/Decompressor Controller Block-Diagram .....	55
Figure 3.8: Transceiver Block-Diagram .....	56

Figure 3.9: Transceiver Operation Principles .....	57
Figure 3.10: Stereo-Panoramic Display .....	58
Figure 3.11: Transport Stream Packet.....	59
Figure 3.12: Transport Stream Combiner Block-Diagram .....	60
Figure 3.13: Transport Stream Dispatcher Block-Diagram .....	61
Figure 3.14: Video Acquisition Subcomponent Block-Diagram.....	62
Figure 3.15: Bayer Color Pattern .....	64
Figure 3.16: Bilinear Interpolation Algorithm .....	64
Figure 3.17: Color Interpolation Subcomponent .....	65
Figure 3.18: Storage Memory Controller Subcomponent Block-Diagram.....	66
Figure 3.19: Display Driver Subcomponent Block-Diagram .....	67
Figure 3.20: Image Compressor/Decompressor Controller Subcomponent Block-Diagram .....	68
Figure 4.1: Architecture of 3D-P Telepresence System .....	72
Figure 4.2: 3D-P Camera Organization .....	74
Figure 4.3: Image Sensor General Timing Diagram.....	78
Figure 4.4: Camera Module Interface Connector .....	79
Figure 4.5: 4-Vision Subsystem Hardware Architecture .....	80
Figure 4.6: FPGA Based Implementation Diagram.....	82
Figure 4.7: Microprocessor Based Implementation Flow Chart.....	83
Figure 4.8: Microprocessor Instruction Execution Process .....	84
Figure 4.9: 4-Vision Hardware Subsystem Architecture .....	88
Figure 4.10: 4-Vision Clock Distribution Scheme.....	93
Figure 4.11: Pre-Processor Component Symbol.....	94
Figure 4.12: Pre-Processor Component Organization .....	95

Figure 4.13: 3D-P Camera Configuration Component Symbol.....	98
Figure 4.14: 3D-P Camera Configuration Component Operation Flow Chart .....	98
Figure 4.15: MARS Hardware Subsystem Architecture.....	102
Figure 4.16: MARS Clock Distribution Scheme .....	103
Figure 4.17: MARS Subsystem Data Flow .....	104
Figure 4.18: Storage of Pixels in SRAM Memory.....	105
Figure 4.19: XGA Timing Specification.....	107
Figure 4.20: Processor Component Symbol.....	109
Figure 4.21: Processor Component Organization .....	111
Figure 4.22: Bayer Color Decoder Organization .....	113
Figure 4.23: General Architecture of the Video-Acquisition and Pre-Processing Subsystem .....	115
Figure 4.24: General Architecture of the Video-Stream Processing and Displaying Subsystem .....	116
Figure 4.25: 4-Vision Hardware Architecture .....	121
Figure 4.26: 4V-J2C Hardware Architecture .....	123
Figure 5.1: Experimental Setup of 3D-Panoramic Telepresence System .....	125
Figure 5.2: Image of the Experimental Setup .....	125
Figure 5.3: Image of the 4-Vision Subsystem.....	126
Figure 5.4: Image of the MARS Subsystem .....	128
Figure 5.5: DC-DC Converters Startup Time .....	131
Figure 5.6: 3D-P Camera Synchronous Operation .....	132
Figure 5.7: 4-Vision Operation Timing Diagram .....	135
Figure 5.8: Packet Reception and Frame Buffering Processes .....	136
Figure 5.9: Pixel Distribution to the Bayer Color Decoders .....	138

Figure 5.10: RGB Components Output to VGA.....	139
Figure 5.11: Original Sample RAW Frame .....	140
Figure 5.12: Compression performance.....	142
Figure 5.13: Image Quality in Accordance to JPEG2000 Compression Rate .....	143
Figure 5.14: Image of the 4-Vision2 Subsystem.....	148
Figure 5.15: Image of 4V-J2C Subsystem .....	149

## List of Tables

Table 2.1: Machine Vision Systems Summary .....	40
Table 3.1: Wireless Standards.....	51
Table 4.1: Image Sensor Signal Lines .....	79
Table 4.2: 4-Vision Subsystem Operational Frequencies .....	92
Table 4.3: Image Sensor Configuration Registers .....	99
Table 4.4: MARS Subsystem Operational Frequencies.....	103
Table 5.1: Startup Timing Summary.....	133
Table 5.2: Row Number to Font Size Correspondence.....	142
Table 5.3: Power Consumption Chart.....	146

## List of Acronyms

**3D-P** – 3D-Panoramic  
**ALU** – Arithmetic Logic Unit  
**ASD** – Autostereoscopic Display  
**ASIC** – Application-Specific Integrated Circuit  
**BER** – Bit Error Rate  
**BRAM** – Block Random Access Memory  
**CCD** – Charge-Coupled Device  
**CFA** – Color Filter Array  
**CMOS** – Complementary Metal Oxide Semiconductor  
**COFDM** – Coded Orthogonal Frequency Division Multiplexing  
**CPU** – Central Processing Unit  
**CR** – Compression Ratio  
**CU** – Control Unit  
**DCT** – Discrete Cosine Transform  
**DEMUX** – De-multiplexer  
**DF** – Data Fetch  
**DLP** – Data Level Parallelism  
**DSP** – Digital Signal Processor  
**DVB** – Digital Video Broadcasting  
**DWT** – Discrete Wavelet Transform  
**EBCOT** – Embedded Block Coding with Optimized Truncation  
**FMC** – FPGA Mezzanine Connector  
**FMO** – Flexible Macroblock Ordering  
**FOV** – Field of View  
**FPD** – Flat Panel Displays  
**FPGA** – Field-Programmable Gate Array  
**FW** – Firmware  
**GOP** – Groups of Pictures  
**GPU** – Graphics Processing Unit  
**HD** – High Definition  
**HDL** – Hardware Description Language  
**HMD** – Head Mounted Display  
**HQ** – High Quality  
**HW** – Hardware  
**ID** – Instruction Decode  
**IF** – Instruction Fetch  
**ILP** – Instruction-Level Parallelism  
**ISO** – International Organization for Standardization  
**IVT** – Interactive Virtual Telepresence  
**JPEG** – Joint Photographic Experts Group  
**JTAG** – Joint Test Action Group  
**LC** – Liquid Crystal  
**MARS** – Multi-stream Adaptive Reconfigurable System

**MIMD** – Multiple Instruction Multiple Data  
**MPEG** – Moving Picture Experts Group  
**MUX** – Multiplexer  
**OLED** – Organic Light Emitting Diode  
**PC** – Personal Computer  
**PCB** – Printed Circuit Board  
**PIC** – Peripheral Interface Controller  
**PSVS** – Pseudo-Stereo Vision System  
**RCS** – Reconfigurable Computing System  
**RF** – Radio Frequency  
**RISC** – Reduced Instruction Set Computing  
**ROI** – Region of Interest  
**RTL** – Register Transfer Level  
**SIMD** – Single Instruction Multiple Data  
**SISD** – Single Instruction Single Data  
**SoC** – System on Chip  
**SRAM** – Synchronous Random Access Memory  
**SW** – Software  
**TDM** – Time Division Multiplexing  
**TFT-LCD** – Thin Film Transistor Liquid Crystal Display  
**TLP** – Thread-Level Parallelism  
**TVS** – Telepresence Viewing System  
**URV** – Underwater Robotic Vehicles  
**USB** – Universal Serial Bus

# 1. INTRODUCTION

## 1.1. Motivation

In the last decade, increasing need for stereo-vision, stereo-panoramic and even semi-spherical stereo-vision systems has been dictated by a wide variety of scientific and industrial applications. The examples of these applications are: telepresence systems for remote robotic operations and unmanned vehicles, telemedicine and telesurgery systems, 3D HDTV and multimedia, video-gaming and professional simulators, various applications for autonomous robots and intelligent machine vision systems. Most of these applications require multi-modal high-performance operations when different video processing algorithms are active in different periods of time. As well, data-structure (e.g. video-frame structure, compressed transport stream formats etc.) may also change dynamically. The above aspects in turn required dynamic adaptation of the video processing system on variations of the workload. Therefore, dynamically reconfigurable platforms became one of the most promising ways for implementation of the above video processing systems.

On the other hand, the progress in technology of reconfigurable logic devices made possible cost-effective implementation of the entire video-processing systems in one chip. This allowed in turn a dramatic increase of system performance and reliability as well as, reduction of system cost and power consumption. Nowadays, Field Programmable Gate Array (FPGA) devices allow not only deploying the large System-on-Chip (SoC) in one FPGA but also dynamic re-configuration of the FPGA logic. This ability allows implementation of Dynamically Reconfigurable Systems on Chip (DRSoC) and thus,



effective adaptation of the multi-modal multi-video-stream acquisition, processing and displaying systems on the basis of FPGA devices.

One of the most important applications for multi-modal and multi-video-stream processing systems is telepresence and teleoperation in environments where actual presence and operation is dangerous or even impossible for human beings. First of all, this is important for the remote monitoring and control of mobile systems working in harsh areas such as space, nuclear power plants, the ocean, underground pipelines, mining and many others. Another important application of this technology could be telemedicine and e-Health, where Immersive Interactive Virtual Telepresence (IVT) systems can be implemented effectively.

However, most of the existing video processing systems developed for the above applications are based on RISC processors or processors with vector processing accelerators (e.g. GPU: Graphic Processing Units). These processors are able to provide certain flexibility in adaptation to the variations of the workload but are limited in performance compared to Application Specific Integration Circuits (ASICs). On the other hand, ASICs can provide the highest performance but cannot change their functionality and thus, adapt to the variations in video-processing algorithms or data structure.

Therefore, the main goal for the presented work was the creation of effective high-performance multi-stream video-acquisition, processing and displaying platform on the basis of FPGA devices for immersive telepresence and teleoperation applications including prototyping, testing and analysis of performance parameters.

## 1.2. Objectives

The main objective of the work is to research and develop the reconfigurable platform for the immersive 3D-Panoramic (3D-P) Telepresence System for mobile applications.

This objective includes the following aspects:

- Extended literature search of existing approaches and methods used in designing of 3D-Panoramic vision systems;
- Develop an architecture of the reconfigurable multi-mode 3D-P Telepresence System;
- Design and implement the 3D-P Telepresence System;
- Analyze the experimental results associated with performance parameters of the considered class of telepresence systems and determine the set of recommendations for the design process.

## 1.3. Contributions

The following contributions were made during the period of the work:

- Extended literature search in the area of stereo, panoramic and stereo-panoramic telepresence systems was conducted. As a result, the analysis of the state-of-the-art for the class of the telepresence systems for mobile robotic applications was provided;
- The framework of the reconfigurable multi-mode 3D-P Telepresence System was developed including multi-stream acquisition, parallel video-processing and parallel

displaying subsystems. As a result, all levels of architecture including system level, on-board and on-chip levels were developed;

- The architecture of 3D-P Telepresence System was implemented in the first prototype. This implementation allowed for a collection of data of the system performance parameters. As a result, the analysis of system performance based on the obtained data was provided.
- Based on the performance analysis obtained from the first prototype, the set of recommendations for the design process of 3D-P Telepresence Systems for mobile applications was determined. The updated version of the system reflecting these recommendations was designed and prototyped.

The first system prototype was presented in the annual conference SVAR-2010: Space Vision and Advanced Robotics held at MDA Space Missions, Brampton, ON in June 2010. The presentation: “3D Panoramic Naturally Interactive Real-Time Telepresence System” [54] was awarded with the first prize for the best presentation and demo.

The updated system prototype was presented at the conference and exhibition “Discovery 11” held at Metro Toronto Convention Centre in May 2011.

## 1.4. Thesis organization

The presented thesis is organized in six chapters. Chapter 1 summarizes motivation, objectives, contributions and thesis organization of this work. The observation of the telepresence systems application domain opens up in Chapter 2. The following material in this chapter presents the main aspects of the human vision system organization followed by observation of existing telepresence systems solutions for hazardous environments and observation of various systems and mechanisms used in the considered class of applications. Chapter 3 presents the proposed architecture of the adaptive 3D-P Telepresence System for mobile applications. Prior to the architecture description, the detailed analysis of operation modes required for various operation tasks is described in detail. Following the mode analysis, the system architecture partitioned on hardware, firmware and software components is presented. 3D-P Telepresence System architecture implementation is performed in Chapter 4. This chapter material covers the details associated with implementation of two versions of the system including hardware, firmware and software development. Chapter 5 contains information about the experimental setup, experimental results and their analyses. This chapter also presents the recommendations for the modifications of the system architecture for the next version of 3D-P Telepresence System and describes the updated system prototype as well. The final chapter of the thesis, Chapter 6, presents the summary of the research and development work performed for the project. Future development work is also outlined.

## 2. COMPREHENSIVE SURVEY OF TELEPRESENCE SYSTEMS FOR 3D-PANORAMIC MOBILE APPLICATIONS

### 2.1. Introduction

The concept of telepresence system as well as the great importance of research in this domain was first underlined by Marvin Minsky in 1980 [1]. Since then, many attempts were made in order to create a system that would provide a sense of physical presence at a remote location to a human.

Visual information being an integral part of a human's ability to process sensory stimulus became a major motivation of using video information in telepresence systems. However, depending on particular telepresence application in addition to video information; audio and haptic might also be included.

Integration of visual system which can reproduce human vision in a remotely controlled mobile system made it possible to perform a wide range of tasks which were either dangerous for humans to be involved in or simply were impracticable. Tasks such as bomb disposal, demining operations, rescue operations, telesurgery, fire fighting, underwater operations and outer space operations can be grouped into three general categories of telepresence applications such as operation in hazardous environments, telemedicine and teleconferencing.

The teleconferencing exhibits a rapidly growing interest of such sectors of business, government, healthcare and education. This is due to economical and environmental reasons which can eliminate the need for transportation, thus saving time, money and energy. Teleconferencing can be described as a communication environment where multiple

geographically separated users are meeting together. A communication environment on each user site is realized by a video/audio-acquisition unit, processing unit, display unit and communication channel. Depending on a particular setup, different types of units as well as different numbers of them can be arranged to create a communication environment. The fact that teleconferencing does not assume any remote human operations resulted in majority of 2D implementations presented on the market. On the other hand, the majority of research is performed on 3D implementations.

Recent examples of commercial telepresence solutions such as Cisco's TelePresence [68] and HP's Visual Collaboration Studio [69] offer real-time High Definition (HD) video and High Quality (HQ) audio for the participants. However, both solutions do not produce stereoscopic views of the participants, thus, eliminating any immersive experience.

The research work performed by Feldmann et al. [2] presents real-time multi-camera 3D teleconferencing system. Each communication site is assumed to have two participants and two multi-view displays for remote parties. Data processing is performed on 5 PC clusters interconnected with high-bandwidth Infiniband network. The experimental results indicate that the system is capable to process 16 video streams in real-time 3D at 20fps.

Teleconferencing system proposed by Kauff et al. [3] is based on a concept of Augmented Reality (AR) which can be described as combination of real-world and computer generated Virtual Environment (VE). All conferees are captured by four video cameras and subsequently rendered such that they appear sitting around a shared virtual table. In order to realize such a system, a PCI-based multi-processor board was developed for PC terminals. The proposed setup from the author makes a seamless transition between the real and virtual world.

Another example of teleconferencing system is proposed by Sen Ma et al. [4]. The group developed panoramic a teleconferencing system based on the image mosaic principle. Two CMOS cameras are arranged to perform video acquisition. The Nios II embedded processor is programmed to perform image processing and consequent Ethernet transmission to a remote terminal. Experimental results indicate that the system satisfies the requirements for the teleconference situation.

The telemedicine applications can be divided on two types: remote medical consultations and remote surgeries [47]. The first aspect is quite similar to teleconferencing applications. As for remote surgeries, also called telesurgeries, it requires remote manipulations with a patient and with remote robotic equipment.

Telesurgery is developed to provide an access to advanced surgical care for patients irrespective of their geographical location [5], [48]. In a situation where a highly qualified surgeon is located in a different county than a patient, transfer time may result in added complications or even death. Therefore, to overcome this problem, Assisted Robotic Telepresence Surgery (ARTS) is being used. Robotic telemanipulators have statically positioned 2D or 3D vision system as well as different set of surgical tools integrated into a robotic system. A surgeon – teleoperator can perform local or remotely controlled robotic surgery while observing his manipulations via specialized monocular or binocular setup. Examples of telesurgery systems are “da Vinci” and “ZEUS”. Both of them demonstrate enhanced operation precision thus allowing surgeons to perform more complex and fine grain operations [6].

Paper presented by Eadie et al. [49] describes types of successfully performed operations as well as provides an analysis of telepresence surgery. The consequences of a telepresence surgery experience lead to another application of telepresence systems in telemedicine called telerobotics. It is used to perform real-time interactive hands-on experience transfer by an expert surgeon to novice surgeons located at the remote site. This is done by providing expert surgeon control over robotic platform in order to assist the remote mentor and, if necessary, take over and complete the task. The work presented by M. Anvari [7] describes a Canadian experience in telerobotic surgical operations performed between two hospitals separated 400 km away. Successful results of remotely completed 22 robotic assisted advanced laparoscopic surgeries demonstrate not only feasibility of such tasks but also provides an example of an effective solution for remote “on the job” training.

Despite of the number of successful results, current telesurgical technology experiences several major barriers. The first and most important barrier is the quality of communication lines. Because of the strict requirements for control and associated video information transmission, time delays are not acceptable. The second barrier is the cost of robotic platforms and associated maintenance cost.

The equipment used for teleconferencing and telemedicine can be considered as static in terms of placement and operation. Nevertheless, equipment used for hazardous environments requires mobile deployment of both a video-acquisition system as well as actuators for remote robotic manipulation. Moreover, since most of the hazardous tasks involve approaching an operational area, existence of panoramic-vision in addition to stereo-vision is required. All of that makes this class of telepresence systems not only the most



complex and the most demandable, but also applicable to all other classes of telepresence. Therefore, the presented work will focus on the class of telepresence systems with regard to hazardous environments.

With this in mind, the following aspects should be analyzed before determination of the proposed system architecture:

- Human vision system organization and operation;
- Observation of existing implementations of telepresence systems for hazardous environments;
- Stereo and stereo-panoramic machine vision systems;
- Stereoscopic display systems;
- Video compression/decompression mechanisms and standards;
- Real-time video processors.

## 2.2. Human Vision System Organization and Operation

Each human eye acquires individual images of a visual scene from two slightly different perspectives. This difference is created because of an eye separation on about 60-80 mm and is referred to as a baseline. Several stages are involved in order to perceive (receive and process) an image by a human.

In the first stage, light rays entering the eye are bent and brought to focus as they pass through the cornea and lens. The iris, located after the cornea, controls the amount of light entering the eye through the pupil. By means of ciliary muscles, the shape of the lens can be changed in order to perform focus operation. Figure 2.1 below demonstrates the general organization of a human eye.

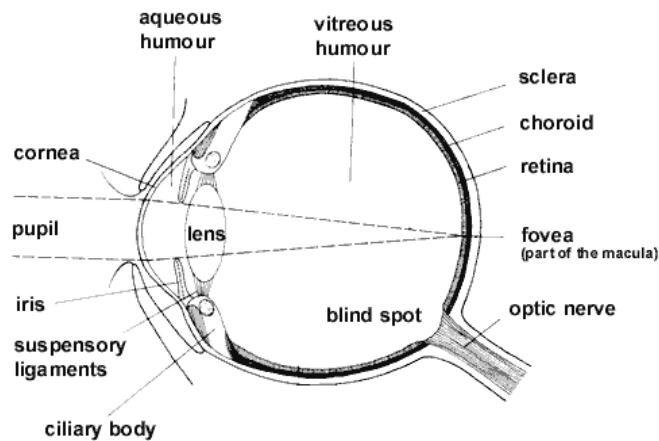


Figure 2.1: Human Eye Organization

In the next stage, a focused image is projected onto photosensitive receiver which is called the retina. The retina consists of two types of regions, central and peripheral. Both of these regions contain arrays of two types of cells, called rods containing approximately 100-120 million cells and cones, containing 6-7 million cells accordingly.

Cones are high-precision cells which manage color vision and detail. The highest concentration of cones can be found in macula's region called fovea. The fovea is the absolute center of vision and is specialized for visual acuity and object recognition. Cone cells dramatically reduce in number beginning from the fovea and up to both temporal and nasal directions. This results in poor visual acuity in peripheral region comparing to foveal vision (central vision).

Rod cells are insensitive to color and are more sensitive to light than cone cells, thus, they manage vision in low light conditions. They are generally dominant in the periphery of

the retina and are used for night vision, motion detection, large objects detection and peripheral vision.

When the light contacts both rods and cones cells, a series of complex chemical reactions occur which in turn create electrical impulses in the optic nerve. These electrical impulses are consequently transmitted to the brain's visual cortex in order to perform image processing.

As it was mentioned before, because of eye horizontal separation, two images are received from slightly different perspectives. The angular difference between two images is called horizontal retinal disparity which is used by the brain to obtain depth information as well as object's spatial location [24]. The retinal visual field of an eye is about  $120^\circ$ . However, the intersection of visual fields from both eyes creates an actual stereoscopic (binocular) region of only about  $50^\circ$ , (see Figure 2.2). In this region, objects that are located within a 30 meter distance can only be perceived in 3D space by a human [9].

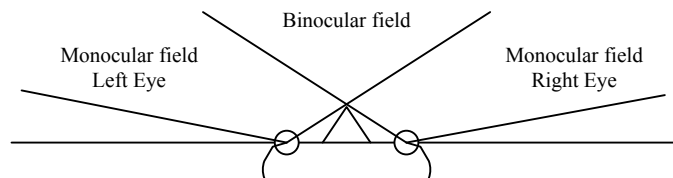


Figure 2.2: Human Visual Fields

An understanding of the above basic principles of human eye operation is very important, since they can be used as a reference for image processing algorithms implementation, as well as for hardware layout organization.

### 2.3. Telepresence Systems in Hazardous Environments

As it was already mentioned, telepresence systems designed to operate in hazardous environments are the most complex and at the same time, the most demandable class of telepresence systems. The spectrum of related tasks include fire fighting, rescue operations, demining operations, underwater operations, outer space operations, bomb disposal etc. In order to prevent direct exposure of a human to such hazardous environments, a variety of systems are being developed [50]. The majority of them are based on a principle of teleoperator placement at a safe remote site equipped with an interactive environment that provides visual feedback and robotic control needed for remote system management. As a result, humans can effectively perform operation without any risk to their lives. An implementation example of such system is presented by Kron et al. [10]. They designed a bimanual haptic telepresence system in order to perform demining operations. Two-arm manipulator with touch and force feedback and the stereoscopic visual system needed for such an application-specific procedure as unscrewing and excluding the detonator were integrated to a quadruped walking robot. Performed experiments demonstrated that operator located 20 km apart from actual teleoperation site successfully completed demining task.

Underwater operations, particularly deep-water operations, are also related to the category of hazardous missions for a human. Periodical inspection and repair procedures of underwater structures such as pipelines, for example, have to be performed to make sure that structural safety and integrity is maintained. Thus, replacement of human divers with remotely controlled Underwater Robotic Vehicles (URV) is highly desirable. An example of such a system is presented by Iastrebov et al. [28]. The Telepresence Viewing System (TVS)

was designed by this research group and was integrated to URV. This enables remote operation from the base vessel providing a stereoscopic view of the scene.

In spite of a number of telepresence systems specifically designed to operate in hazardous environments, there are still a number of challenges related to most of them. For example, limited FOV (Field of View) of a robotic vision system, ineffectiveness of a remote robotic driving performance caused by lack of peripheral vision, poor video image quality and associated frame rate, and communication channel delays are all challenges. The analysis performed by Chen et al. [11] confirms that most of the factors stated above result in human performance degradation which in turn makes task completion questionable.

## 2.4. Machine Vision Systems

For several decades a number of machine vision systems have been developed. Depending on application, video-acquisition of a scene can be stereo, panoramic or a combination of stereo-panoramic. These acquisition methods can be classified into two categories: dioptric methods, where only refractive elements (e.g. lenses) are employed, and catadioptric methods, where reflective components (e.g. mirrors) are used in combination with refractive elements. Dioptric systems include single or multiple cameras attached to various lenses. Catadioptric systems include mirrors of different shapes (e.g. curved, planar) and a camera with attached optics.

The majority of stereo video-acquisition systems are based on a principle of human's binocular vision which was described in Section 2.2. However, the function of an eye's optic system and associated retina are fulfilled by optic lens (e.g. C-mount or CS-mount) mounted

on top of an image sensor (e.g. CMOS or CCD). Providing a separation between acquisition units, image processing tasks (e.g. stereo matching, disparity calculation, etc.) performed by the brain are realized by stream processors which are described in the Section 2.7.

An example of stereo vision system based on the previously mentioned principles and designed for remotely operated robot is provided in [12]. The proposed stereo vision head consists of two image sensors separately mounted on a frame and horizontally separated from each other, illustrated in Figure 2.3. Each image sensor provides resolution of  $640 \times 480$  pixels at maximum 30 fps. In order to compensate distortions caused by servo motors used to follow operator's HMD movements, stabilization sensors are being exploited. Image processing is performed by means of two PC clusters.

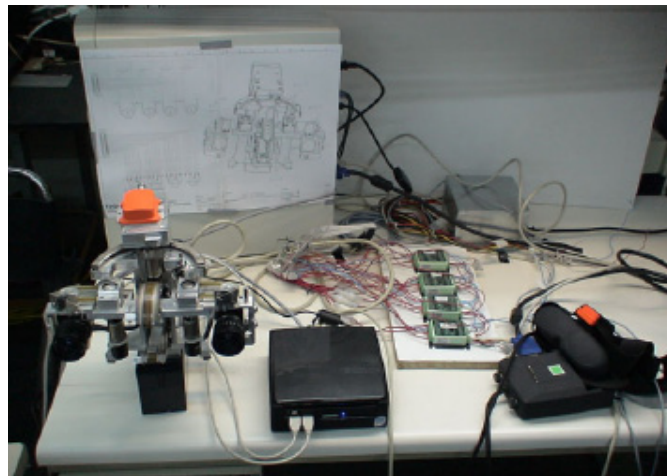


Figure 2.3: Stereo Vision Head System (courtesy to [12])

Other examples of stereo vision systems based on human vision are presented by Khaleghi et al. [13], by Jan Pohanka et al. [14] and by S. Jin et al. [15], Figure 2.4 (a), (b),

(c) accordingly. Each system, being application specific, introduces different baseline parameters. In order to simplify the camera calibration process and reduce computations associated with image rectification and disparity calculation, image sensors are placed on the same PCB. Image processing units selected for the first two systems are BlackFin DSPs. Third system has Virtex 4 FPGA, programmed to perform necessary computations.

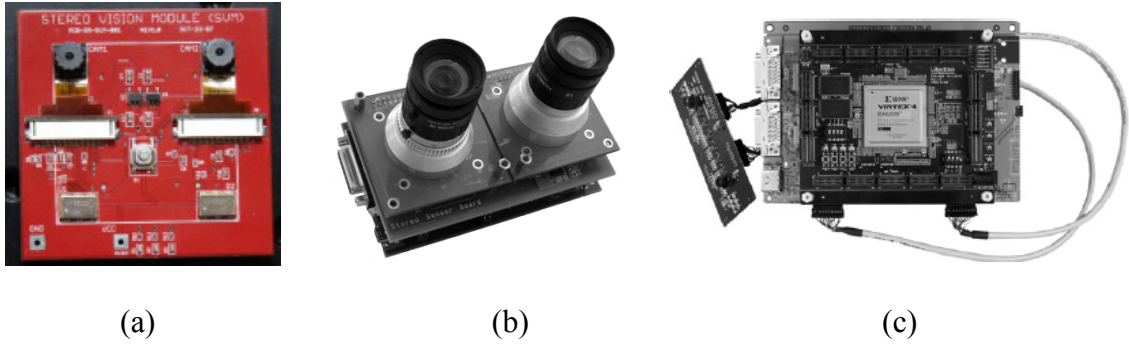


Figure 2.4: Stereo Vision Systems: (a) courtesy to [13], (b) courtesy to [14], (c) courtesy to [13]

A completely different technique of designing stereo-vision systems is based on a single-camera and a set of mirrors located in front of it. Recent work performed by Theodore P. Pachidis and John N. Lygouras [16] presents Pseudo-Stereo Vision System (PSVS) composed of a camera, a beam splitter and three statically positioned mirrors. As it can be noted from Figure 2.5, views acquired by mirrors 2 and 4 can be considered as if they were captured by two virtual cameras. Mirror 1 is a beam splitter which permits the reflection of 50% of the incident light from the 2<sup>nd</sup> mirror, whereas it permits through its body the other 50% of light coming from the 3<sup>rd</sup> mirror. As a result, the superposition of two views creates a stereoscopic image acquired by a single real camera.

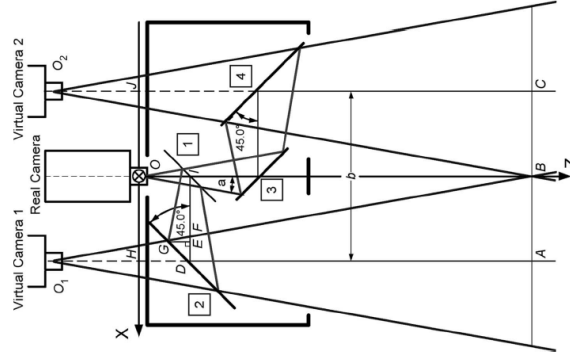


Figure 2.5: Pseudo-Stereo Vision System (courtesy to [16])

Panoramic and stereo-panoramic vision systems are also presented by a number of research groups. R. Petty et al. [17] presented a setup consisting of a pair of line-scan sensors mounted on a rotating platform. Each CCD line-scan sensor consists of a column of 1024-elements. During rotation, each one-dimensional sensor obtains two-dimensional perspective images of an object workspace (see Figure 2.6). This can be done with FOV of an image varied up to  $360^\circ$ . After the rotation of the sensors with respect to a static scene of an object of interest is completed, two overlapping fields of view define the stereoscopic region. For any object located within the stereoscopic region, disparity calculations may be performed.



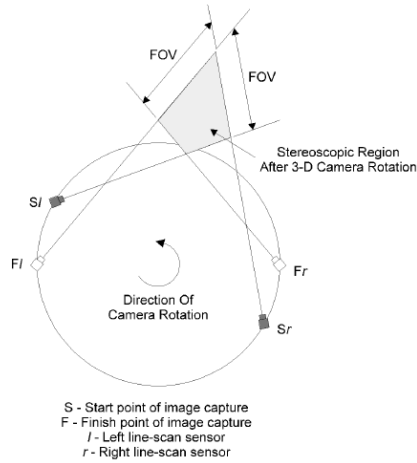


Figure 2.6: Line-Scan Sensor Rotation System (courtesy to [17])

Similar to the previous system, an approach utilizing the panning technique (when a line scan-camera rotates up to a full view of  $360^\circ$  around a rotation axes) is presented by M. Barth and C. Barrows [18]. However, instead of a two line-scan sensors, one is used to provide a complete two-dimensional panoramic acquisition (see Figure 2.7).

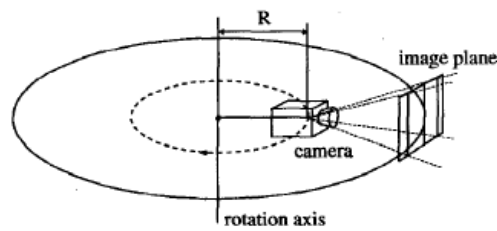


Figure 2.7: 2D Panoramic System Based on Panning Technique (courtesy to [18])

S. Tzavidas and A. Katsaggelos [23] proposed alternative implementation of a rotating camera method without moving parts. Instead of rotating a single or a pair of cameras,

multiple cameras are mounted on a rig with fixed geometry, as shown in Figure 2.8. As it can be seen, all cameras look outward from the center of the rig. A circular projection technique is employed in order to construct a stereo-panoramic video sequence.

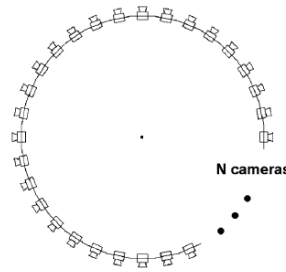


Figure 2.8: Multiple Camera Panoramic System (courtesy to [23])

Another group of stationary panoramic vision systems use specially shaped mirror(s) combined with conventional camera(s). As a mirror reflects an image of the environment to a camera, it can be then converted to a panoramic perspective image. Depending on the shape of the mirror, which can be conical, hyperboloidal [19], spiral [20] etc., different transformation methods are required.

J.S. Chahl and M.V. Srinivasan [21] developed a technique for capturing panoramic images using a curved mirror and a conventional CCD camera providing extra-wide FOV, Figure 2.9. The shape of the mirror is such that a linear relationship exists between the angle of incidence of light onto the mirror surface and the angle of reflection onto the image sensor; in respect to the center of array.

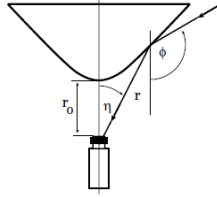


Figure 2.9: CCD Camera and Curved Mirror Panoramic System (courtesy to [21])

Setup presented by T.Nakao and A. Kshitani [22] is based on a stationary camera and 2-axis mirror rotation mechanism as illustrated in Figure 2.10. Performing a 360° mirror rotation using set of motors, allows panoramic scene acquisition with a single stationary camera.

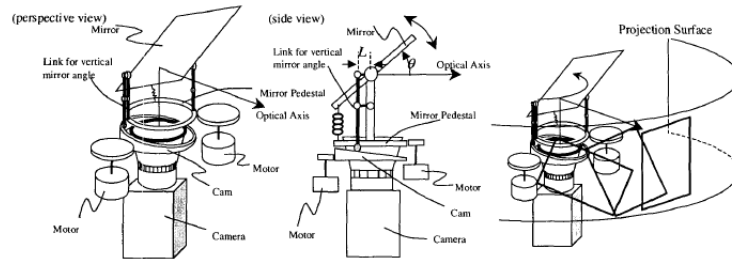


Figure 2.10: Rotating Mirror Architecture for Panoramic Vision (courtesy to [22])

## 2.5. Stereoscopic Display Systems

Over the years a number of 3D display techniques have emerged. Existing techniques can be divided into display systems requiring the user to wear eyeglasses for filtering or

displaying purposes and into systems where the user can observe a stereoscopic scene in a natural manner.

A very first stereoscopic display system was introduced in 1853 and it is still being used now. An example of such a system is color-multiplexed (anaglyph) display. It consists of anaglyph glasses and a pair of projector devices. The glasses have two different lens-filters which are chromatically opposite. For example, one of the commonly used color schemes can use a red filter to pass through only the red component of an image for the left eye and the second cyan filter to pass through only Blue and Green components of an image for the right eye. Both projectors have the same color filters as used for the glasses. The image pair is displayed overlapping, such that the left image contains only the red component and the right image contains only the cyan component. Portions of the image that are red will appear dark through the cyan filter, while portions consisting only of green and blue will appear dark through the red filter. Therefore, each eye sees perspective as it is supposed to see which results in stereoscopic sensation.

Although anaglyph offers an affordable solution to create a good depth sensation, the quality of perceived colors is very poor. Moreover, despite the quick adaptation to stereoscopic view, after 15 minutes unpleasant aftereffect causes reduction of red-cyan perception of the real world.

Another example of stereoscopic display system requiring eyeglasses is the Polarized Display System. It is based on the same principle as anaglyph, however, instead of colored filters, polarized ones are used. The polarized filter is used to perform a conversion of a number electromagnetic waves emitted by a light source and represents an undefined

polarization into a beam with defined polarization. The common types of polarization filters are linear and circular polarization filters.

When an un-polarized light enters the linear polarizing filter, only electromagnetic waves oscillating in a singular orientation plane defined by a polarization degree (vertical or horizontal transmitting axis) can fully pass through, (see Figure 2.11 [a]). However, waves oscillating in a transmitting axis of  $90^\circ$  to one another are absorbed or reflected by the filter, thus producing near absolute blockage of light (see Figure 2.11 [b]). Finally, waves having both horizontal and vertical components will be partially transmitted.

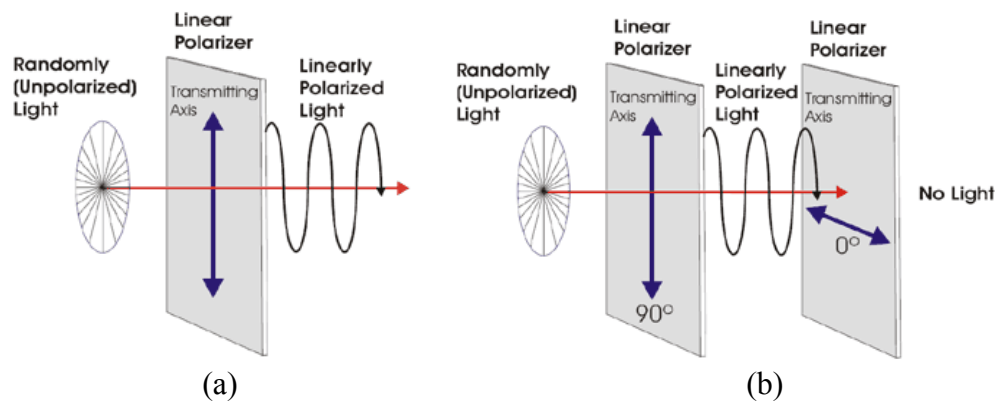


Figure 2.11: Linear Polarization Scheme (courtesy to API [71])

The circular polarizing filter represents the combination of a linear polarizer and a quarter wave retarder [71]. After the linear polarization is performed, light passes through the quarter wave retarder which retards the velocity of one of the polarization components (X or Y), one quarter of a wave out of phase from the other polarization component (see

Figure 2.12). Depending on which polarization component is retarded, one will have either a left handed or right handed circular polarizer. The left handed circular polarized light cannot pass through a right handed circular polarizer and vice versa. This occurs because of the reflected opposite handed circularly polarized light passing through the wave retarder, which becomes linearly polarized again, but  $90^\circ$  to its original orientation. Therefore light passing through a circular polarizer filter, reflecting off another surface, cannot pass back through the circular polarizer.

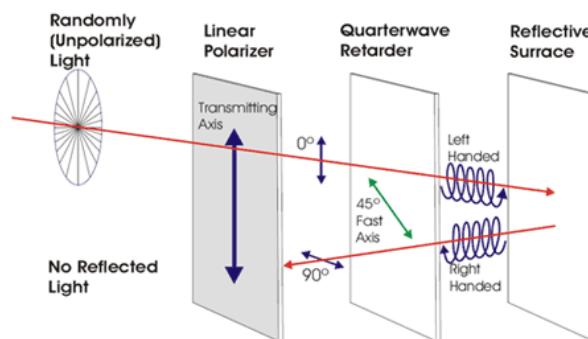


Figure 2.12: Circular Polarization Scheme (courtesy to API [71])

Circular polarization prevents the black-out effect and has a reduced “ghosting” effect compared to linear polarization. Despite better color rendering and lack of adaptation period comparing to anaglyph display, polarized stereo systems need a much stronger light source since polarized light loses at least 50% of its original intensity.

Shutter glasses are also related to the stereoscopic display class. In order to enable stereo viewing, shutter glasses are used in conjunction with a display screen. The shutter is made of

glass containing a Liquid Crystal (LC) layer. When the voltage is applied, the glass becomes dark, otherwise the glass is transparent. Thus, shutters are alternately opaque and transparent so that only one eye can see through the glasses at any given moment. The display system alternately presenting the right and left eye perspective images has to generate frames twice the frequency required for viewing. Thus, glasses being synchronized with the display let through only one image at a time for each eye enabling illusion of a three dimensional space.

Shutter glasses mostly eliminate “ghosting” which is a problem for two previously described setups. Moreover, unlike Red-Cyan color filter 3D glasses, LC shutter glasses provide 3D viewing in the full color spectrum. However, the constant shuttering might bother viewers who are very sensitive to low refresh rates (e.g. 60Hz) and cause flickering which can result in headaches. In addition, the majority of eyewear is designed so that peripheral vision, which is not shuttered, is completely covered to avoid viewing differences, thus creating the lack of peripheral perception.

Head Mounted Display (HMD) represents another existing solution of stereoscopic display class. HMDs can have a form of glasses or helmets worn on the head and consist of two small displays positioned near each users’ eyes blocking out all external light or vision. They are commonly used in virtual reality applications and normally include a mechanism that follows the movement of the observer’s head. Recent HMDs are realized by LCD or OLED technology. Being connected directly to a video source, displays present left and right images to the appropriate eyes providing a strong binocular disparity and 3D illusion. However, lower-priced HMD systems provide very narrow FOV while wide FOV which is

natural for humans, is only observed in HMDs priced over \$10,000 [70]. Examples of HMD devices used in telerobotics and telesurgical systems can be found in [26],[12] and [8].

Autostereoscopic Displays (ASDs) represent stereo displaying techniques which eliminates use of any type of eyewear or headgear needed for filtering or displaying visual information [27].

State-of-the-art systems are based on Flat Panel Displays (FPDs) equipped either with lenticular screens or parallax barriers to provide binocular disparity. In both cases, a pair of images for the left and right eye is cut in vertical strips. Then, vertical strips associated with the left and right viewpoints are interlaced and combined into one image. This interlaced image is then displayed on FPD.

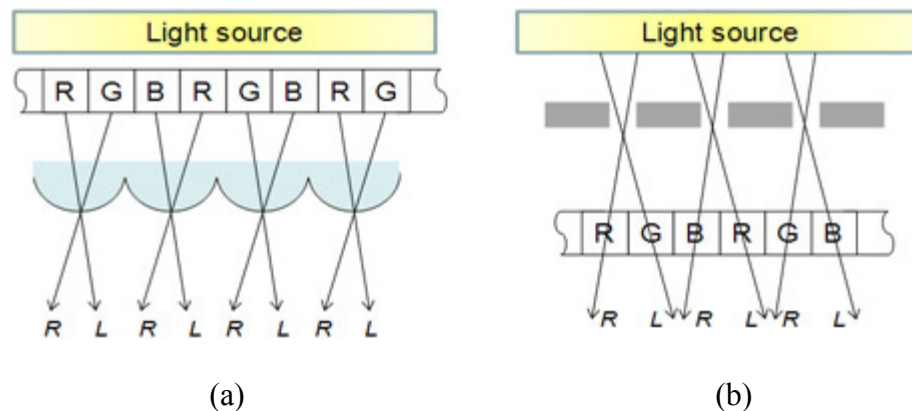


Figure 2.13: Autostereoscopic Displaying Schemes

Lenticular lens represents an array of magnifying lenses such that when viewed from slightly different angles, different parts of images are magnified. Attaching a lenticular lens



to the front of FPD screen magnifies parts of interlaced image that correspond to the left viewpoint and can only be seen by the left eye and vice versa, (see Figure 2.13 [a]). Such an acquisition process from two viewpoints creates 3D perception.

Parallax barrier is a fine column based grid of liquid crystals placed in between of TFT-LCD and associated backlight, (see Figure 2.13 [b]). By controlling the switching of the liquid crystal light emitting from the backlight, the interlaced image separates into two images. This causes different images to be seen by the right and left eyes, creating a sense of depth.

Both lenticular lens and parallax barrier technologies are heavily dependent on viewer location. Small deviation of view to either side causes a distortion of images. Thus, a very specific “sweet spot” for getting the 3D effect should be maintained. Moreover, because FPD resolution is divided between two images the overall horizontal screen resolution is reduced by a factor of two.

## 2.6. Video Compression/Decompression Mechanisms and Standards

As it was already mentioned in the beginning of this chapter, video information plays an essential role for applications such as telepresence. Besides that, the requirements for video quality and associated frame rate are very high. As a result, bandwidth of a communication channel required to transmit such HD data has also been in accordance to data rate. However this in most of the cases is not feasible. Therefore, compression mechanisms play an essential role for bit rate reduction of the digital video signal. The compression ratio can be described as a degree to which the encoder reduces the bit rate and equals:

$$\text{compression ratio (CR)} = \text{decoded bit rate} / \text{encoded bit rate}$$

Compression mode can be lossless or lossy. The lossless mode guarantees the exact recovery of the original data after compression and subsequent decompression. The lossy mode assumes that decompressed data would not be identical to the original.

Over the years a series of compression standards for still images such as JPEG, JPEG2000 as well for video sequences such as M-JPEG, M-JPEG2000, MPEG group including MPEG-1, MPEG-2, MPEG-4 and H.264 has been established. Prior to understanding MPEG standards, the JPEG standard is described since it is used as a building block for the complete MPEG group.

Joint Photographic Experts Group (JPEG), named after the committee that created it, is a worldwide ISO standard for compression of digital still images. The JPEG defines both controlled lossy and lossless compression based on Discrete Cosine Transform (DCT) compression algorithm. The basic JPEG encoder flow consists of four stages. At the first stage, the input image is divided into blocks of  $8 \times 8$  pixels. The choice of the block size is specifically selected to reduce computation requirements related to larger blocks. On the next stage, a two-dimensional discrete cosine transformation (DCT) is performed for all  $8 \times 8$  pixel block values. Following the transformation, blocks' coefficients are quantized. The quantization step size for each of the 64 DCT coefficients is specified in a quantization table, which remains the same for all blocks. In the next stage, in order to further reduce the amount of data represented in each block redundancy reduction is performed. As a result, the bit stream generated in this way, combined with signaling information such as quantization tables, resolution details, etc. shape a JPEG bit stream.

The corresponding decompression is performed by following all inverse procedures (i.e. redundancy reduction, quantization, DCT) and then putting 8x8 pixel blocks back together results in an output image construction.

When used in mid and high bit-rate applications, JPEG compressed pictures show no visual difference to the original uncompressed ones. On the other hand, applications with low bit-rate requirements involve greater quantization steps which results in artifacts appeared as “blockiness” due to 8×8 pixel block structure.

As it was already mentioned the JPEG standard was designed for a compression of only still images. On the other hand, in order to compress a video sequence, Motion-JPEG (M-JPEG) was offered. The concept of M-JPEG assumes that each video frame of an image sequence is coded as a separate JPEG image. Despite advantages of JPEG such as flexible CR and provided robustness with no dependency between the frames, M-JPEG was not adopted for several reasons. The first reason is the lack of a standard for M-JPEG meaning that there are considerable incompatibilities between the methods used by different manufacturers [29]. The second reason is that independent frame compression, without taking into consideration similarity between successive frames, makes no use of video compression techniques which results in a very low compression ratio.

In order to overcome shortcomings of M-JPEG, Moving Pictures Experts Group (MPEG) defined an algorithm for coding moving pictures based on work of JPEG. The main conceptual difference introduced in the group of MPEG standards is a consideration of similarities in consecutive video pictures which results in higher compression factors.

Therefore, the following paragraph will explain the general principle of video compression common for MPEG group and after, standard specific features will be described.

Each MPEG video sequence is divided into one or more groups of pictures (GOPs). Typical pattern of GOPs is represented in Figure 2.14. GOPs are defined by three types of pictures: I-, P- and B-pictures. In a sequence of picture frames, the start frame, called Intracoded picture (I-picture), is spatially encoded with no reference to other pictures using the JPEG engine. I-picture frame is used as a main reference for further frames and can be decoded without referencing to other pictures. This allows termination of transmission error propagation to the next frames. Since I-pictures are compressed using the JPEG standard, it provides only moderate compression.

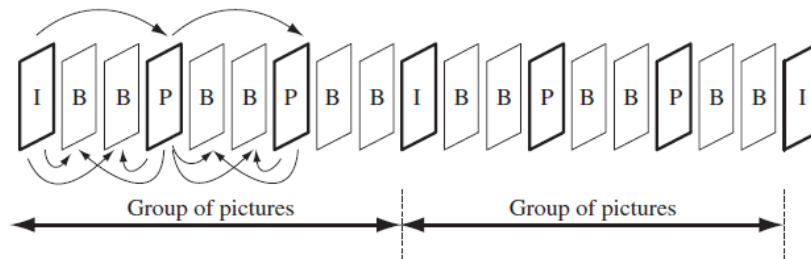


Figure 2.14: MPEG GOP Sequence

The second type of pictures is Predictive-coded pictures (P-pictures). Pictures of this type are predicted from the preceding I- or P-pictures using the forward motion-compensated prediction mechanism. For that, a picture is broken down into macroblocks, which are the basic elements for motion estimation, and then the block-matching technique

is applied to the preceding picture. For example, a block-matching can be performed by shifting macroblock with a specific step (e.g. pixel, half of the pixel etc.) to all possible positions within a given search area. When a match is found, motion vector is generated, specifying the location of the best matching macroblock in the reference frame to be used for prediction picture. As a result, a set of motion vectors are generated and transmitted to the decoder in order to reconstruct the picture from the original reference. This allows achieve higher compression due to vector transmission instead of complete image parts as it is for I-pictures. However, because P-pictures serve as a reference for future P- and B-pictures prediction, errors occurring during transmission can propagate to the succeeding pictures and multiply over the time.

The bidirectional-predicted pictures (B-pictures) are coded using bidirectional motion-compensated prediction from both the past and future references of I- or P-pictures. The procedures involving macroblock-matching technique are the same as those described for P-pictures, however, the search is performed in past and future references as depicted by arrows in Figure 2.14. As a result, forward motion vector, which references to a best matching block in the previous picture, and a backward motion vector, which references to a best matching block in the following picture, is generated. By averaging between the past and the future blocks, motion-compensated prediction is formed. Therefore, B-pictures provide the highest compression in contrast to I- and P-pictures and do not propagate errors since they are not used as references.

In spite of better compression ratios of B-pictures, increasing the number of B-pictures reduces correlation between the previous and next I- or P-pictures. This results in quality

degradation as well in increase of computational complexity. Moreover, the search algorithm needed for prediction requires very large computational power. Because of the frame dependencies, errors in I- or P-pictures result in error multiplication for future predicted frames.

MPEG-1 is the first generation of digital compression standards for video and two-channel stereo audio applications. It was proposed based on the assumptions that acceptable video and audio quality can be obtained for a total bandwidth of about 1.5Mbps. The input resolution supported by MPEG-1 is optimized to operate at  $352 \times 240$  pixels at 30 fps, commonly referred as SIF (Source Input Format) video. However, the maximum supported resolution can be as high as  $4k \times 4k$  pixels at 60fps. MPEG-1 is also defined for progressive frames only; therefore, it has no direct support for interlaced video applications commonly used in broadcasting television applications.

The second phase of MPEG, named MPEG-2, appeared with demand for higher quality and higher resolution video applications operating at 4-15 Mbps. The MPEG-2 standard is heavily based on MPEG-1; however, numbers of improvements are introduced.

In order to cover a wide range of applications, MPEG-2 classified groups of profiles and levels (bitstream scalability, color space resolutions, frame resolutions, bit rates, sampling rates) are important for applications. Thus, seven distinct profiles are defined as simple, main, signal-to-noise ratio (SNR) scalable, spatially scalable, high, 4:2:2, and multiview. Within each profile four levels are defined as low ( $352 \times 288$  pxls and 30 fps), main ( $720 \times 576$  pxls and 30 fps), high-1440 ( $1440 \times 1152$  pxls and 60 fps), high (1920  $\times$  1152 pxls

and 60 fps). Thus, all possible combinations of profiles and levels cover the most practical MPEG-2 applications. Moreover, MPEG-2 is suitable for coding not only the progressive video format but as well the interlaced video format commonly used in broadcasting television applications. Therefore, frame-based and field-based motion-compensated prediction and two corresponding DCT modes are developed.

MPEG-4 Part 2 and later developed MPEG-4 Part 10 (H.264/AVC) belong to the MPEG-4 standard.

MPEG-4 Part 2 [32] is based on coding techniques similar that of MPEG-1/MPEG-2, however it defines four types of coding tools: video object coding for the coding of natural and/or synthetic generated, rectangular or arbitrarily shaped Visual Objects (VOs), mesh object coding for the coding of visual objects represented with mesh structures, model-based coding for the coding of a synthetic representation and animation of the human face and body, and still texture coding for the wavelet coding of still textures. Similarly to MPEG-2, the MPEG-4 standard defines profiles; each of which targets a specific group of applications. The most commonly implemented profiles are Simple Profile (SP) and Advanced Simple Profile (ASP). SP targets low-complexity and low-delay applications such as mobile video communications, whereas the ASP provides higher coding performance and can be used in applications where video at higher bit rates is required.

Although H.264/AVC [33] is similar to preceding standards in the block-based video compression technique, it provides significantly higher compression capability due to several conceptual revisions and improvements.

Similarly to previous standards, pictures are partitioned into macroblocks, which are fundamental coding units. Also, each picture can be divided into a number of independently decodable slices, where each slice consists of one or more macroblocks. The slice type (e.g. I-slice, P-slice, B-slice) defines which prediction modes are available for the macroblocks. Within a single frame different types of slices can be presented. The concept of slice group, known as Flexible Macroblock Ordering (FMO) feature, provides loss/error robustness, enhanced Region of Interest (ROI) customization and low-delay construction of composited image scenes. Moreover, H.264/AVC is distinguished from its predecessors in a number of additional features. These features include spatial directional prediction; an advanced motion compensation model using variable block size prediction, quarter-sample accurate motion compensation, multiple reference picture prediction, and weighted prediction; an in-loop deblocking filter. Evaluation performed using H.264/AVC [30] standard show that coding yields bit rate savings at approximately half a bit rate used by MPEG-2 and approximately 35% over MPEG-4 Part 2 SP.

JPEG2000 standard [31] was created to address weaknesses of JPEG standard as well to provide a number of new features. The JPEG2000 standard is characterized by a maximum allowed image size to be  $(2^{32}-1) \times (2^{32}-1)$  and the maximum number of components in an image to be  $2^{14}$ , color component depth ranging from 1 to 38 bits, superior low bit-rate performance, continuous tone (grayscale and color) and bi-level (B/W) compression, lossy and lossless compression, robustness in the presence of bit errors, backward compatibility with JPEG and many more [34].



The typical JPEG2000 encoder system consists of several fundamental building blocks, as shown in Figure 2.15. These blocks include image pre-processing, Discrete Wavelet Transform (DWT), quantization and Embedded Block Coding with Optimized Truncation (EBCOT).

Image preprocessing is an optional step and consists of tiling (partitioning of image into a number of rectangular non-overlapping blocks), DC level shifting and multi-component transformation (reduction of the correlations among the multiple color components in a multicomponent image). After optional preprocessing step, DWT performs decomposition of each image tile into four sub-bands. On the next stage each sub-band is independently quantized by a quantization parameter. The quantized sub-bands are then divided into a number of smaller code-blocks of equal size.

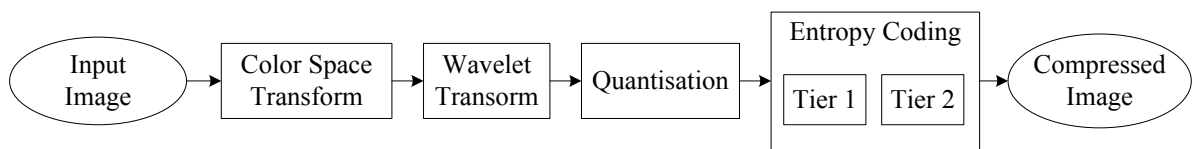


Figure 2.15: JPEG2000 Encoder Flow

Following this, each code-block is passed to EBCOT engine consisting of Tier1 and Tier2. Tier1 performs entropy coding, while Tier2 generates the output bitstream.

The corresponding decoder architecture is similar to the encoder architectures with data flow in the reverse direction.

The results obtained by [35], [36], [37] indicate that JPEG2000 compression scheme outperforms the JPEG scheme in all aspects, thus, providing higher compression ratios and quality on both higher and lower bit-rate scenarios. The main drawback of the JPEG2000 standard compared to its predecessor is that the coding algorithm is much more complex and computationally intensive than that implemented in JPEG.

Similarly to the MJPEG system, the JPEG2000 standard defines file format as well instructions for coding motion sequences. The motion JPEG2000 represents an image sequence encoded using JPEG2000 coding algorithm for still pictures. Therefore, MJPEG2000 inherits all advantages of JPEG2000 and is aimed for applications where high-quality frame-based compression is needed.

## 2.7. Real-Time Video Processors

The class of applications requiring task execution under specific timing constraints is related to real-time system class. In specialized telepresence systems, real-time video stream processing involves manipulation with a great amount of continuously generated data. Therefore, in order to handle it four types of processing devices such as Central Processing Unit (CPU), Graphics Processing Unit (GPU), Application Specific Integrated Circuit (ASIC) and Reconfigurable Computing System (RCS) have been used.

CPU is a general-purpose processor related to a Single Instruction Single Data stream (SISD) uniprocessor category. The processor consists of three major components such as register set, Arithmetic Logic Unit (ALU) and Control Unit (CU). These components allow

for the execution of tasks consisting of the set of instructions stored in memory. In general, all instructions are executed in a sequential order. There are five stages per each instruction execution cycle such as Instruction Fetch (IF), Instruction Decode (ID), Data Fetch (DF), Instruction Execution (EX) and Store result (S). However, four out of five execution stages perform only service operations (IF, ID, DF, S). Therefore, in order to improve processor performance two basic techniques are used: clock frequency boosting and/or Instruction-Level Parallelism (ILP) [38].

There is another type of CPU which instead of one processor has multiples of them (e.g. two, four, and eight) interconnected and located on the same die. A Multiprocessor, also called multi-core Processor, has Multiple Instruction streams Multiple Data streams (MIMD) [38] architecture. In MIMD, each processor has its own control unit that issues instructions and operates on its own data. This allows operation on multiple threads in parallel which is called Thread-Level Parallelism (TLP) [38]. Due to such architecture it is possible to reduce not only overall task execution time but also to reduce operation frequency of the multiprocessor.

In contrast to general-purpose CPU, Graphical Processing Unit (GPU) architecture is specifically tailored for image processing algorithms which are computationally intensive and parallelizable. The fact that image processing tasks require similar calculations performed on quantities of data led to a Single Instruction stream Multiple Data stream (SIMD) [38] architecture. A modern GPU consists of a set of multiprocessors each of which consist of a set of processors with SIMD architecture. This allows execution of the same instruction for all multiprocessors but operation on different data at each clock cycle, thus exploiting the Data Level Parallelism (DLP) [38].

Despite GPU's dramatic performance increase in graphical applications, not all algorithms can be vectorized. Moreover, even making use of DLP, each individual processor still performs sequential instruction execution thus requiring higher clock rates for performance improvement. As a result, the increase of power consumption is unavoidable.

As opposed to general-purpose CPU and GPU processors, ASIC's internal architecture is specifically customized for a particular application and not for a general use. An ASIC engineer can create functional blocks with known electrical characteristics (e.g. propagation delay, capacitance and inductance, etc.), employ data parallelization, implement cache memory according to specification etc. Moreover, modern ASICs can include processors, memory blocks and other large building-blocks thus creating so-called System-on-a-Chip (SoC) architecture. All of this allows for the achievement of maximum performance possible for a particular application. Moreover, due to customizable architecture power consumption can also be decreased. However, for all those advantages there is a price to pay expressed in design complexity, associated cost, high cost per a unit when producing a small number of units and the longest time to market out of all possible implementation methods.

Field-Programmable Gate Array (FPGA) is a class of Reconfigurable Computing Systems (RCSs) devices that has performance close to an ASIC, but has a flexible architecture. By using Hardware Description Language (HDL) any design (e.g. DVB modulator, MPEG-4/H.264 decoder, DDR2 Controller, etc.) can be described on a Register Transfer Level (RTL), consequently synthesized and loaded to the FPGA's configuration memory. Unlike sequential processors, FPGAs can provide any level of parallelism thus

maximizing throughput and minimizing power consumption by lowering operation frequency.

The basic architecture of FPGA consists of an array of configurable logic embedded into a configurable interconnected structure and surrounded by configurable I/O blocks [39]. In addition, modern FPGAs [40],[41],[42],[43] include other specialized elements such as memory blocks, DSPs, microprocessors etc. These specialized blocks perform more specific tasks than configurable logic, however they can also be configured in accordance with application needs. In case any modifications to the existing circuitry (e.g. communication standard) running on FPGA are necessary, the HDL code can be modified and reprogrammed again. This procedure can be performed as many times as necessary. All of this makes FPGA a versatile and useful device for high performance applications (e.g. video stream processing). Moreover, fast prototyping and as a result, shorter time-to-market, can be achieved due to reprogrammable architecture. However, high production cost of FPGAs makes them useful for only low volume products.

The performance comparison of CPU, GPU and FPGA for image processing applications is presented by Shuichi Assano et al. [44]. The experimental results are suggesting that GPU has a potential for achieving almost the same performance as FPGA, but only for tasks involving independent pixel processing. As for more complicated algorithms which use shared data, GPU demonstrates very poor performance. In contrast, multi-core CPU performs much better for shared data processing.

Another comparison of CPU and GPU processors performed by Nan Zhang et.al. [45] shows significant speedup achieved by GPU used for edge detection and filtering algorithms tested on various image resolutions.

The comparison of performance obtained with FPGA and GPU implementations of real-time stereo vision system is performed by Rathees Kalarot and John Morris [46]. The results reveal that despite a much slower system clock, FPGA implementation is superior due to extensive pipelining.

## 2.8. Conclusion

The literature observation of existing telepresence systems oriented for various applications has shown that the class of the systems designed for hazardous environments is the most complicated for implementation. This is due to the following reasons: i) telepresence systems used for hazardous environments, in a general case, requires mobile deployment of both a video-acquisition system as well as actuators for remote robotic manipulation; ii) necessity of panoramic vision in addition to stereo-vision has to be provided for natural perception of the remote environment by a human-operator ; iii) various modes of operation have to be supported in order to provide adaptation of the system to different operation tasks. The above reasons makes this class of telepresence systems the most complex and applicable to all other classes of telepresence systems (e.g. telemedicine, teleconferencing etc). Therefore, the presented work focuses on the class of telepresence systems in regards to hazardous environments.

To determine the potential architectural organization of telepresence system for the selected class of applications, the following aspects were analyzed: i) existing implementations of telepresence systems for hazardous environments; ii) organization of the human visual system; iii) existing machine vision systems; iv) stereoscopic display systems; v) real-time video processor solutions and vi) mechanisms of video compression and decompression.

Based on the performed observation and analysis, the following issues associated with the existing machine vision systems were determined: i) lack of peripheral vision; ii) utilization of electromechanical (moving) parts; iii) relatively low video quality performance including frame resolution and/or frame rate; iv) relatively high power consumption in case of CPU based implementations. The Table 2.1 summarizes the implementations details of existing machine vision systems.

Table 2.1: Machine Vision Systems Summary

<b>Literature source</b>	<b>Stereo</b>	<b>Panoramic</b>	<b>Processor type</b>	<b>Video-acquisition method</b>	<b>Video quality</b>
Pohanka et al. [14]	yes	no	FPGA & DSP	Dioptric	n/a <sup>*</sup>
Jin S. et al. [15]	yes	no	FPGA	Dioptric	60fps
Amanatiadis A. et al. [12]	yes	no	CPU	Dioptric	640x480, 30fps
Khaleghi et al. [13]	yes	no	MCU & DSP	Dioptric	20fps
Iastrebov et al. [28]	yes	no	n/a <sup>*</sup>	Dioptric	800x600
Nakao T. et al. [22]	no	yes	CPU	Catadioptric	320x240
Chahl J.S. et al. [21]	no	yes	CPU	Catadioptric	n/a <sup>*</sup>
Petty R. et al. [17]	yes	no	CPU	Dioptric	1024x1
Barth M et al. [18]	no	yes	CPU	Dioptric	1024x1

<sup>\*</sup> Not available information

In the next Chapter the novel architecture of reconfigurable platform for 3D-Panoramic Telepresence System for Mobile Applications that mitigates the above mentioned problems is proposed.



### 3. ARCHITECTURE DEVELOPMENT OF ADAPTIVE 3D-PANORAMIC TELEPRESENCE SYSTEM FOR MOBILE APPLICATIONS

#### 3.1. Introduction

Limitations introduced by existing telepresence systems and highlighted in the previous chapter led to a series of specifications. These specifications have to be defined prior to developing the reconfigurable platform architecture of the 3D-Panoramic Telepresence System.

The general organization of a system conforming to the concept of telepresence and oriented for mobile applications in hazardous environments assumes an existence of two sites distantly separated from each other. The teleoperator's control site equipped with an interactive environment providing realistic visual feedback, communication and control needed for the remote system management, is referred to as a master side. The remotely controlled site featuring data acquisition, communication and actuator resources is referred to as a slave side.

The telepresence system as it is described above, has to have a set of operation modes which may switch during different stages of task execution. For example, tasks such as space operations (e.g. satellite docking, object grasping, etc.), demining operations, fire fighting, etc. require several operation modes associated with an object of interest of localization, approaching an object, and interaction with an object in order to realize the operational target.

Following subsections of the chapter contain the detailed analysis of various modes of operation that may be required for both the master and slave sides of the telepresence

system. Based on the operation modes analysis, the general system architectural organization is then determined. Each subsystem of architecture is described in detail and associated hardware, firmware and software partitioning is performed.

### 3.2. Analysis of Operation Modes

In order to define possible modes of telepresence system operation the principles of human vision described in Section 2.2 have to be taken into consideration. In particular, binocular (i.e. stereo) and monocular (i.e. peripheral) visions have to be analyzed since they play an essential role in mode stack determination.

Both types of vision acquire information regardless of the task being performed. However, the way acquired information is interpreted by the brain has in fact direct relation to the task at hand. Following examples will illustrate how human vision system performs in various conditions.

Consider example when a human is on a stage of object of interest localization. The distance from the viewpoint to the closest object located within FOV is greater than 30 meters. On such distances binocular disparity is close to zero, which means that any object appearing in binocular field will be perceived as a 2D object. Thus, by considering the fact that peripheral vision is always 2D the resulting visual scene will be interpreted by a human as a 2D panorama (panoramic mode).

When the object is localized, next it is necessary to approach to it. In this case, both binocular and monocular visions will operate similar to the previous stage. However, when a 30 meter range is reached, object of interest located within FOV will have nonzero binocular

disparity. This means that human's depth perception mechanism will be able to perceive an object of interest in a 3D space. In addition to that, it will be possible to estimate distance to the object using triangulation approach. As a result, perceived visual scene consisting of stereo and peripheral regions will represent 3D-panoramic vision (3D-P mode).

Next operation stages for a human include preparation stage or straight transition to interaction with an object. For both of these stages, vision system will perform stereo-panoramic acquisition of a scene as well. However, due to operating proximity with objects the Region of Interest (ROI) narrows and as a result most of the vision resources reallocate on particular details of an object located in the binocular region (3D mode).

In turn, each of the above examples requires adaptation mechanisms needed to satisfy various operational conditions. These adaptation mechanisms include dynamic change of video frame resolution, video frame rate, compression levels, etc. Consider following requirements examples associated with above described stages.

For the object of interest localization stage a 2D panorama has to have the highest possible resolution and low compression ratio in order to provide precise localization of the object of interest. The associated frame rate for this stage is not required to be high.

For the approaching stage, the frame rate and compression level requirements increase. That is due to maneuvering operations which require accurate monitoring of a stereo-panoramic scene while preserving moderate video resolution.

For the close proximity operations stage, the highest compression level on the peripheral region is required while providing highest resolution and video rate on binocular field.

Above provided examples demonstrate that in order to accommodate various tasks requirements the set of operation modes has to be considered for the design architecture. This will allow telepresence system to perform adaptation to the task similar to that performed by a human.

### 3.3. System Architecture Organization

The analysis of operation modes performed in the previous section made it possible to define the set of functional specifications needed for telepresence system general architecture development.

The proposed telepresence system architecture consists of two major subsystems: Video-Acquisition and Pre-Processing Subsystem and Video-Processing Subsystem, (see Figure 3.1). Both subsystems are interconnected via communication channel consisting of data and control buses.

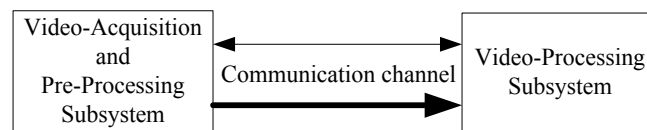


Figure 3.1: Telepresence System Architecture

The Video-Acquisition and Pre-Processing subsystem located at the slave side performs acquisition of video streams, pre-processing, video compression, transport stream formation and following transmission to the master side. In order to realize such subsystem the

following four components are employed (see Figure 3.2): 3D-P camera, video pre-processor, video compressor, transport stream combiner and transceiver.

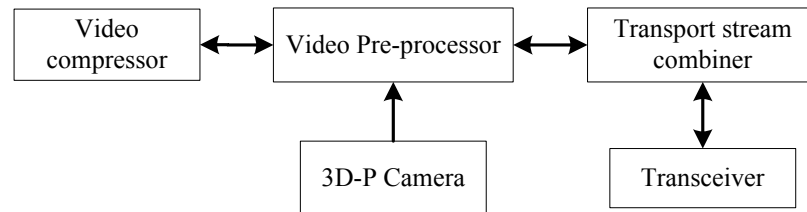


Figure 3.2: Video-Acquisition and Pre-Processing Subsystem Block-Diagram

The Video-processing subsystem located on the master side performs reception of the compressed video stream, distribution of video data to a storage memory and/or following decompression, execution of various image processing algorithms and output of the results to the display system. In order to realize such subsystem the following five components are employed (see Figure 3.3): transceiver, transport stream dispatcher, video decompressor, video-processor, storage memory controller and display.

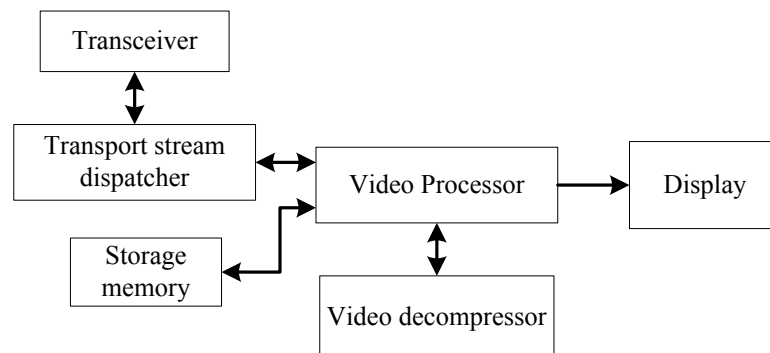


Figure 3.3: Video-Processing Subsystem Block-Diagram

The component set represented on both Figure 3.2 and Figure 3.3 could be realized in a three types of forms such as hardware (HW), firmware (FW) and software (SW).

The HW component is described as any electronic component manufactured as a circuit or as a set of electrically interconnected components located on a common area.

The FW component is circuit described on logic level using HDL and represented in a form of configuration bitstream.

The SW component is an instruction based program described on High-level or Low-level programming language and executed on sequential general-purpose processor such as microprocessor or microcontroller.

Following subsections of the chapter describe in detail the organization of the individual component proposed for the system architecture. Moreover, for each component the HW/FW/SW partitioning is performed.

### 3.3.1. 3D-P Camera

The 3D-P Camera is composed of four camera modules each consisting of an image sensor and an optic lens mounted on top of it. A light waves being refracted by an optic lens are captured by an image sensor composed of an array of color pixels. The light energy acquired by each pixel is converted to a voltage and then an additional circuit converts a voltage to a digital form.

The camera modules, illustrated on Figure 3.4, are positioned in such a way that they have the same vertical coordinate and are horizontally aligned.

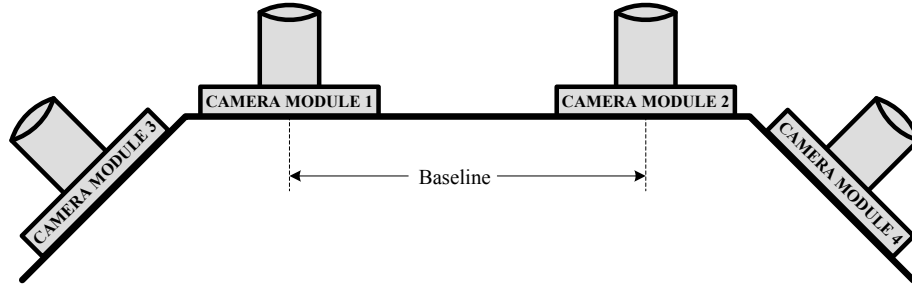


Figure 3.4: Camera Modules Organization

On horizontal axis two out of four camera modules are positioned on the central region with separation corresponding to human eyes baseline. The other two camera modules are positioned from left and right sides of a central region. Such organization allows having stereo-panoramic vision, described in Section 2.2 and according to which central region is responsible for binocular vision and corresponding left and right regions are responsible for peripheral vision.

In order to provide operation of a 3D-P Camera in various modes, each image sensors has to be configured according to the requested settings (e.g. resolution, frame rate, color gain, etc.). For such purposes Image Sensor Configuration Controller, illustrated in Figure 3.5 and consisting of a HW or FW component and associated SW component is considered for the 3D-P Camera. In addition to configuration task, Image Sensor Configuration Controller provides synchronization mechanism for all image sensors.

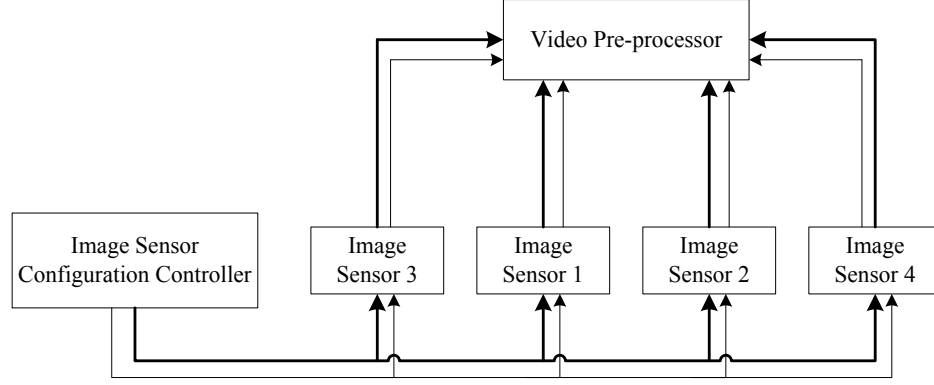


Figure 3.5: Image Sensor Configuration Controller Block-Diagram

It has to be noted that depending on particular application needs scaling of a number of camera modules does not require significant system modifications. For example, applications requiring in addition to stereo-panoramic vision ability to look up or/and down and observe 3-dimentional scene require in total 6 or 8 image sensors accordingly.

### 3.3.2. Video Compressor/Decompressor

In the design under consideration, mobile telepresence system is required to perform transmission of four real-time video streams from the slave to the master side. However, the amount of information to be transmitted is significantly higher than bandwidth of any wireless standard existing at the moment of writing this work. In order to estimate the bandwidth capacity required to transmit streaming video data from any number of image sensors operating on specific video standard, the following video rate ( $R_{video}$ ) equation is used:

$$R_{video} = [N_{sensors} * Res_{video} * Res_{pixel} * R_{frame}] \text{ bps}, \quad (3.1)$$



where  $N_{sensors}$  is a number of image sensors,  $Res_{video}$  is a video frame resolution,  $Res_{pixel}$  is a number of bits used to represent every pixel,  $R_{frame}$  is a video frame rate per second.

Consider an example of stereo-panoramic system setup described in Section 3.3.1 consisting of four image sensors. Each sensor is configured to resolution of 1280x720 (720p) pixels, frame rate corresponds to 30fps and each pixel is encoded using 8bits. By using Equation 3.1 the video data rate ( $R_{video}$ ), indicating the required transmission channel bandwidth ( $BW_{ch}$ ), is calculated as follows:

$$R_{video}(720p30) = BW_{ch} = [4 * [1280 * 720] * 8 * 30] = 843.75 \text{ Mbps}$$

It has to be noted, that the above example is just a representation of one of the possible modes of operation. As for the modes requiring higher quality of acquired video, associated BW requirements go over 1Gbps.

Consider two examples of the same stereo-panoramic system operating at standards 720p60 and 1080p60 accordingly. The required channel capacities are calculated as follows:

$$R_{video}(720p60) = [4 * [1280 * 720] * 8 * 60] = 1.64 \text{ Gbps}$$

$$R_{video}(1080p60) = [4 * [1920 * 1080] * 8 * 60] = 3.7 \text{ Gbps}$$

The following Table 3.1 demonstrates wireless standards that can be used in mobile applications and the associated maximum theoretical downlink and uplink transmission rates available to a single connection under ideal circumstances.

Table 3.1: Wireless Standards

Standard	Max Downlink (Mbit/s)	Max Uplink (Mbit/s)
802.11a	54	54
802.11b	11	11
802.11g	54	54
802.11n	200	200
802.16e	70	70

As it can be seen from the above table, none of the standards can satisfy BW requirements calculated in either of three examples provided above. Moreover, maximum theoretical BW of a standard means that an actual data rate is only 25%-50% of the theoretical rate. As a result, the following coefficient indicating a minimum data rate reduction degree required to meet channel capacity is expressed as:

$$k = \frac{R_{video}}{BW_{ch}} \quad (3.2)$$

Considering the results of bandwidth capacities required to accommodate three different video standards and by taking as an example an actual transmission rate of 802.11g standard the following three coefficients are calculated as:

$$k_1 = \frac{843.75 \text{ Mbps}}{27 \text{ Mbps}} = 31.3$$

$$k_2 = \frac{1.64 \text{ Gbps}}{27 \text{ Mbps}} = 62.2$$

$$k_3 = \frac{3.7 \text{ Gbps}}{27 \text{ Mbps}} = 140.3$$

From the above results, showing the minimum data rate reduction degree, the following conclusion can be made: the only way to perform transmission of real-time video data that

considerably exceeds channel capacity is to apply one of the video compression standards described in details in Section 2.6.

By taking into consideration all advantages and disadvantages of existing standards, the JPEG2000 standard was selected for the design implementation.

The primary criterion in selecting the standard was resistance to the bit errors that might occur during transmission. The example in Figure 3.6 demonstrates four images compressed at ratio 80:1 using MJPEG2000, MJPEG, MPEG-2 and MPEG-4 standards. The Bit Error Rate BER for all four cases is  $10^{-4}$ .

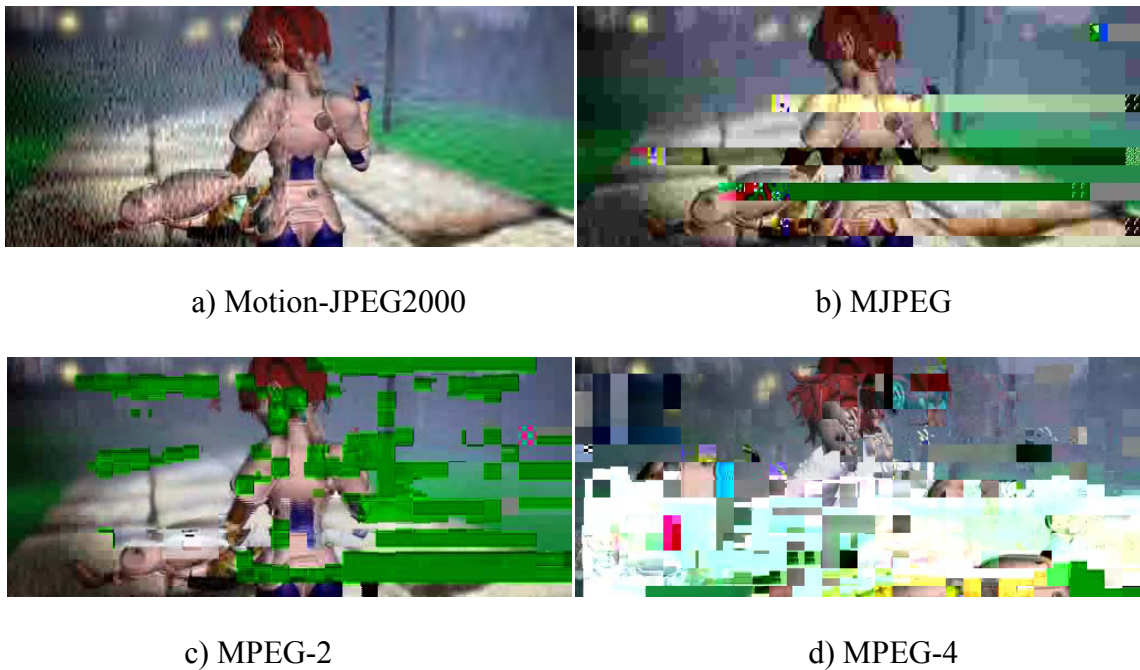


Figure 3.6: Resistance to Bit Errors of Various Compression Methods (courtesy to Analog Devices [25])

As it can be seen from the figures, DCT based compression standards demonstrate artifacts appeared as “blockiness”. Moreover, because of frame dependences and prediction algorithms used in MPEG standards, errors in one frame multiply and propagate to the following P and B frames. As a result, a set of corrupted frames will carry no useful visual information which makes task completion questionable.

On the other hand wavelet based JPEG2000 standard demonstrates the best image quality out of four. In addition to that, since each frame is compressed independently of other frames the transmission errors in one do not propagate to the successive frames. As a result, loss of one frame out of 30 for example, doesn’t create any visual perception problems that might influence on any decision during operation or operation in general.

Moreover, the above described advantages of JPEG2000 standard increase the productivity of image processing algorithms. The disparity calculation, image merging, image rectification algorithms take advantage of image quality obtained after decompression.

As it was mentioned in Section 3.3.1, depending on application needs different number of camera modules might be required. However, in order to perform compression of video streams and their following decompression a number of compressor/decompressor units required for both subsystems has to be determined. An architectural decision is based on following factors. Firstly, the amount of information coming in parallel from independent camera sources might not be equal to the processing bandwidth of a single compressor /decompressor unit. That is why a parallel stream processing is needed. In this case a

number of compressor/decompressor units working in parallel can be determined from the following equation:

$$N_{compr/decomp} = \frac{R_{video}}{P_{rate}} = \frac{N_{sensors} * [Res_{video} * Res_{pixel} * R_{frame}]}{[Res_{video} * Res_{pixel} * R_{frame}]}, \quad (3.3)$$

where  $R_{video}$  is a video rate defined by equation 3-1,  $P_{rate}$  is a processing rate of a single compressor/decompressor unit associated with a particular video standard.

It has to be noted, that in case video standard of a stream corresponds to the video processing abilities of a compressor /decompressor unit Equation 3.3 gives:

$$N_{\frac{compr}{decomp}} = N_{sensors}$$

Secondly, existence of additional compressor/decompressor units allows increase reliability of the system. In situation when compressor/decompressor unit associated with more prioritized visual field fails, switching to the resources associated with less prioritized visual field can be performed.

In order to simplify the architecture design of video compressor/decompressor, the case described by Equation 3.3 is taken into account. Thus, all four parallel video streams produced by stereo-panoramic system are associated with individual compressor /decompressor unit. This involves four HW compressors for the slave side and four HW decompressors for the master side.

In order to perform configuration, control of compressor/decompressor and exchange data with system components the Image Compressor/Decompressor Controller component consisting of FW and SW is considered for the system.

The above described architecture organization of slave and master side is identical in terms of connectivity among the blocks. Therefore, it was combined on a single block-diagram shown on Figure 3.7. Despite the controller component is illustrated as separate block, it is a part of internal architecture of video pre-processor/video processor HW which is described in Section 3.3.6.

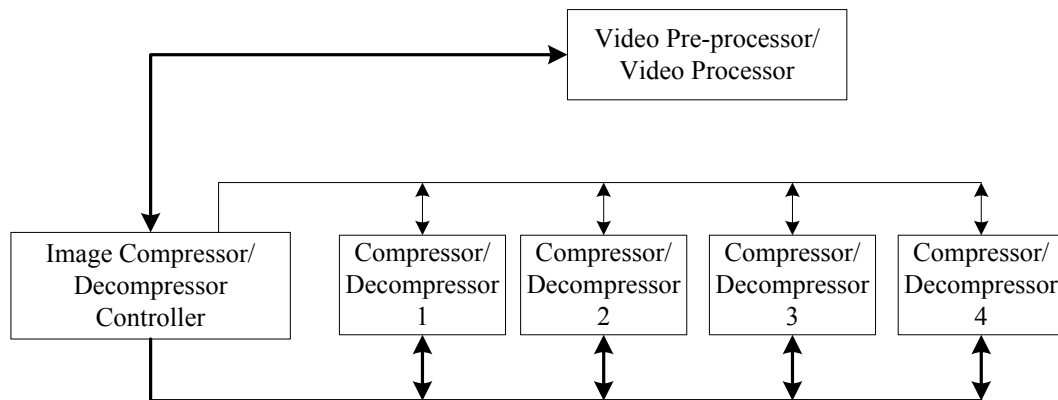


Figure 3.7: Image Compressor/Decompressor Controller Block-Diagram

### 3.3.3. Transceiver

Transceiver is a component that provides functionality of both a transmitter and a receiver. Being integrated to the slave and master subsystems it allows bidirectional exchange of the control information as well as transmission of video data in a form of a transport stream, described in Section 3.3.5, from the slave to the master side. The block-diagram of the transceiver system is shown on Figure 3.8.

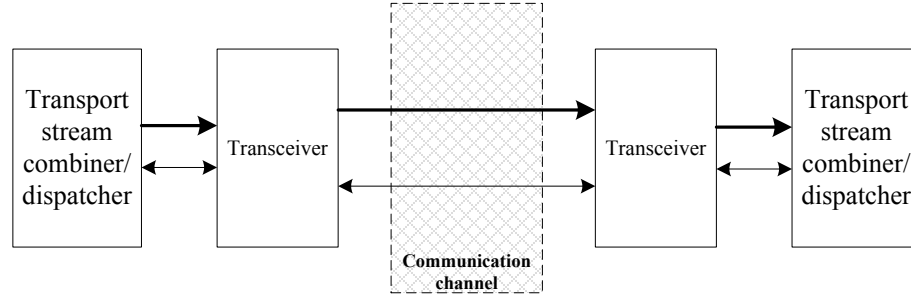


Figure 3.8: Transceiver Block-Diagram

In order to provide mobility for both subsystems a wireless communication is considered for the system implementation. Moreover, due to a wide range of applications, transmission on a satellite/terrestrial levels using various digital modulation schemes (e.g. COFDM, TDM) has to be supported. Therefore, the proposed transceiver architecture is equipped with a HW serializer/deserializer component. It performs data conversion between parallel and serial interfaces in each direction. The principle of transceiver block operation is illustrated on Figure 3.9. A data to be transferred is applied on parallel input interface of serializer. Being converted into serial form it is then directed to modulator which performs modulation of an analog carrier signal by a digital bit stream. Next, modulated signal is directed to RF transceiver which performs an actual data transmission. On the receiver side, the received RF signal is demodulated, i.e. converted from analog to digital form and directed to the deserializer. In turn, deserializer converts serial data into parallel form and outputs it to parallel output interface.

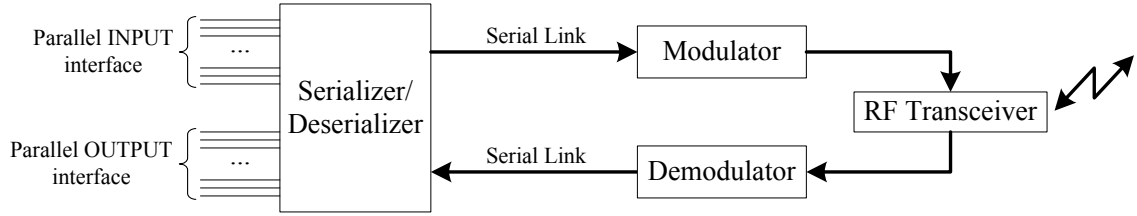


Figure 3.9: Transceiver Operation Principles

#### 3.3.4. Display

The display located on the master side of mobile telepresence system has to provide an operator with the vision of a remote scene. Moreover, the stereo-panorama acquired on a slave side has to be displayed in a way natural to a human vision, thus providing full immersion with a remote environment.

In order to realize such requirements specific organization of display projectors, circular polarized filters and projection screens has to be performed.

Out of existing stereoscopic display methods described in Section 2.5, circular polarization was selected due to a lack of any side effects for a human being and the good visual quality provided by the method. Side effects of other display systems are expressed in headaches caused by flickering of shutter glasses, reduced color perception caused by anaglyph glasses, rapid fatigue due to heavy equipment wearing in the case of HMD. It has to be noted that a visual quality obtained by circular polarization is dependent on polarizer material quality, image projection device performance and projection screen material quality.



The proposed stereo-panoramic display consisting of four display projectors and three projection screens is shown on Figure 3.10. Projector 1 and 2 associated with a left and right eye are equipped with circular polarized filters and are positioned one on top of another. Projectors 3 and 4 associated with peripheral vision of left and right eye are positioned on left and right side from central projectors.

Both central and panoramic projectors have corresponding projection screens. Projection lines coming from projector 1 and 2 fall on the central screen thus displaying stereoscopic vision, while lines coming from projectors 3 and 4 fall on panoramic left and right screens thus displaying peripheral left and right visions.

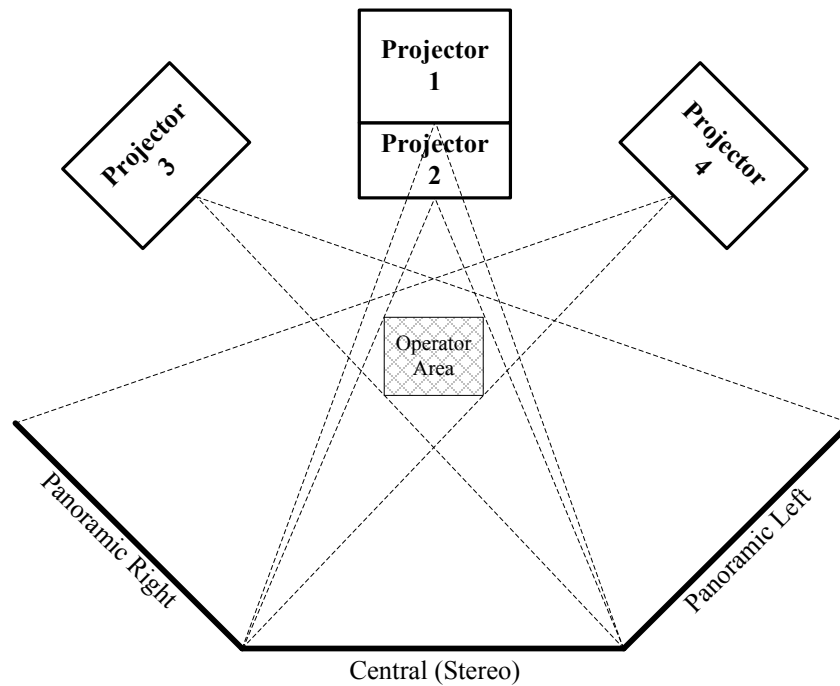


Figure 3.10: Stereo-Panoramic Display

As a result, an operator wearing glasses polarized in accordance with projector 1 and 2 filters polarization and located in front of display will have stereo-panoramic perception of a scene similar to that perceived in a real world environment.

### 3.3.5. Transport Stream Combiner/Dispatcher

Prior to transmission stage, multi-stream video data that has passed all pre-processing and compression stages has to be combined into a transport stream. Transport stream specifies a container format which encapsulates packetized elementary streams, error correction and stream synchronization features needed to maintain data integrity associated with satellite or terrestrial transmission.

A typical transport stream packet consisting of header and a payload is illustrated on Figure 3.11. A header contain synchronization bits (SYNC), packet priority bits (PR), error indicator bits (EI), packet identifier bits (PID), stream adaptation control bits (SAC), cyclic redundancy check bits (CRC). Whereas payload consists of an actual data and error correction codes. An example of transport stream organization is MPEG transport stream, specified in MPEG-2 Part 1 standard [51].

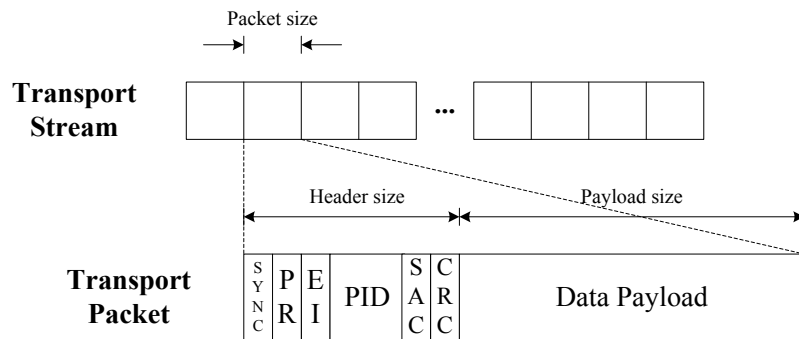


Figure 3.11: Transport Stream Packet

In order to perform assembly of multi-stream video data into video transport stream, a combiner component implemented in a form of FW loadable to HW is proposed for the system. The block-diagram of the combiner is shown on Figure 3.12.

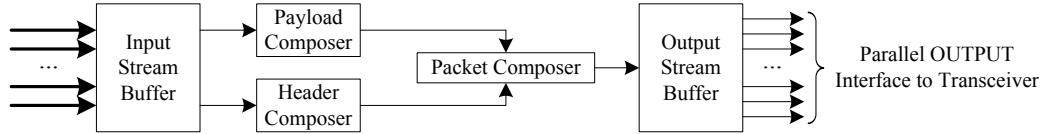


Figure 3.12: Transport Stream Combiner Block-Diagram

On a first stage data coming from parallel video streams is buffered on input stream buffer. Next, each video stream is split onto data portions corresponding to payload size. At the same time for each data portion the header information is calculated. When header information followed by data payload is ready, composition into transport packets takes place. Thus, for each video stream individual transport packets are generated and then forwarded to output stream buffer. Finally, each transport packet is outputted to parallel input interface of transceiver system in accordance to the order of video data associated with each stream.

On a receiver side, in order to perform an inverse operation with a multi-stream video data, i.e. disassembly of transport streams onto separate video data streams, a dispatcher component implemented in a form of FW loadable to HW is proposed for the system. The block-diagram of the dispatcher is shown on Figure 3.13.

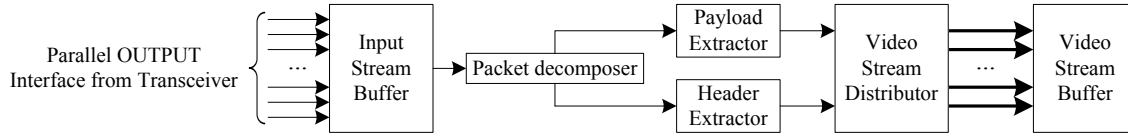


Figure 3.13: Transport Stream Dispatcher Block-Diagram

The transport stream packets containing data from multiple sets of streams are received by the transceiver and outputted through its parallel interface. Following this, data is being buffered on input stream buffer. Then, packet decomposer performs extraction of header information with a data payload from the packet. Next, analysis of header information and data payload followed by service procedures (e.g. error correction and data integrity verification, packet priority analysis, etc) are performed. Uncorrupted data is then split on data portions corresponding to number of video streams encapsulated onto a payload and distributed to video stream buffer for further processing.

### 3.3.6. Video Pre-Processor/Video Processor

In order to provide control, data exchange, synchronization among components located within video-acquisition/pre-processing and video processing subsystems as well to provide execution of various image processing algorithms two FW components are proposed for the system architecture. The video pre-processor FW component located on the slave side subsystem and video processor FW component located on the master side subsystem.

Due to both subsystems requirements associated with real-time video stream processing as well as requirement to support various modes of system operation described in Section

3.2 lead to video pre-processor and video processor consideration in a form of reconfigurable HW described in Section 2.7.

Following subsections describe the video acquisition and color interpolation subcomponents associated with video pre-processor FW component. Next, the storage memory controller, display driver and image compressor/decompressor controller subcomponents associated with video processor FW component are presented.

It has to be noted, that the stereo image rectification and image merging subcomponents are also considered for the system architecture. However, the discussion of these components is out of scope of this thesis work.

### 3.3.6.1. Video Acquisition Subcomponent

In order to perform acquisition of data coming from multiple video streams generated by image sensors the video acquisition subcomponent is proposed for the system architecture. The block-diagram of this subcomponent is shown on Figure 3.14.

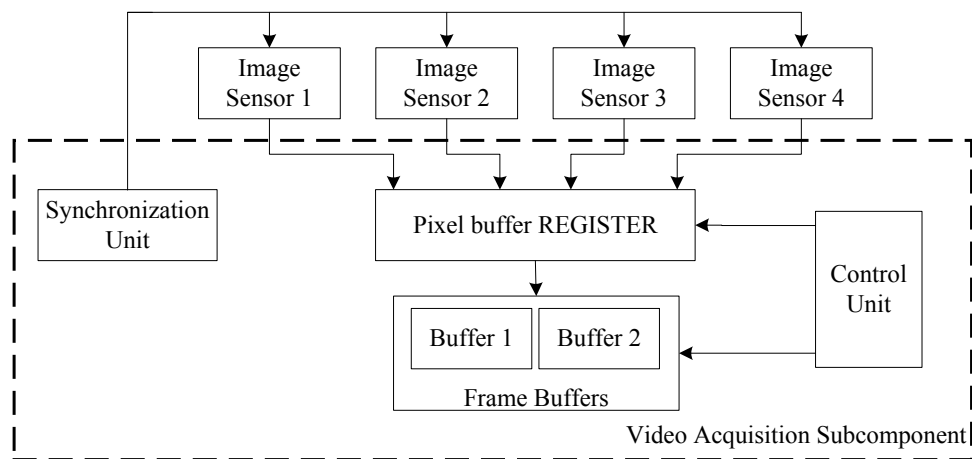


Figure 3.14: Video Acquisition Subcomponent Block-Diagram

Providing synchronous operation of all image sensors employed by the system, each of them generates pixel data every readout operation cycle. The pixels from all sensors then are buffered on an intermediate register. That is done to form a single pixel data container which is then buffered in a frame buffer. By the end on pixel transmission of a frame the first buffer becomes full and the data becomes ready for further processing. At that moment, the second buffer catches up pixel data from the next frame when it is available and the process repeats again.

### 3.3.6.2. Color Interpolation Subcomponent

As it was already mentioned in Section 3.3.1 image sensor consists of an array of color pixels. The color of each individual pixel is defined by the color filter array (CFA) called Bayer pattern [52].

The concept of the Bayer Pattern is based on the observation that the human eye is more responsive to the green channel of emissive light sources than to the red and blue channels. This information is used to aid in the design of image and video sensors: rather than include red, green and blue sensors for every pixel of a sensor, only one sensor is used. The sensor pattern is shown in Figure 3.15 below. As can be seen, the green channel has twice the sensor density of the red and blue channels, and therefore twice the resolution.

	0	1	2	3	4	5	6	
0	G	R	G	R	G	R	G	
1	B	G	B	G	B	G	B	
2	G	R	G	R	G	R	G	
3	B	G	B	G	B	G	B	...
4	G	R	G	R	G	R	G	
5	B	G	B	G	B	G	B	
6	G	R	G	R	G	R	G	
				⋮				

Figure 3.15: Bayer Color Pattern

Since each pixel is filtered to record only one of three colors, the data from each pixel cannot fully determine color on its own. To obtain a full-color image, various interpolation algorithms [55],[56],[57],[58] are used to recover a set of red, green and blue values for each pixel point. Different algorithms requiring various amounts of computing power result in varying-quality of final images. For the proposed system bilinear interpolation is selected due to relatively low computational requirements and at the same time acceptable quality for video applications. The operation of bilinear interpolation algorithm is demonstrated for blue and green channels of two pixels in Figure 3.16 below.

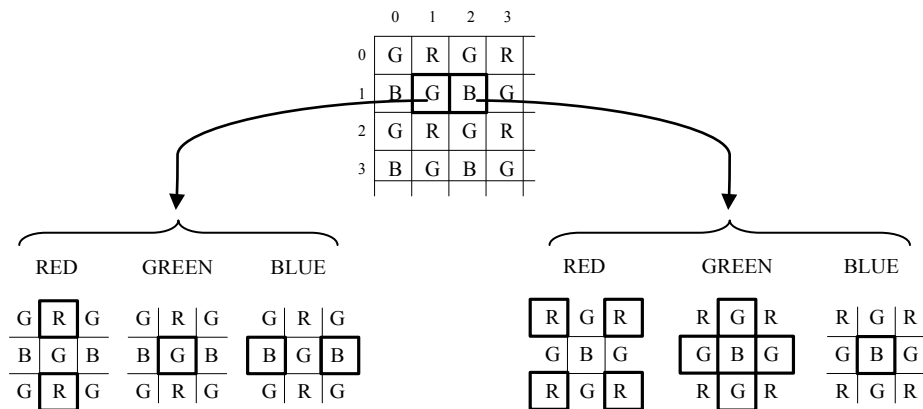


Figure 3.16: Bilinear Interpolation Algorithm

By performing an averaging of neighboring values for a particular color channel all three color channels are calculated for every pixel. Following equations are used to interpolate all three color channels for every Red, Green and Blue pixel location.

$$R = R; \quad G = \frac{G_T + G_B + G_L + G_R}{4}; \quad B = \frac{B_{TL} + B_{TR} + B_{BL} + B_{BR}}{4}; \quad (3.4)$$

$$R = \frac{R_T + R_B}{2}; \quad G = G; \quad B = \frac{B_L + B_R}{2}; \quad (3.5)$$

$$R = \frac{R_{TL} + R_{TR} + R_{BL} + R_{BR}}{4}; \quad G = \frac{G_T + G_B + G_L + G_R}{4}; \quad B = B; \quad (3.6)$$

Subscripts for color channels define its location. For example, T-top, B-bottom, L-left, R-right, TL-top left, TR-top right, BL-bottom left, BR-bottom right.

The block-diagram of the proposed color interpolation subcomponent is illustrated on the Figure 3.17.

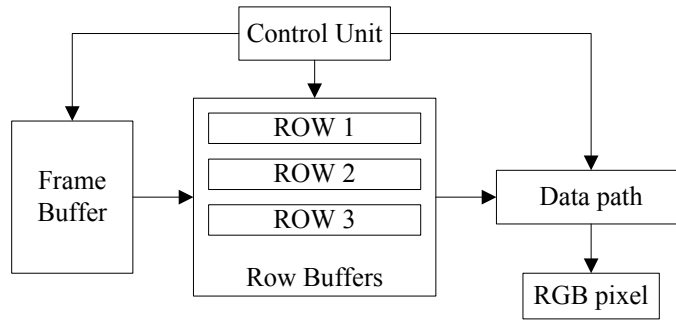


Figure 3.17: Color Interpolation Subcomponent

An image frame from a video stream sequence being buffered on frame buffer is distributed row-by-row to a three separate row buffers. That is done to provide the data path with color pixels from the surrounding pixel locations and thus satisfy requirements of



bilinear interpolation algorithm described above. After the requested pixel values are received into the data path, interpolation of the required colors using Equations 3.4, 3.5, 3.6 is performed. As a result, three color components associated with each pixel are generated.

### 3.3.6.3. Storage Memory Controller Subcomponent

As it was mentioned in Section 3.3 the compressed video stream being received on a master subsystem can be forwarded to the storage memory. Therefore, in order to perform writing or reading operations with memory, the storage memory controller subcomponent implemented in a form of FW is proposed for the master subsystem. The block-diagram of the subcomponent is shown on the Figure 3.18 below.

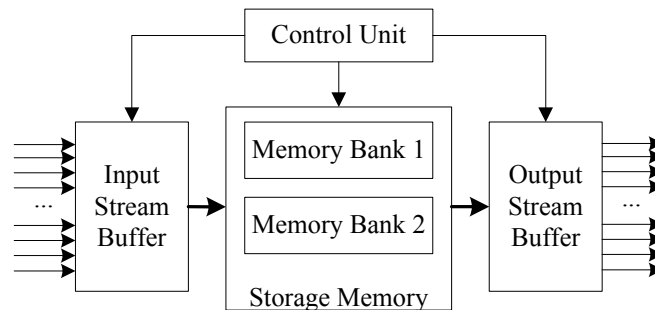


Figure 3.18: Storage Memory Controller Subcomponent Block-Diagram

Depending on the requested operation, the control unit initiates write or/and read operation with the storage memory banks. For write operation, the data coming from the dispatcher interface is buffered on input stream buffer. Then, an actual write operation to one of the memory banks is performed. For read operation, the control unit initiates reading

cycle of either of the banks. As a result, the compressed data is outputted to the output stream buffer for further processing steps.

#### 3.3.6.4. Display Driver Subcomponent

In order to provide the stereo-panoramic display system described in Section 3.3.4 with multiple streams of processed video data the display driver subcomponent is proposed for the video processor component.

As it can be seen from the block diagram on the Figure 3.19, a processed data from the data path is distributed to four RGB registers. Each of these registers hold an RGB pixel values needed to output a single pixel on a display.

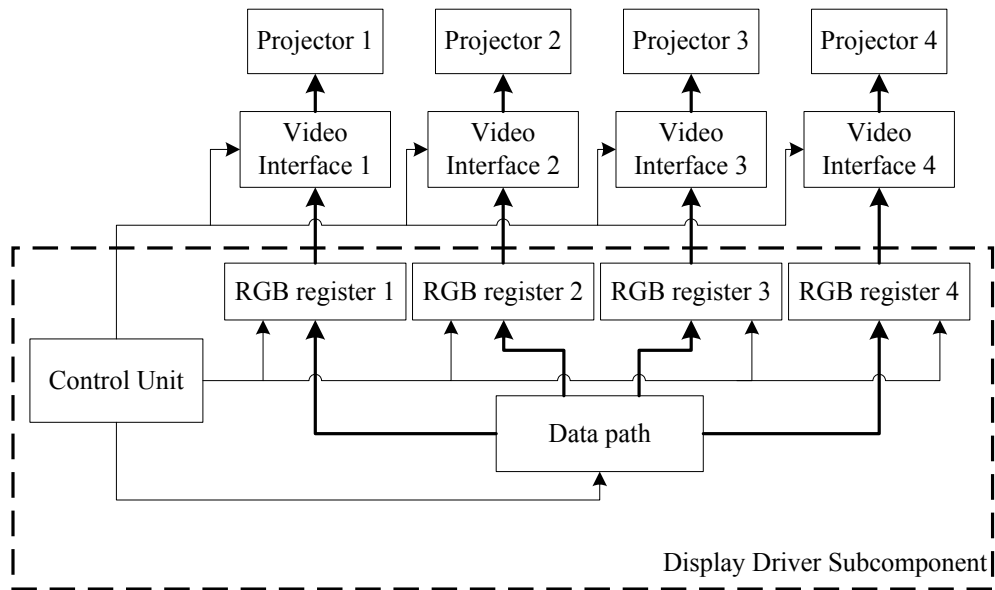


Figure 3.19: Display Driver Subcomponent Block-Diagram

When all four registers had received pixel data corresponding to each video stream, synchronous application of control signals along with a pixel data to video interface is initiated by the control unit. As a result, every display projector attached to video interface synchronously receives a pixel values in accordance to timing specifications of a display resolution standard (e.g. XGA, 720p, 1080p, etc.) selected for the system.

### 3.3.6.5. Image Compressor/Decompressor Controller Subcomponent

In order to perform management of operation modes, control, data exchange within the system requires implementation of dedicated controller for each compressor /decompressor HW component. Therefore, the Image Compressor/Decompressor controller subcomponent illustrated on Figure 3.20 is proposed for the system.

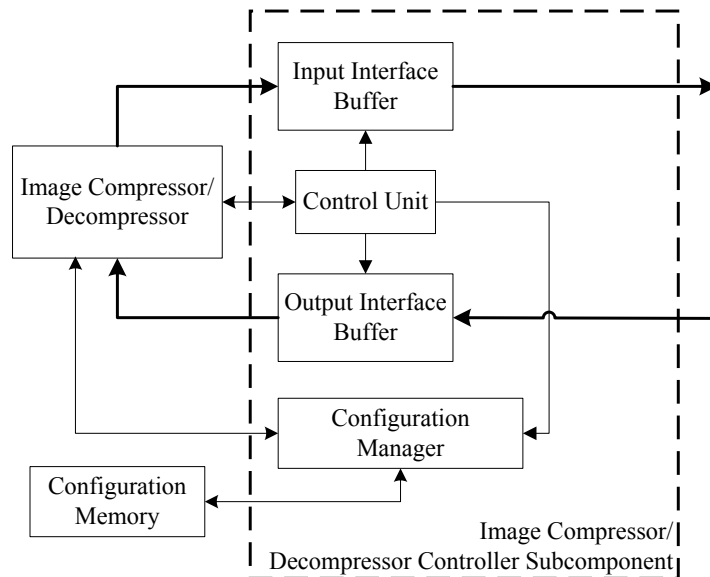


Figure 3.20: Image Compressor/Decompressor Controller Subcomponent Block-Diagram

The part of the block-diagram which is isolated in dashed rectangle is a FW part which is loadable to either of subsystem processors. The other two blocks are HW elements.

Prior to operation with an image compressor/decompressor HW the configuration manager performs loading the required compressor or decompressor functionality from the configuration memory. If the compressor functionality was selected, the control unit of image compressor/decompressor subcomponent performs the control signals exchange with the HW part. After that, the data to be compressed is supplied from output interface buffer to the input of the compressor. Being processed the compressed data is outputted from compressor interface to the input interface buffer of FW subcomponent and then forwarded to the combiner subcomponent described in Section 3.3.5.

The similar procedural steps are performed in case decompressor functionality is required. However, the input to the decompressor will be a compressed data and the output from it will be decompressed data that is then forwarded to image processing subcomponents.

### 3.4. Conclusion

In this chapter, the general architecture of the platform for 3D-Panoramic Telepresence System was proposed and developed. Due to the specifics of telepresence system in hazardous environments, two operational sites were introduced. The master site – teleoperator’s control center, and the slave site – remotely controlled system deployed in a hazardous area and performing operations requested from the control center. In order to provide adaptation to various stages of a task execution, the set of operational modes

including Panoramic, 3D-Panoramic and 3D was defined and analyzed. Based on the modes' specifications as well as information presented in Chapter 2, two subsystems corresponding master and slave sites were introduced. As a result, the architecture organization of the functional components in the Video-Acquisition and Pre-Processing Subsystem as well as in the Video-Processing Subsystem was determined and described in detail. In addition to this, partitioning onto hardware, firmware and software for all components was performed.

## 4. IMPLEMENTATION OF THE PLATFORM FOR 3D-PANORAMIC TELEPRESENCE SYSTEMS

### 4.1. Introduction

The objective of this chapter is to describe the implementation specifics and prototyping of the reconfigurable platform for the class of the 3D-Panoramic (3D-P) Telepresence Systems. This implementation is done according to the architectural organization presented in Chapter 3. The platform prototype will allow testing and verification of all aspects of system performance. The collected data will be used to perform the analysis of the results in Chapter 5.

The implementation of the proposed system is divided into two stages. The first stage focuses on implementation and prototyping of all major components in order to test feasibility of the proposed architecture. Therefore, the first platform prototype of the Video-Acquisition and Pre-Processing subsystem was designed including hardware, firmware and software parts. In addition to this, the Video Processor Firmware Component was created and deployed in the Video Processing Subsystem based on an existing hardware platform (MARS).

The complete platform prototype was then modified for the next version according to the analysis of the performance results gained on the first prototype of the platform.

Therefore, the second stage of platform implementation followed up the feasibility stage by introducing modification according to recommendations based on experimental results.

The following section of this chapter begins with a presentation of the system associated with the first stage of implementation. Firstly, the block-diagram of the entire architecture and corresponding functional and technical specifications are introduced. Then, the details

of implementation relevant to hardware, firmware and software components of the subsystems are provided.

Presentation of the system associated with a second stage of implementation is provided next. Architectural organization, functional and technical specifications as well as hardware implementation details are provided in the same format as the first stage.

## 4.2. System Design Implementation (stage 1)

The complete 3D-P Telepresence System consists of two main hardware subsystems: Video-Acquisition and Pre-Processing Subsystem (which will be referred to as 4-Vision subsystem) and Video-Stream Processing and Displaying Subsystem (which will be referred to as MARS subsystem). The block-diagram of the system architecture is shown in Figure 4.1 below. Thick lines represent data communication busses as opposed to thin lines representing control and synchronization signals.

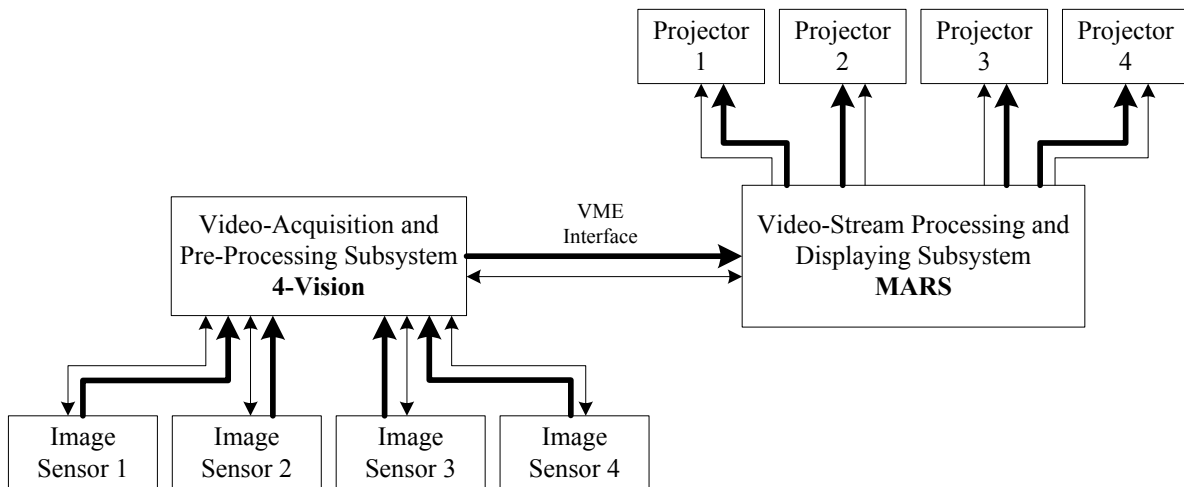


Figure 4.1: Architecture of 3D-P Telepresence System

The 4-Vision subsystem performs stereo-panoramic video acquisition from four image sensors incorporated to the 3D-P Camera and organized in accordance to Section 3.3.1. Then, it combines multi-stream data into packets and transmits them to MARS subsystem.

The MARS subsystem receives data packets, performs decomposition of data from each packet and proceeds with operations needed to prepare the video streams for the display. Finally, the stereo-panoramic video stream is outputted to four projectors of the Display described in Section 3.3.4.

The above described functionality is accompanied with technical specifications for the system. This includes video streams requirement to satisfy resolution of 1024x768 pixels (XGA standard), at 30 fps and 8 bits per color channel; stereo-panoramic horizontal field of view to be close to that of a an average human vision providing from 50° to 60° in stereoscopic region and from 120° to 140° of an overall panorama; stereoscopic acquisition from viewpoints separated in accordance to human eyes and equal to 60 mm; minimized power consumption due to consideration for the mobile applications.

#### 4.2.1. 4-Vision Subsystem

The following sections describe the 4-Vision subsystem hardware architecture as well as architecture of firmware and software components associated with the subsystem. Despite the 3D-P Camera component is organized as a separate plug-in hardware it is considered to be a part of 4-Vision subsystem as well. Therefore in order to reveal all aspects related to video acquisition and processing on the 4-Vision subsystem, the next section begins with the 3D-P Camera hardware architecture organization.



#### 4.2.1.1. 3D-P Camera Hardware Architecture

The 3D-P Camera consists of four uniform camera modules which are mounted on the support frame. The modules' organization and arrangement on the frame is done in accordance to Section 3.3.1. The baseline distance separating the central camera modules responsible for stereoscopic region is set to 60mm. This value corresponds to an average human eyes separation distance. Figure 4.2, illustrates the organization of the 3D-P Camera alongside the 4-Vision subsystem.

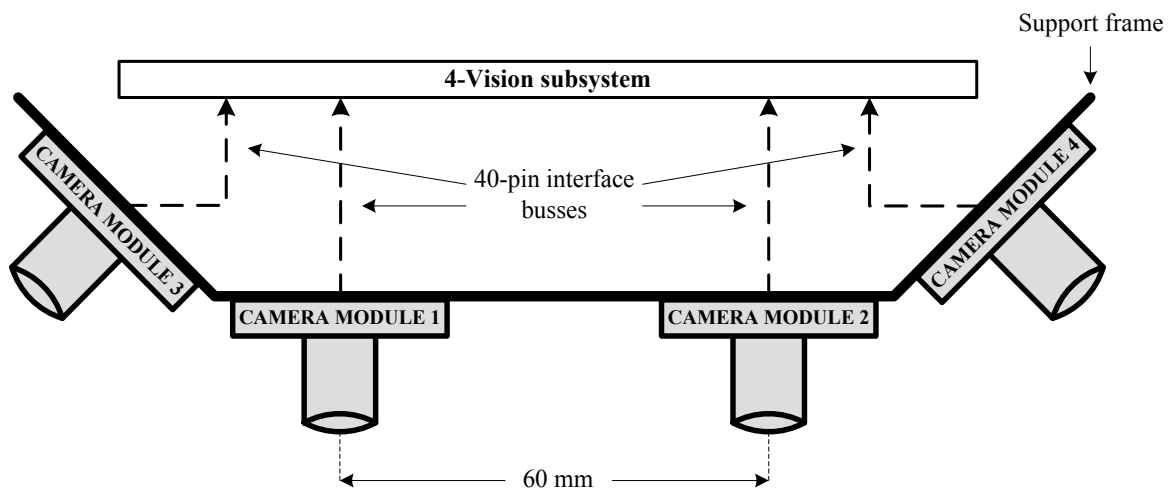


Figure 4.2: 3D-P Camera Organization

Each camera module of the setup features an image sensor, optic lens, lens mount and 40-pin interface connector.

#### 4.2.1.1.1. Selection of proposed elements for the camera module

Selection of an image sensor element requires consideration of video quality specifications such as frame resolution, frame rate and pixel resolution described in Section 4.2. Therefore, the following calculation of a video data rate ( $R_{video}$ ) based on video specifications is performed in order to estimate an operation frequency and the data bus dimension of the required image sensor:

$$\begin{aligned} R_{video} &= [Res_{video} * Res_{pixel} * R_{frame}] = \\ &= [(1024 * 768)pixel * 8bit * 30fps] = 22.5MBps , \end{aligned} \quad (4.1)$$

where  $Res_{video}$  is a video frame resolution,  $Res_{pixel}$  is a number of bits used to represent every pixel,  $R_{frame}$  is a video frame rate per second.

From the result produced by Equation 4.1 it can be concluded that in order to provide the required data rate, an image sensor has to operate at minimum 22.5MHz clock and perform parallel pixel byte data output on every clock cycle.

In order to satisfy such requirements the ½ inch CMOS image sensor from Aptina (MT9T031) [60] was selected for the camera module implementation. Through the configuration registers settings it provides the required frame resolution of 1024x768 pixels, frame rate of 30fps and other additional parameters such as exposure time, color gain etc. The MT9T031 uses a Bayer color pattern described in Section 3.3.6.2. The operational frequency of the image sensor is fixed to 48MHz regardless of actual data rate. The pixel

output resolution is fixed as well and equal to 10 bits. Such an overhead of resources provides an additional potential for the future system implementation stages.

The selection of an optic lens for the camera module also requires consideration of technical specifications. Therefore, out of great number of lenses presented on the market the Fujinon (DF6HA-1B) [61] was selected based on the following criterions:

- Compatibility with ½ inch image sensor model;
- Fixed focal operation equal to 6mm;
- Wide iris range equal to F1.2-F16;
- Horizontal angle of view equal to 56°;
- C-mount standard support;
- Cost-performance

The third element associated with the camera module is a lens mount. Since it is required to support the selected Fujinon lens, the PointGery C-mount holder [62] is selected for the system implementation. The C-mount provides threaded joint with the optic lens and allows strong attachment with the camera module by means of fixation screws.

The final element used in a camera module is an interface connector. Due to requirements associated with the number of transmission lines needed to exchange control, synchronization and data signals between camera module and the 4-Vision subsystem the 40-pin parallel ATA socket connector with 2.54mm pitch is selected for the camera module implementation. In addition to signal transmission, this interface provides supply voltage as

well as ground for the module power circuitry. The layout of the interface connector is presented in the following section.

#### 4.2.1.1.2. Principles of Camera Module Operation

As it was already mentioned, the selected CMOS sensor provides various configuration settings such as frame rate, frame resolution, exposure time, color gain and other parameters. All of these parameters are associated with the configuration registers of a sensor and are programmed through a two-wire serial interface (signals S\_CLK and S\_DATA) incorporated to 40-pin interface connector. The programming procedure is performed by the Microcontroller which is described in Section 4.2.1.2.2 and the associated 3D-P Camera configuration software component which is described in Section 4.2.1.4. It has to be noted, that after power OFF/ON cycle image sensor resets its settings to the factory default ones (2048x1536, 12fps). Therefore, sensor reconfiguration is performed on every power ON cycle of the system.

In order to provide operation of the image sensor the master clock frequency equal to 48MHz is applied to corresponding clock input (signal CLKIN) in accordance to clock distribution scheme described in Section 4.2.1.2.2. As a result, the pixel data (DOUT [9:2]) and associated control signals (FRAME\_VALID and LINE\_VALID) synchronized with the pixel clock (PIXCLK) at 48MHz are outputted by the image sensor. Figure 4.3 illustrates the above described principles and provides general timing information in respect to a single frame.

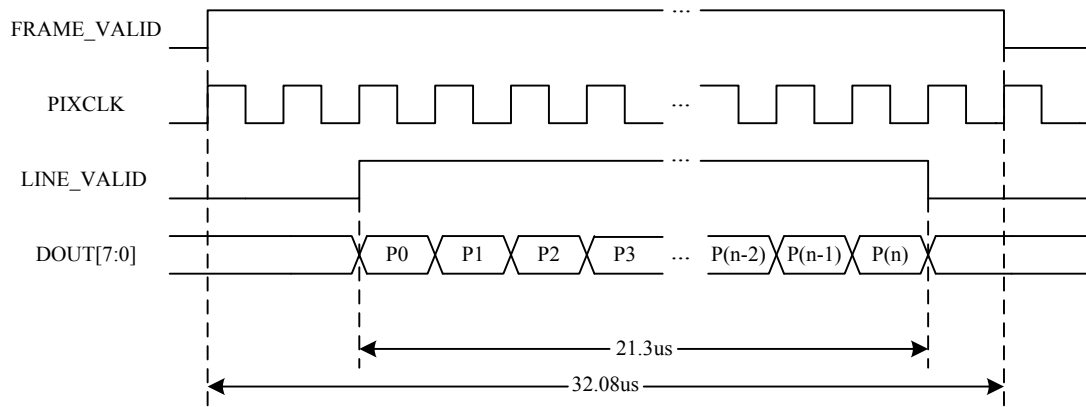


Figure 4.3: Image Sensor General Timing Diagram

Every pixel value generated during active frame and line periods are stabilized and ready for latching on a falling edge of the pixel clock generated by the image sensor.

As it was mentioned in the previous section, 40-pin parallel ATA socket interface connector is selected for the camera module implementation. The arrangement of control, synchronization, data and power lines is illustrated in Figure 4.4 below.

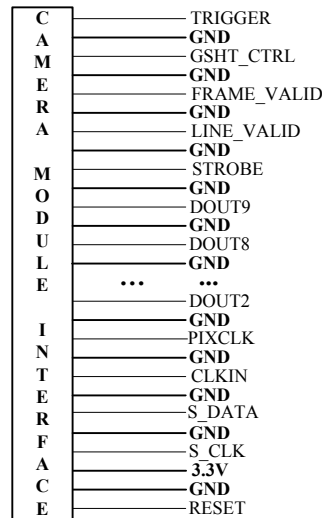


Figure 4.4: Camera Module Interface Connector

There are twenty ground lines presented on the interface. These lines are specifically interlaced with every other signal line in order to reduce a signal-to-signal crosstalk.

The following Table 4.1 summarizes the functional description of signal lines incorporated to camera module interface:

Table 4.1: Image Sensor Signal Lines

<b>Signal</b>	<b>Description</b>
TRIGGER	Control over snapshot sequence
GSHT_CTRL	Global shutter control
FRAME_VALID	Indicator of a frame valid period. Active HIGH
LINE_VALID	Indicator of a line valid period. Active HIGH
DOUT[9:2]	Pixel data output, DOUT[9] (MSB), DOUT[2] (LSB)
PIXCLK	Pixel data outputs are valid during falling edge of this clock. The frequency equals to master clock frequency, i.e. 48MHz
CLKIN	Master clock input of the sensor. Frequency equals to 48MHz
S_DATA	Data line for serial programming interface
S_CLK	Clock line serial programming interface
RESET	Activates (LOW) asynchronous reset of sensor. All registers change to factory defaults.

#### 4.2.1.2. 4-Vision Subsystem Hardware Architecture

The 4-Vision subsystem is responsible for 3D-P Camera configuration/control, acquisition of stereo-panoramic video from four video streams and delivery it to MARS subsystem in an appropriate format for processing.

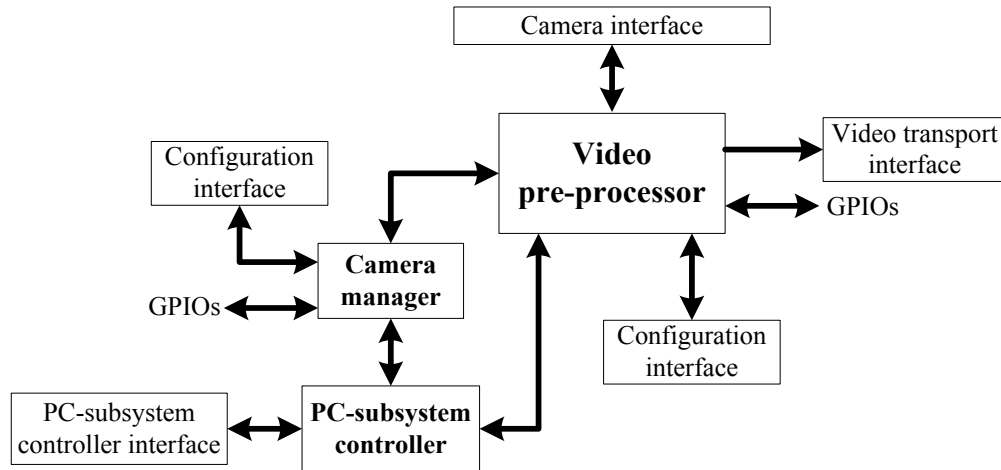


Figure 4.5: 4-Vision Subsystem Hardware Architecture

In order to fulfill all of these requirements the subsystem's hardware architecture is organized as illustrated in Figure 4.5 using the following hardware elements and communication interfaces:

**Elements:**

- Video Pre-Processor;
- 3D-P Camera manager;
- PC-subsystem controller.

**Interfaces:**

- 3D-P Camera interface;
- Video transport interface;
- Video Pre-Processor configuration interface;
- 3D-P Camera manager configuration interface;
- PC-subsystem controller interface.

In addition to the above mentioned elements and interfaces the LED indicators and mechanical actuators are incorporated into the subsystem through General Purpose Inputs Outputs (GPIOs) for the additional testing and control purposes.

#### 4.2.1.2.1. Selection of the Proposed Elements for the 4-Vision Subsystem

The selection of video pre-processor element as a main processor for the subsystem implementation is based on the performance specifications and operational requirements. Following analysis illustrates the performance estimation of FPGA versus 32bit RISC Microprocessor involving acquisition of pixel data from each of four video streams into 32bit data packet and following transmission of a packet along with a control signals.

According to the image sensor specification and principles of its operation provided in the previous section, every color pixel is generated during active frame and line periods at 48MHz pixel clock. This creates a time window equal 20.83ns available for execution of the required operations.

In case of FPGA based implementation the latency associate with the processing of the first pixel in every line is equal to two clock cycles. However, the consequent pixels in the line are processed on every clock cycle, one after another. The following Figure 4.6 illustrates a graphical representation of this process.



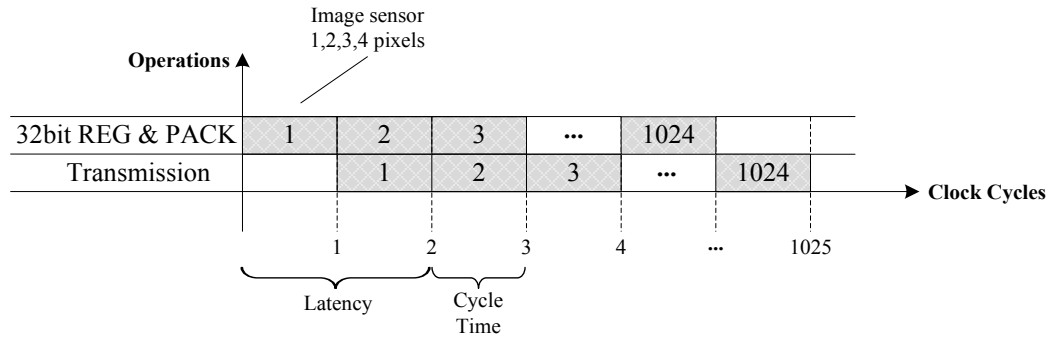


Figure 4.6: FPGA Based Implementation Diagram

Therefore, the total execution time ( $T_{exe}^{FPGA}$ ) required for processing of a single pixel row on FPGA based processor is calculated as:

$$T_{exe}^{FPGA} = [L + (n - 1) * C_{time}] * T_{cycle} = \quad (4.2)$$

$$= [2c.c. + (1024 - 1)c.c. * 1c.c] * 20.83ns = 21.35 \mu s,$$

where L is a number of cycles required for obtaining the first result, n is a total number of the required cycles,  $C_{time}$  is a cycle time and  $T_{cycle}$  is a clock period at 48MHz.

In order to perform the identical pixel processing operations on every 1024 pixel in a row, a 32bit RISC Microprocessor operating on 48MHz frequency is required to execute instructions in accordance to the flow char illustrated of Figure 4.7 below.

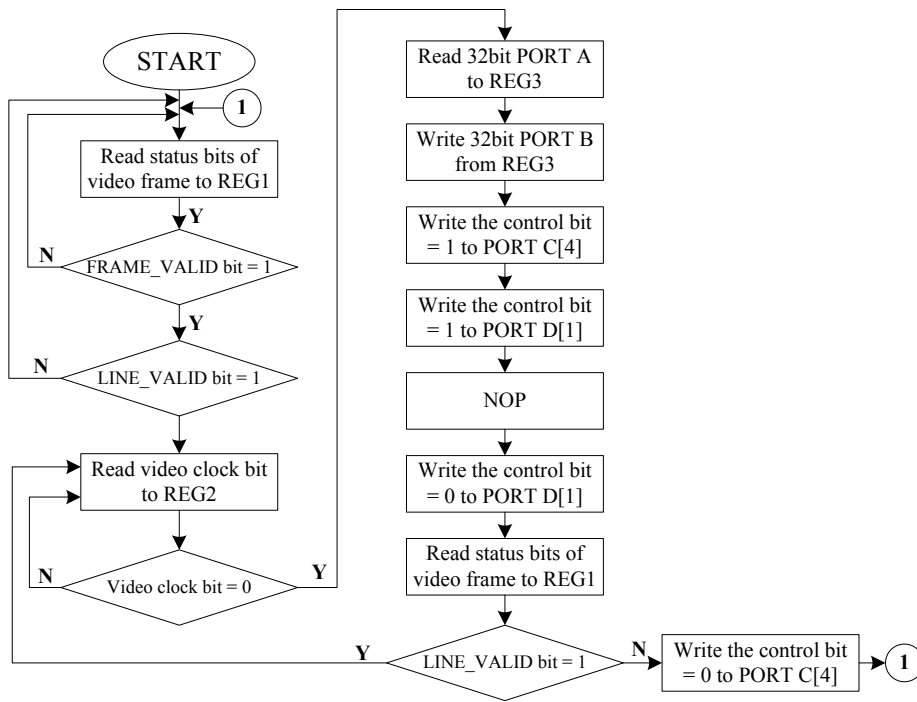


Figure 4.7: Microprocessor Based Implementation Flow Chart

In ideal conditions (no instruction or data hazards are introduced [38]) the number of instructions required to process a single pixel according to the presented flow chart equals to 14 instructions.

Figure 4.8 below illustrates two iterations of 14 instruction execution and associated latency and cycle time.

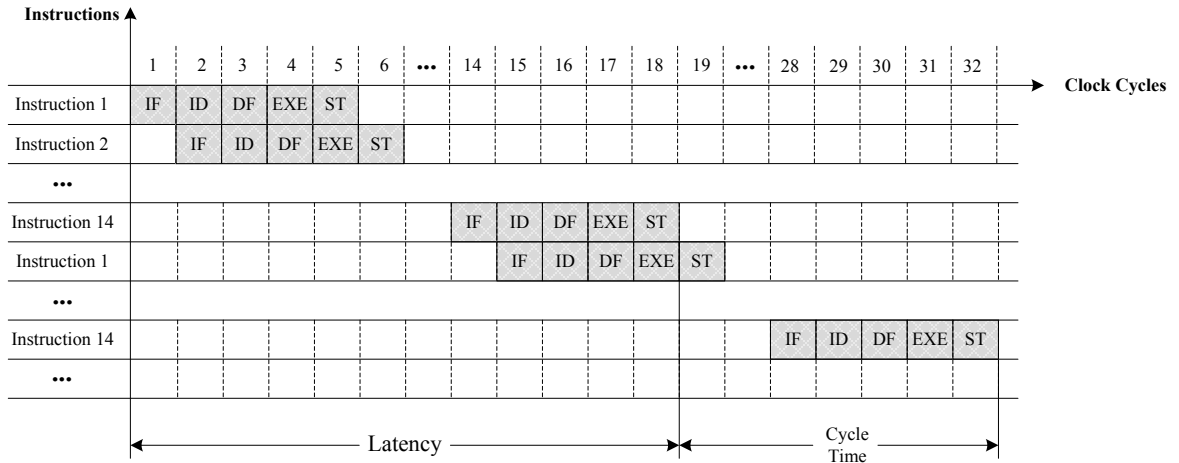


Figure 4.8: Microprocessor Instruction Execution Process

As it can be seen from the above figure, the number of clock cycles required for processing of first 32bit pixel word is equal to 18 clock cycles and is related to initial latency. However, the cycle time for subsequent pixels is equal to 14 clock cycles. That is due to instruction pipelining used in Microprocessors. As a result, by using Equation 4.2 the total execution time required for processing of a single row of pixels on Microprocessor ( $T_{exe}^{\mu P}$ ) is calculated as:

$$T_{exe}^{\mu P} = [18c. c. + (1024 - 1)c. c. * 14c. c. ] * 20.83ns = 298.75 \mu s$$

Based on the execution time results of both FPGA and Microprocessor the performance speedup of one solution over another is calculated as:

$$Speedup = \frac{T_{exe}^{\mu P}}{T_{exe}^{FPGA}} = \frac{298.75\mu s}{21.35\mu s} \approx 14 \text{ times}$$

In order to match the performance of FPGA, the Microprocessor has to operate at higher frequency to compensate the additional overhead associated with instruction based processing. The Equation 4.3 below computes the required Microprocessor frequency ( $F_{\mu P}$ ) based on task execution time associated with FPGA ( $T_{exe}^{FPGA}$ ) and number of clock cycles required for Microprocessor based implementation:

$$F_{\mu P} = \frac{1}{\frac{T_{exe}^{FPGA}}{[L + (n - 1) * C_{time}]}} = \frac{1}{\frac{21.35\mu s}{18c. c. + (1024 - 1)c. c. * 14c. c.}} = 672 \text{ MHz} \quad (4.3)$$

As it can be seen from all above performed calculations, due to highly parallelizable architecture of FPGA it is possible to achieve the required performance while operating at a much lower frequencies. This allows substantial reduction of power consumption as well as simplification an overall subsystem design and its associated cost.

Moreover, implementation of additional algorithms requires increase of operation frequency of Microprocessor on GHz levels. In such cases in addition to high power consumption requirements specialized cooling systems are required to remove an excessive heat from the device in order to prevent it from malfunctioning or damage.

Considering above analysis and accompanied calculation results, FPGA device from Lattice (LFXP6C) [43] is selected as a video pre-processor element. In addition to satisfying all processing requirements, LFXP6C provides the following features:

- Single 3.3V supply rail for the device logic core and all I/O Banks;
- Integrated on-chip non-volatile memory for configuration storage;
- On-chip configuration loader.

The selection of 3D-P Camera manager element for 4-Vision subsystem does not require performance analysis due to low speed nature of the performed operations. Instead, it requires I<sup>2</sup>C protocol support for image sensors configuration and sufficient number of I/Os to provide connectivity with subsystem's elements for debugging and control purposes. Therefore, out of a great number of existing microcontroller solutions meeting the above requirements the Microchip (PIC24FJ128GA010) [63] is selected for the subsystem implementation. Additional criteria considered during selection process are:

- Availability of licensed compiler and device libraries [64] for the selected device;
- Availability of MPLAB ICD 3 [63] device programmer for the selected device;
- The low CPR of the device.

The selection of PC-subsystem controller is based on single camera module video rate calculated in Section 4.2.1.1.1. The 22.5Mbps are taken as a reference due to have an ability of transmitting real-time video stream from subsystem to a PC for verification and testing purposes. In addition to that, a support of dual channel interface for connectivity with multiple subsystem elements is also considered. Therefore, to satisfy all the above requirements the FTDI (FT2232HL) USB controller device [65] was selected as a PC-subsystem controller element. The following list summarizes the main features of the selected device:

- Single channel synchronous FIFO mode transfers at 25MBps;
- Dual-channel asynchronous operations;
- Availability of device drivers for Windows operating system.

The selection of interfaces for the subsystem implementation is dictated by the bandwidth requirements between subsystem elements and by other pre-defined data exchange protocols that have to be followed to provide communication with the devices.

The 3D-P Camera interfacing requirements are specified by the camera module interface described in Section 4.2.1.1.2. Therefore, to provide compatibility with all camera modules, identical four 40-pin parallel ATA socket connectors are incorporated into the subsystem in total providing 77 signal lines.

To provide video transport interface with the existing MARS subsystem featuring VME 96 socket connector interface, the matching VME 96 terminal connector interface is selected for the 4-Vision subsystem implementation.

The configuration interfaces for Video Pre-Processor and 3D-P Camera manager elements are dictated by their specifications. For the first mentioned element JTAG configuration interface [53] is used, and for the second one ICSP interface, due to compatibility with the ICD 3 programmer device mentioned above.

Since the PC-subsystem controller, based on specification of FT2232HL is required to be differential USB interface, out of possible six standard connector interface types, Type B is selected due practicality in wider range of applications.

#### 4.2.1.2.2. Principles of the 4-Vision Subsystem Operation

Prior to the description the principles of subsystem operation, the block-diagram in Figure 4.5 is redrawn based on elements selection performed in the previous section. The resulted block-diagram is illustrated in Figure 4.9 below.

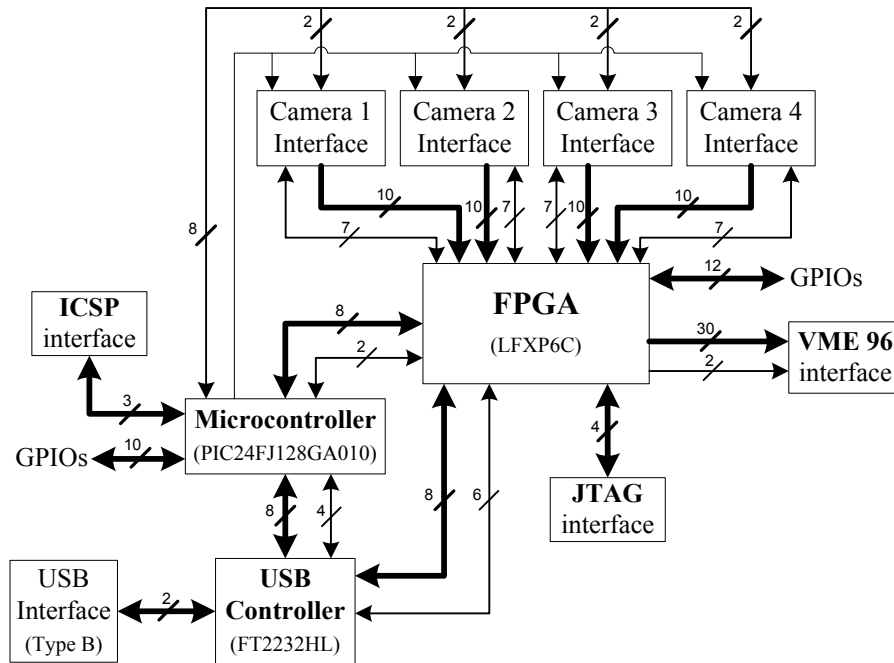


Figure 4.9: 4-Vision Hardware Subsystem Architecture

The explanations associated with the sizes of data busses (thick lines) and the sizes of control lines (thin lines) of particular connection are provided gradually throughout this section. The clock signals are specifically not included into the block-diagram because they required separate presentation in a form of clock distribution scheme performed at the end of this section.

As it was mentioned in the previous section, the selected FPGA features an internal non-volatile memory for configuration storage and an internal circuitry for self-configuration. At the first power ON of the subsystem, the FPGA's non-volatile memory, being factory erased, is loaded with the required configuration firmware (configuration bitstream), described in Section 4.2.1.3, via JTAG programming interface (signals FPGA\_TCK, FPGA\_TMS, FPGA\_TDI, FPGA\_TDO) from the PC running the required FPGA configuration application. When this stage is completed, FPGA's internal circuitry initiates self-configuration. During this process configuration bitstream from on-chip non-volatile memory is loaded directly to SRAM configuration memory, thus, realizing the required FPGA's logic circuit. In case of consequent power OFF/ON cycles, FPGA begin SRAM memory configuration bypassing configuration via JTAG interface.

Being successfully configured, the FPGA device is ready to perform all required operations. However, additional stages involving the 3D-P Camera and Microcontroller are performed prior to that.

The Microcontroller, being programmed with the software component via dedicated ICSP interface (PGD, PGC, MCLR) by using ICD 3 programmer attached to PC running the configuration application performs the following operations:

- Synchronous reset of all image sensors;
- Image sensors video settings configuration

On the power ON stage four image sensors start their operation on the different moments of time (i.e. the beginning of a new frame, indicated by the FRAME\_VALID signal, for all four sensor elements doesn't occur at the same moment of time). Such unsynchronized operation creates unnecessary processing delays, requires an extra logic resources to



compensate the required pixel-by-pixel synchronization and in overall introduces instability to the subsystem operation.

In order to avoid the above mentioned problems and to provide pixel-by-pixel synchronization, the simultaneous reset of all image sensors is performed by the Microcontroller. For that, the Microcontroller's signal line (CAM\_RST) is interconnected with RESET signal line of each camera module interface. On every power ON of the subsystem CAM\_RST line is set to logic LOW for duration of 2ms and then returned to logic HIGH state. As a result all image sensors simultaneously reset and return to operation at the same moment of time.

After the reset procedure is completed, image sensors video settings configuration takes place. For that, Microcontroller's four pairs of signal lines are interconnected with four camera module interface connectors. Every pair consisting of SCLK and SDATA lines provides I<sup>2</sup>C interface with image sensor internal registers. The software component responsible for registers configuration is described in Section 4.2.1.4 in details. Moreover, the required settings list accompanied with the descriptions is provided in a form of table.

When the 3D-P Camera initialization and configuration stages are completed, FPGA by means of firmware described in Section 4.2.1.3 performs video acquisition from four camera modules, combination of pixel and control data into the 32bit format and data transmission to MARS subsystem via VME 96 interface bus.

The VME 96 interface bus in addition to 32bit data is required to transfer control signal indicating start or stop of active frame window and the 48MHz clock associated with the transmission. For that, 34 signal lines interleaved with the ground lines, needed for crosstalk reduction, are required. However, due to limitation of VME bus introduced by MARS

subsystem, only 32 signal lines are available for transmission. Therefore, to overcome such signal lines deficit, two least significant color bits from two camera modules responsible for peripheral vision are not included to the transmission. These bits are considered to have a constant zero value during all image processing operations.

The USB controller based on its configuration settings provides either simultaneous asynchronous operation of both FPGA and Microcontroller, or synchronous operation only with the FPGA at 25MBps rates between the nodes. Depending on the particular verification and testing procedures either of modes can be selected at a time. In both cases data transmission is performed by 8bits accompanied by four main control signals. However, in case of synchronous transmission two additional signal lines one controlling direction of data bus (input or output) and the other one providing 60MHz operation clock are required.

In order to provide additional debugging, control and monitoring of subsystem operation during various execution stages, GPIOs from the FPGA side are connected to 8 LEDs and 4 Pushbuttons and GPIOs from the Microcontroller side are connected to 6 LEDs and 4 Pushbuttons. Appendix D contains the schematic and associated layout of the 4-Vision subsystem.

As it was mentioned at the beginning of this section, the clock distribution scheme requires individual consideration. Prior to discussion of the selected scheme the operational frequencies required for hardware elements as well as for communication interfaces are summarized in the following Table 4.2.

Table 4.2: 4-Vision Subsystem Operational Frequencies

<b>HW element / interface</b>	<b>Frequency(s), MHz</b>
3D-P Camera	48
FPGA	96, 48
VME 96 bus	48
Microcontroller	24

The master clock frequency required to provide operation to each camera module is equal to 48MHz. This means, that in order to perform operations on FPGA involving acquisition of data from the 3D-P Camera, the sampling frequency based on the Nyquist rate is required to be 96MHz. The other tasks performed on FPGA require 48MHz clock frequency not only for internal operations but also for the external ones. An example is the transmission of 32bit data on VME bus interface performed at 48MHz.

The lack of operation frequency specification for Microcontroller is explained by the types of operations it performs including reset, serial configuration, status monitoring, debugging etc. Therefore, to provide operation of Microcontroller the 24MHz frequency is selected out of maximum possible 32MHz associated with the device.

In order to provide operation of the subsystem based on the above specified frequencies the clock distribution scheme illustrated in Figure 4.10 is used.

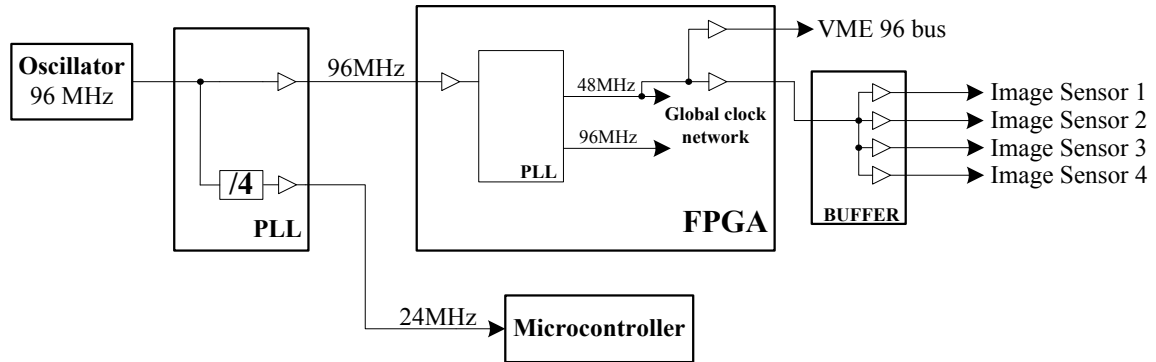


Figure 4.10: 4-Vision Clock Distribution Scheme

As it can be seen from the block diagram, the external oscillator generates frequency equal to 96MHz. Then, by means of clock divider PLL this frequency is buffered and applied to FPGA clock capable input, and at the same time is divided on 4 and applied to Microcontroller clock input. Internally, FPGA PLL unit synthesizes the frequency and produces 96 MHz and 48 MHz for the internal circuitry operation. In addition to that, 48MHz is applied to the external buffer in order to drive four image sensors and to VME 96 bus to provide synchronous operation with the MARS subsystem.

#### 4.2.1.3. Video Pre-Processor Firmware Component

The Video Pre-Processor firmware component (further – Pre-Processor component) is designed to capture video streams generated by four camera modules, combine them into one composite stream and transmit it through VME 96 interface to the MARS subsystem. The VHDL code listing of the Pre-Processor component is provided in Appendix C. The following two sections describe this component in details.

#### 4.2.1.3.1. Component Symbol

The Pre-Processor component has four identical control and data interfaces, one for each of the four camera modules it interacts with. In addition, it has VME interface consisting of control and data lines needed to perform the transmission to the MARS subsystem. Finally, the component has the clock input/output signals, and two general purpose utility ports. The top-level symbol of the Pre-Processor component is shown in Figure 4.11 below.

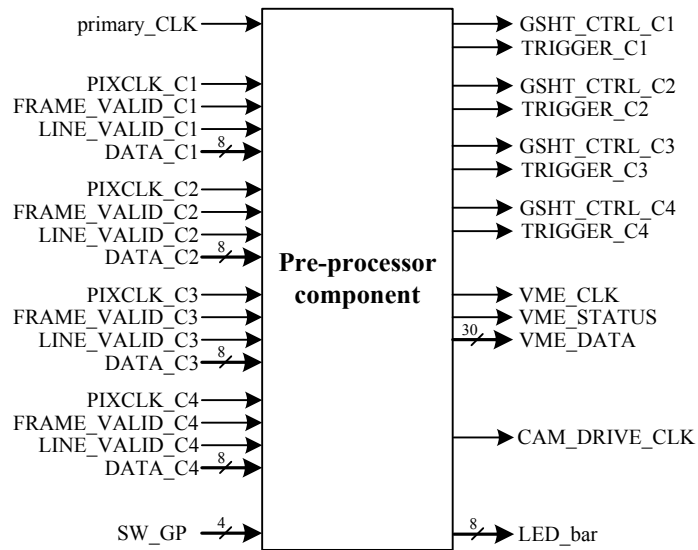


Figure 4.11: Pre-Processor Component Symbol

In accordance to the clock distribution scheme, the component receives external 96MHz clock on primary\_CLK input port. Internally, this clock is synthesized and outputted to CAM\_DRIVE\_CLK and to VME\_CLK ports. The four interfaces assigned to the four camera module interfaces, each organized as described in Section 4.2.1.1.2, consist of three types of signals: control inputs, data inputs and control outputs. The FRAME\_VALID\_Cx

and LINE\_VALID\_Cx receive validity information of a pixel byte received on DATA\_Cx ports synchronized with the clock received on PIXCLK\_Cx port. The GSHT\_CTRL\_Cx and TRIGGER\_Cx ports set the control over optional sensor's operations. The VME interface consists of 30bit data bus VME\_DATA port, synchronization VME\_CLK port and a valid data indicator VME\_STATUS port. Finally, SW\_GP and LED\_bar ports are used for debug and control purposes.

#### 4.2.1.3.2. Behavioral and Structural Description

The Pre-Processor component consists of two modules: acquisition and combination module and packet transmitter module. Figure 4.12 demonstrates interconnection between two modules.

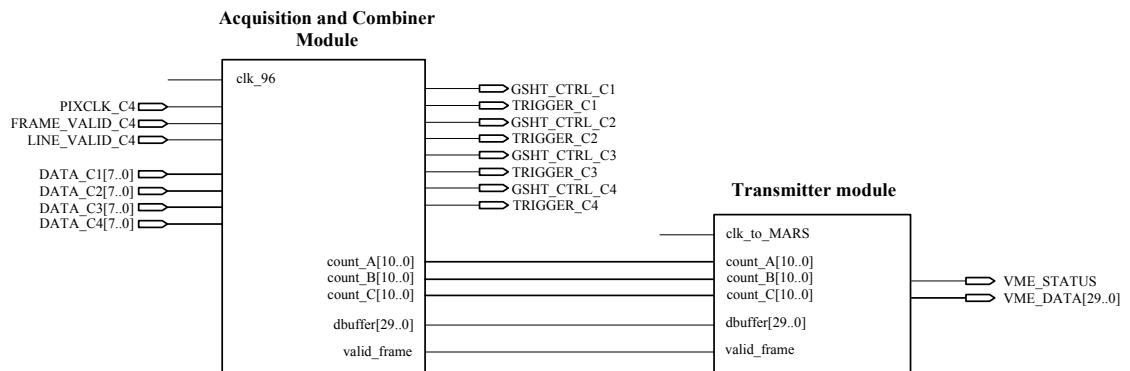


Figure 4.12: Pre-Processor Component Organization

Both modules operate in accordance to the clock distribution scheme described in Section 4.2.1.2.2. The acquisition and combiner module operates on 96MHz clock brought to the module via clk\_96 signal input. The transmitter module operates on 48MHz clock brought to the module via clk\_to\_MARS signal input.

The acquisition and combiner module monitors FRAME\_VALID\_C4 port and LINE\_VALID\_C4 port states on a falling edge of pixel clock received on PXLCLK\_C4 port and generated by the single camera module. The timing diagram of the camera module operation along with the details is provided in Section 4.2.1.1.2. When the active frame window is detected (FRAME\_VALID\_C4 = '1'), the internal flag signal valid\_frame is set logic HIGH. Prior to the first line valid signal, the vme\_status signals attached to the VME\_STATUS port is set to logic HIGH three clock cycles beforehand. That is done to accommodate the initial latency of SRAM buffers on MARS subsystem. When the first line of the frame is detected (LINE\_VALID\_C4 = '1'), the pixel byte from each of four camera modules is combined into a 30bit word and applied on dbuffer [29..0] register. As it was already mentioned in the previous sections, the least significant bits from two camera modules responsible for peripheral vision are discarded in order to accommodate the available 30 signal lines on the VME interface.

On the next clock cycle, the data from the dbuffer [29..0] register being connected to transmitter module is applied on VME\_DATA port and together with clk\_to\_mars signal attached to the VME\_CLK port is transmitted to the MARS subsystem.

During the acquisition and combination process count\_B [10..0] counter accumulates the amount of registered pixels words. After a 1024 pixel word is reached, the line counter

count\_C [10..0] is incremented by one and the count\_B [10..0] is reset. When the 768 line is reached the VME\_STATUS port signal is set to logic LOW and count\_C [10..0] is reset.

#### 4.2.1.4. 3D-P Camera Configuration Software Component

The 3D-P Camera configuration software component (further – 3D-P Camera configuration component) is used to perform reset of all image sensors and their internal registers configuration to the required video settings. Once these tasks are executed, the 3D-P Camera configuration component remains idle during system operation unless the request for the reconfiguration is triggered from the pushbutton or from a PC application.

##### 4.2.1.4.1. Component Symbol

The 3D-P Camera configuration component has four identical I<sup>2</sup>C interfaces consisting of clock (CLKx) and data (SDIx) ports. These ports provide interface with configuration registers of each image sensor. The 3D-P Camera reset (CAM\_RST) port is used to drive the reset of all image sensor settings to the factory defaults. The clock input port is used to drive the component's logic at 24MHz generated in accordance to the clock distribution scheme described in Section 4.2.1.2.2. Finally, SW\_GP port is used for the control purposes. The component symbol is show in Figure 4.13 below.



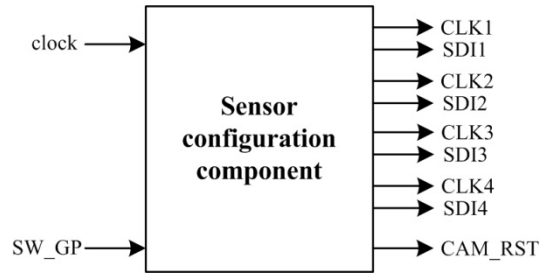


Figure 4.13: 3D-P Camera Configuration Component Symbol

#### 4.2.1.4.2. Behavioral Description

Prior to the beginning of operation with the 3D-P Camera, the 3D-P Camera configuration component performs reset of all image sensors. In order to do that, CAM\_RST port is set to logic LOW for the period of time equal to 3ms and then is returned to logic HIGH state. When the reset operation is completed, all four sensors begin synchronous operation based on their default settings. The next operation of the program cycle involves configuration of image sensors registers responsible for video setting parameters. This operation is performed by generating the clock and control signals on corresponding CLKx port and SDIx port accordingly. The communication protocol details as well as full register list can be found in the MT9T031 datasheet.

In order to provide operation of 3D-P Camera based on the video quality specifications defined in the beginning of Section 4.2, the register associated with resolution, vertical and horizontal blanking periods and other additional parameters are modified during the configuration process. The following Table 4.3 summarizes relevant registers, their addresses and the values to be programmed to obtain the required functionality.

Table 4.3: Image Sensor Configuration Registers

Register	Address (Hex)	Value (Hex)	Description
R3	0x03	0x05FF	Row size
R4	0x04	0x07FF	Column size
R5	0x05	0x011A	Horizontal blanking
R6	0x06	0x001C	Vertical blanking
R9	0x09	0x030F	Pixel row integration time
R34	0x22	0x0011	Skip 2x rows and bind
R35	0x23	0x0011	Skip 2x columns and bind
R53	0x35	0x002F	Global color gain control

The flow chart in Figure 4.14 illustrates the execution process of the main function of the 3D-P Camera configuration component.

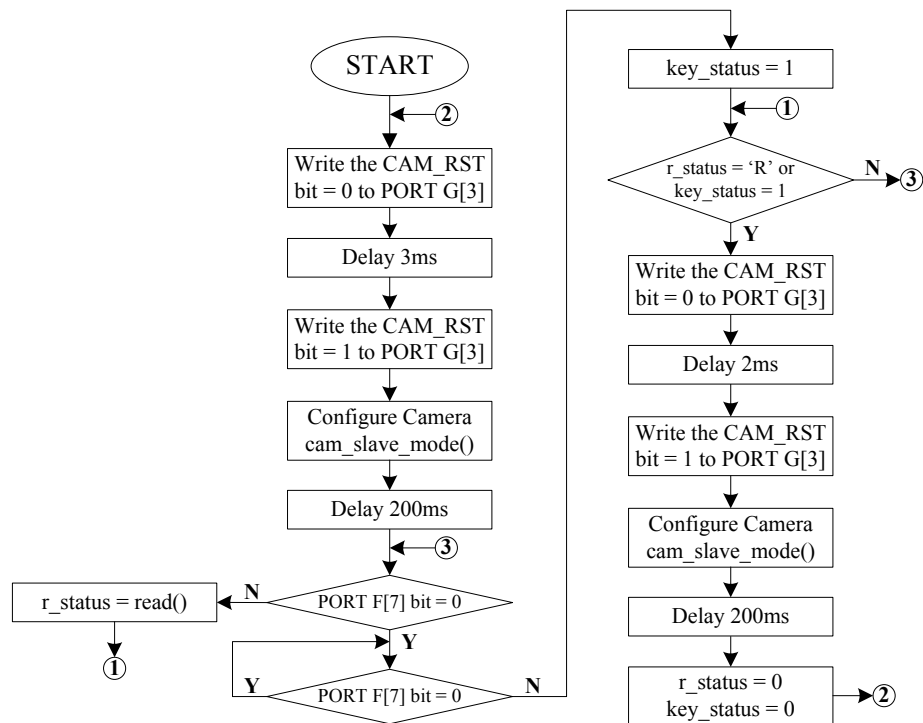


Figure 4.14: 3D-P Camera Configuration Component Operation Flow Chart

After power ON, the 3D-P Camera reset and configuration stages are executed. For the cases when it is necessary to perform 3D-P Camera reconfiguration with different parameters or it is simply required to perform iteration of 3D-P Camera initialization performed after power ON cycle, the pushbutton and PC controlled mechanisms are provided.

The pushbutton from one side is tied to 3.3V via 4.7KOhm pull-up resistor and connected with the port F[7]. From the other side it is attached to the ground. If the logic LOW is read from the port F[7] (pushbutton is pressed) the component performs second reading cycle of the port F[7] until logic HIGH is registered (pushbutton is released). Next, the logic HIGH is assign to `key_status` variable, thus, indicating the event of pushbutton activation. This triggers the 3D-P Camera reset and configuration mechanism.

The control from a PC application sending the control signal via USB interface is performed by retrieving value from the USB port using function `read()`. If the 'R' character is received, the component performs the 3D-P Camera reset and configuration operations.

The complete C code listing describing the 3D-P Camera configuration component is provided in Appendix A.

#### 4.2.2. MARS Subsystem

The MARS subsystem is a hardware platform that was developed before the architecture of 3D-P Telepresence System was designed. However, the amount of the processing resources as well as existence of wide range of interfaces made it possible to consider it for

the first implementation stage. As a result, all computationally intensive tasks associated with video stream processing and following displaying are performed on MARS subsystem.

The following section begins with description of general hardware architecture of MARS subsystem and focuses only on those hardware elements which are enabled for the application needs. Next to the hardware section, the video processor firmware component designed for MARS subsystem is described in details.

#### 4.2.2.1. MARS Subsystem Hardware Architecture

The MARS subsystem is responsible for reception of video data packets from 4-Vision subsystem, decomposition of data from each packet, application of color interpolation algorithm and output of the result to the Display. In order to fulfill all the above tasks the following hardware elements and interfaces incorporated to the MARS subsystem are selected:

##### **Elements:**

- Reconfigurable logic processor based on Xilinx FPGA (XCV4LX160) [40]
- Two memory banks based on Cypress SRAM (CY7C1462AV33) [66]
- Four digital to analog converters based on Analog Devices (ADV7125) [25]

##### **Interfaces:**

- Video transport interface based on VME 96 bus standard
- Four VGA interfaces (D-SUB)
- JTAG configuration interface for FPGA

In addition to the above elements and interfaces the LED indicators and mechanical actuators incorporated into the subsystem are selected for a testing and control purposes.

Following block-diagram in Figure 4.15 illustrates the interconnection of the selected hardware elements and interfaces. The explanations associated with the sizes of data busses (thick lines) and the sizes of control lines (thin lines) are provided in the next section.

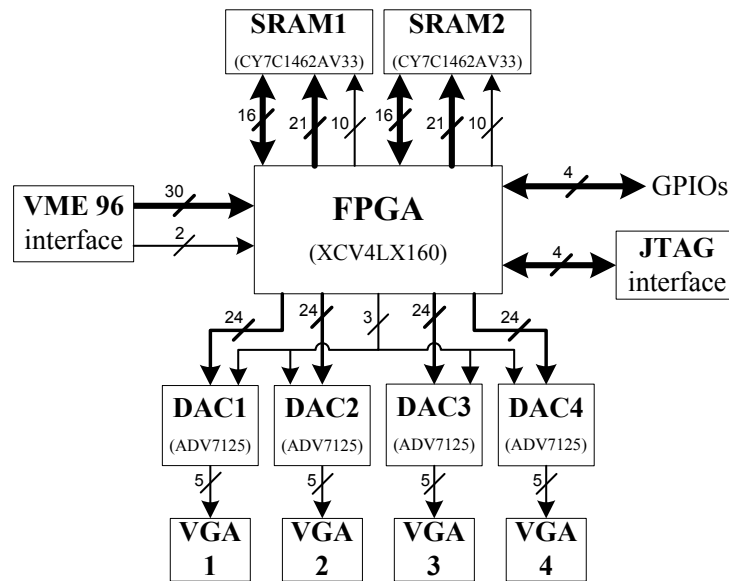


Figure 4.15: MARS Hardware Subsystem Architecture

The clock signals being included into the control lines require individual consideration. Figure 4.16 illustrates the clocking scheme including external and internal clocks implemented on MARS subsystem.

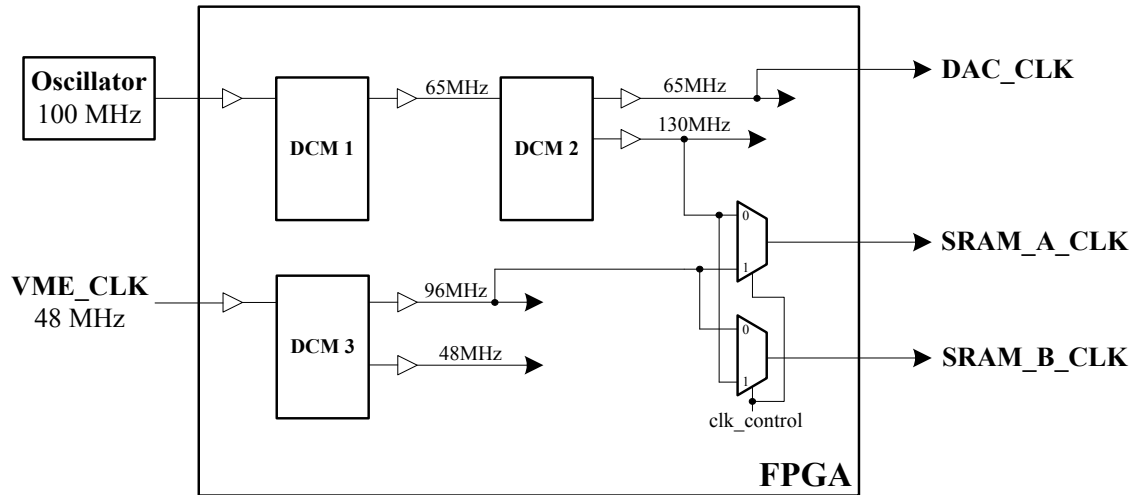


Figure 4.16: MARS Clock Distribution Scheme

As it can be seen from the above block-diagram, the MARS subsystem has five external clock sources out of which two sources are inputs and three sources are outputs, and it has six internal clock sources generated by the internal DCMs (Digital Clock Managers).

The following Table 4.4 summarizes all clock frequencies used by the subsystem. Signal names provided in the second column correspond to naming used in VHDL code (internal clocks).

Table 4.4: MARS Subsystem Operational Frequencies

Frequency, MHz	Signal name	Description
48	clk_48	VME data reception from 4-Vision subsystem
65	clk_65	Bayer Color Decoder; BRAM; DAC
96	clk_96	Sampling of VME signals; SRAM
100	clk_100_from_buff	Feedback for DCM 1
130	clk_130	BRAM; SRAM

#### 4.2.2.1.1. Principles of MARS Subsystem Operation

On every power ON of the subsystem the FPGA device is loaded with configuration firmware described in Section 4.2.2.2. For that, JTAG interface port of MARS subsystem is attached to the Xilinx Platform Cable USB programmer [40] which in turn is attached to the PC running the required FPGA configuration application. After the FPGA's SRAM configuration memory is programmed the subsystem is ready to perform operations defined by the logic circuit.

In order to initiate the reception of video data packets from 4-Vision subsystem and perform the required processing and displaying procedures, the pushbutton SW\_GP4 from pushbutton group SW\_GP[4..1] must be activated. The following operations on data performed by the subsystem are defined by the block-diagram in Figure 4.17.

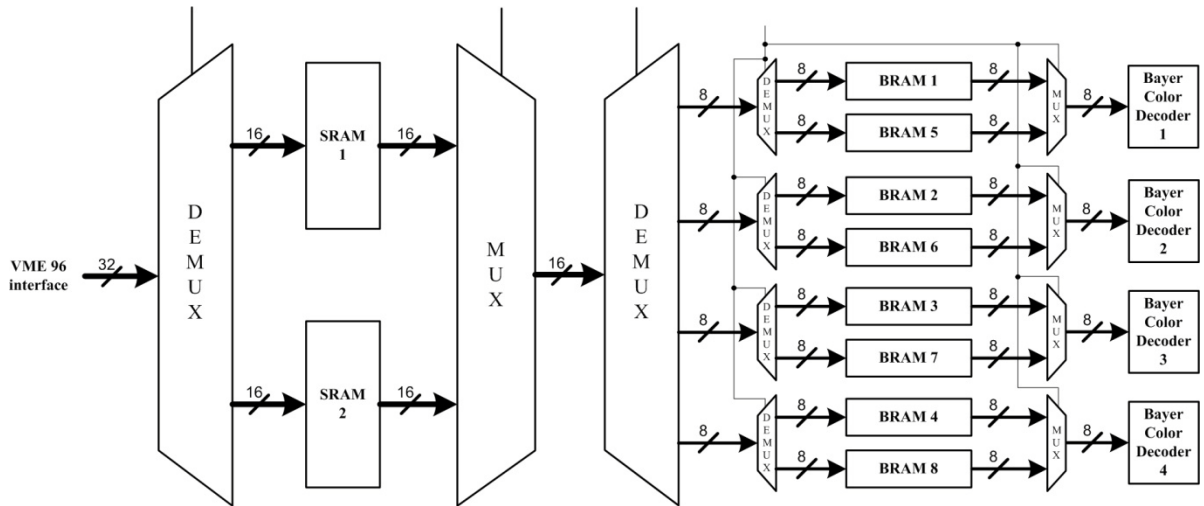


Figure 4.17: MARS Subsystem Data Flow

Being activated, the MARS subsystem receives the 30bit data accompanied by the control and 48MHz clock signals on VME 96 bus interface. The details of VME 96 interface organization on MARS subsystem can be found in VME 96 interface organization section of 4-Vision subsystem.

Based on the control signal, MARS subsystem initiates buffering of incoming data to one SRAM bank and at the same time performs reading and processing of previously buffered frame from the other SRAM bank. That is done to allow processing of a data at much higher rates than it is received on VME interface port.

Each of the two SRAM banks integrated into the MARS subsystem can store and address 16bit words at 2M locations. Therefore, when a 30bit data is received at 48MHz on VME interface, it is stored in the FPGA's internal register and then is written to two consecutive SRAM memory locations at 96MHz. As a result, the pixels associated with central left (CL) and right (CR) views and pixels associated with panoramic left (PL) and right (PR) views are stored in SRAM as it illustrated in Figure 4.18 below.

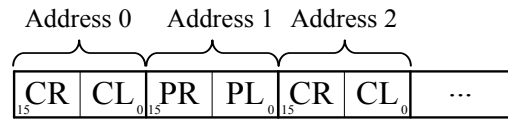


Figure 4.18: Storage of Pixels in SRAM Memory

Storage of four video frames in either of SRAM memory banks requires following amount of address locations ( $A_{space}$ ) and associated address lines ( $A_{res}$ ):



$$A_{\text{space}} = \frac{[Res_{\text{video}} * Res_{\text{pixel}} * N_{\text{frames}}]}{M_{\text{res}}} = \frac{(1024 * 768)\text{pixel} * 8\text{bit} * 4}{16\text{bit}} \approx 1.5\text{M}$$

$$A_{\text{res}} = \frac{\log A_{\text{space}}}{\log 2} = \frac{\log 1.5 * 10^6}{\log 2} = 20.52 \approx 21 \text{ line}$$

where  $Res_{\text{video}}$  is a video frame resolution,  $Res_{\text{pixel}}$  is a number of bits used to represent every pixel,  $N_{\text{frames}}$  is a number of video frames to be stored in memory,  $M_{\text{res}}$  is a memory data bus resolution.

After writing to either of SRAMs is completed, the internal control logic triggers a frame reading operation from the memory that was written. The reading process is performed at 130MHz. That is done to provide enough time for the processing and displaying algorithms operating at 65MHz to complete before the next frame is ready in the other SRAM.

Prior to the beginning of any pixel processing operations a data from the SRAM is classified in accordance to the video stream. For that, four pairs of BRAMs (Block RAM) each consisting of 1024 address locations each of which capable of storing 8bits are used. The odd pixel rows of four video streams are stored in BRAM 1, BRAM 2, BRAM 3 and BRAM 4. When 1024<sup>th</sup> pixel is reached, the control signal of four DEMUXs performs switching of data coming from the SRAM to BRAM 5, BRAM 6, BRAM 7 and BRAM 8 responsible for storing even pixel rows. As it can be seen from Figure 4.17, the DEMUX control signal is also attached to the four MUXs. These MUXs perform an output of buffered rows to the corresponding Bayer Color Decoders which perform color interpolation in accordance to algorithm described in Section 3.3.6.2.

After the 4 clock cycle delay RGB 24bit values corresponding to four video channels are outputted from the Bayer Color Decoders to the display driver. The display driver generates the control signals based on XGA timing specification illustrated in Figure 4.19 and together with RGB data sends them to four DACs driving four projectors of the Display, organized in accordance to Section 3.3.4, via VGA interfaces.

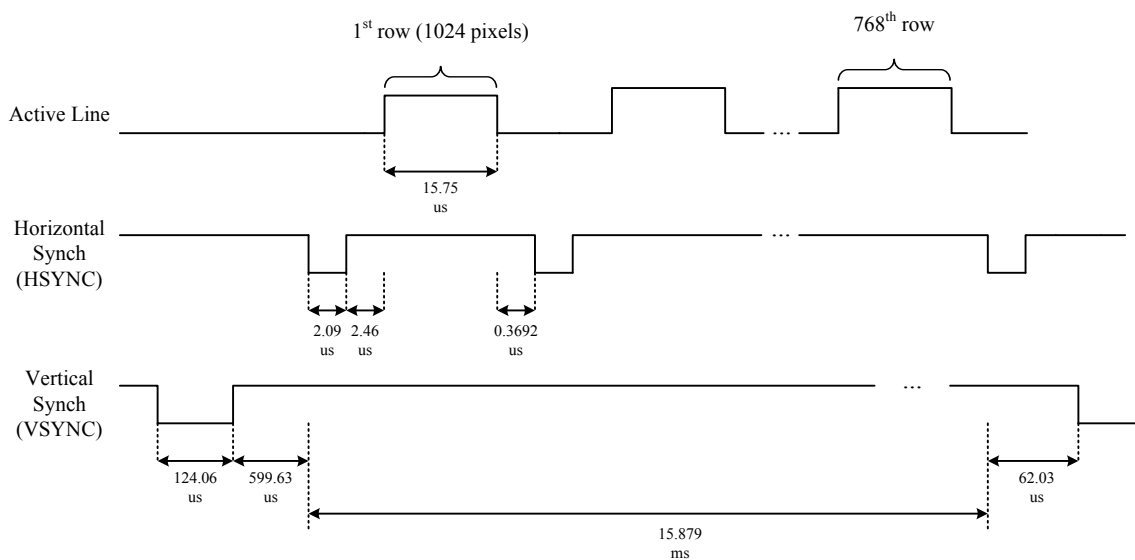


Figure 4.19: XGA Timing Specification

In spite of frame rate of 3D-P Camera equals to 30fps, an actual output rate is 60fps. That is due to minimum 60Hz output rate of projector units used for the display of video streams. As a result, the operations involving reading of the frames from the SRAM memory banks are repeated twice.

#### 4.2.2.2. Video Processor Firmware Component

The Video processor firmware component (further – processor component) is designed to receive the video data packets, perform decomposition of pixels in rows followed by frame buffering, perform color interpolation for all four video streams using Bayer Color Decoder subcomponent and output the results to the Display. The VHDL code listing of the Processor component is provided in Appendix B.

##### 4.2.2.2.1. Component Symbol

The processor component consists of three main interfaces. The first is the VME interface through which the data is transmitted from the 4-Vision subsystem. The second is the interface to two SRAM memory banks which are used for video frame buffering. The third interface consists of the four identical VGA interfaces used to output video streams to the Display. The top-level symbol of the processor component is shown in Figure 4.20 below.

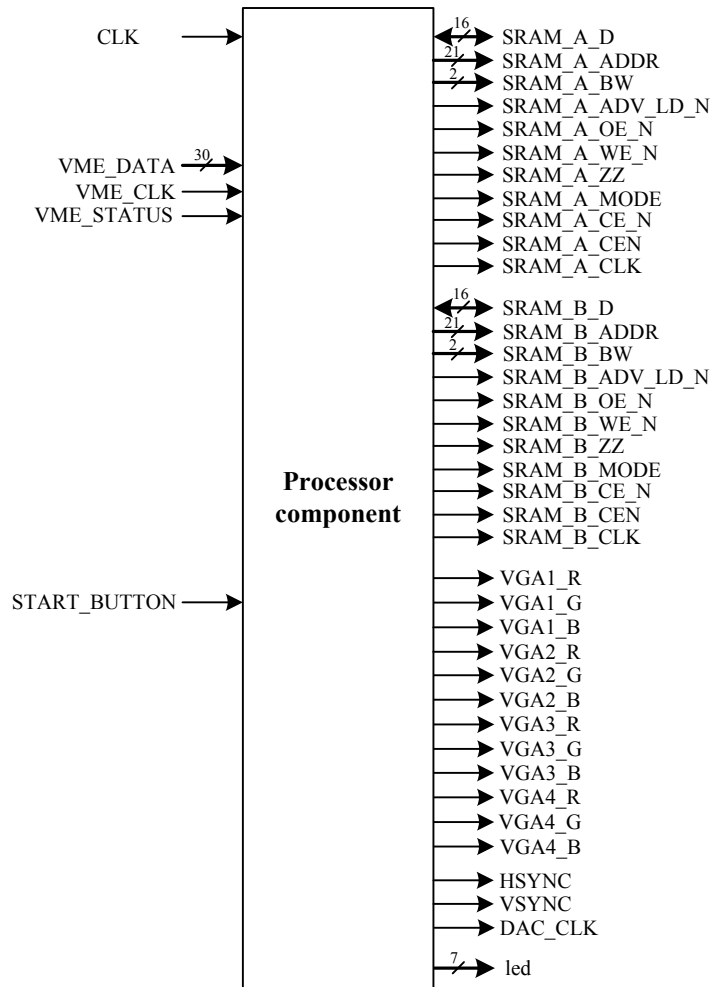


Figure 4.20: Processor Component Symbol

The component receives the external 100MHz clock on CLK input. The following clock distribution is performed in accordance to the scheme described in the previous section. The VME interface consists of the VME\_DATA 30bit port, the VME\_STATUS and VME\_CLK ports. The VME\_CLK port provides synchronization of data and control signals arriving on the VME interface. The VME\_STATUS port indicates validity of data on VME\_DATA port.

Both SRAM interfaces consist of 16bit data I/O port SRAM\_x\_D, the 21bit address port SRAM\_x\_ADDR , nine control ports and clock output port SRAM\_x\_CLK for synchronous operation. Further SRAM interface details can be obtained by consulting the CY7C1462AV33 datasheet.

Each of the four VGA interfaces consists of three 8bit color channel ports VGAX\_R, VGAX\_G and VGAX\_B. The DAC\_CLK port together with HSYNC and VSYNC ports are used to control the output of color pixels to the four video DACs in accordance to XGA timing illustrate in Figure 4.19.

Finally, the START\_BUTTON port is used for the subsystem activation, and led output port is used as an indicator of the subsystem state.

#### 4.2.2.2.2. Behavioral and Structural Description

The processor component consists of the Frame Buffering and Displaying subcomponent and Bayer Color Decoder subcomponent. Both subcomponents are constructed from a number of smaller modules illustrated in Figure 4.21. The functionality of Frame Buffering and Displaying subcomponent is defined by Modules 1-9. The Module 10 defines the functionality of four unified Bayer Color Decoder subcomponents combined in one block for illustration purposes. All the synchronously operating modules (1-4 and 7-10) receive clock signals on their inputs in accordance to the clock distribution scheme described in Section 4.2.2.1.



When the start\_buttn signal being tied to START\_BUTTON port changes its state from logic HIGH to LOW, the Module 2 initiates the reset of all control signals and internal registers. On the next clock cycle the capture\_done signal changes its state to LOW thus indicating the start of operations with data and control signals arriving on VME interface. When the transition from LOW to HIGH of Lattice\_status signal tied to VME\_STATUS port is detected, the valid\_frame signal flag indicating validity of VME data is set to HIGH. This triggers frame buffering operation into SRAM memory banks in accordance to the scheme presented in Section 4.2.2.1.1. A valid data from VME\_DATA\_BUS[29..0] register is written to SRAM\_x\_D[15..0] in two 96MHz clock cycles. The first 16bit word is written as is, however the second one, representing panoramic left and right vision is written being concatenated with two zero bits. That is done to compensate least significant pixel bits ignored due to VME I/O limitation.

When the first group of four frames is written to SRAM, the Module 2 changes the wr\_sram\_num control signal of Module 5 (DEMUX) in order to perform next frame group writing into the second SRAM bank. At the same time the Module 2 sets the step\_1 control signal to HIGH, thus initiating pixel distribution at 130MHz from the SRAM onto row BRAMs via bram\_in\_buffX[7..0] ports.

When the first 1024 pixel row of each of four frames is buffered into corresponding BRAMs, the bram\_we\_control signal switches DEMUXs to redirect the SRAM data to the second BRAM group dedicated for each video stream. As it is illustrate in Figure 4.17, with the switching of DEMUXs, the MUXs connected to the output bram\_x\_DOUT[7..0] ports of BRAM switch as well. This initiates reading and transfer of buffered rows at 65MHz to the input decX\_data[7..0] ports of four Bayer Color Decoders represented by the Module 10.

The control signal `bayer_decode_en` enables synchronous operation of all Bayer Color Decoders. The data received by each of four decoders is accumulated in the three internal BRAM buffers each capable of storing one row of 1024 color pixels. That is done to provide an access to the adjacent pixels during color components computations performed in accordance to the algorithm described in Section 3.3.6.2. The following block diagram in Figure 4.22 illustrates the Bayer Color Decoder organization.

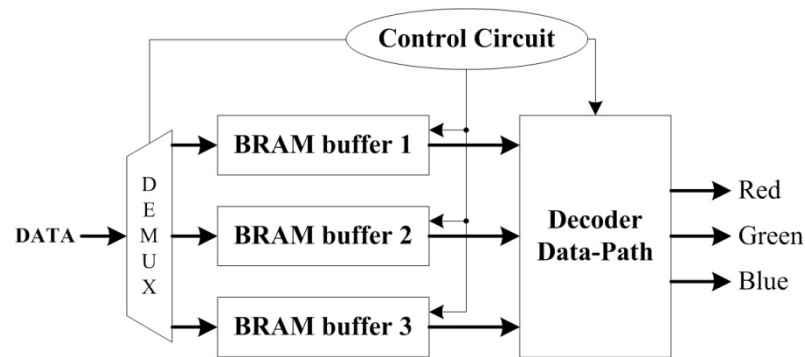


Figure 4.21: Bayer Color Decoder Organization

When the three buffers accumulate the required data, the control circuit initiates data transfer to the decoder data-path. As a result, after 5 clock cycle latency three color components for every pixel of a frame are outputted from the Red, Green and Blue ports of Bayer Color Decoders to the input `VGAx_byte_R[7..0]`, `VGAx_byte_G[7..0]` and `VGAx_byte_b[7..0]` ports of the Module 9.

The Module 9 is controlled via `vga_start` signal from the Module 2 and is responsible for output of stereo-panoramic video to four VGA channels in accordance to XGA video timing specification provided in Figure 4.19.



The active HIGH vga\_start signal is enabled prior to the first four 24bit RGB pixel data arrives to the inputs of Module 9. That is done to accommodate the data to the VGA timing specification. As a result, the RGB pixels from the Bayer Color Decoders are outputted to the VGAX\_R[7..0], VGAX\_G[7..0] and VGAX\_B[7..0] ports only during active line period defined by v\_data\_flag and h\_data\_flag signals and equal to 15.75us.

### 4.3. System Design Implementation (stage 2)

The second stage of 3D-P Telepresence System implementation follows up feasibility stage by introducing an advanced version of the system complemented with the functional elements that were not considered for the first implementation stage.

In accordance to the architecture organization specified in Chapter 3, the video-acquisition and pre-processing subsystem located on the slave side is required to perform stereo-panoramic video-acquisition, video pre-processing, video compression, transport stream formation and following transmission to the master side using transceiver equipment attached to the subsystem via transceiver interface. Realization of subsystem with the above described functionality requires utilization of the following key elements: four image sensors, four video compressors, video Pre-Processor, transport stream combiner/dispatcher, transceiver interface. The block-diagram in Figure 4.23 illustrates organization of the above elements within the subsystem.

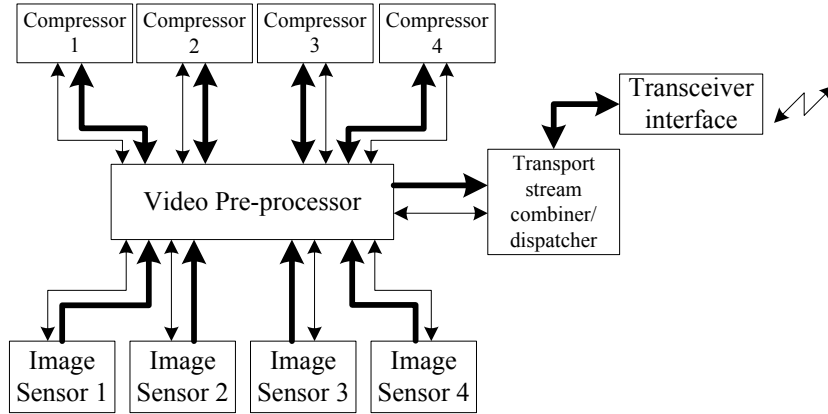


Figure 4.22: General Architecture of the Video-Acquisition and Pre-Processing Subsystem

On the master side, the video-stream processing and displaying subsystem is required to perform the reception of the transport stream, extraction of compressed video data from the packets, recording of compressed video to the storage memory for the future use. At the same time, perform video decompression, processing of the decompressed video and finally output the stereo-panoramic video stream to the Display described in Section 3.3.4. In order to realize such subsystem, the following key elements are employed: four decompressors, video processor, transport stream combiner/dispatcher, transceiver interface, storage memory and video transport interface. The block-diagram in Figure 4.24 illustrates organization of the elements within the video-stream processing and displaying subsystem.

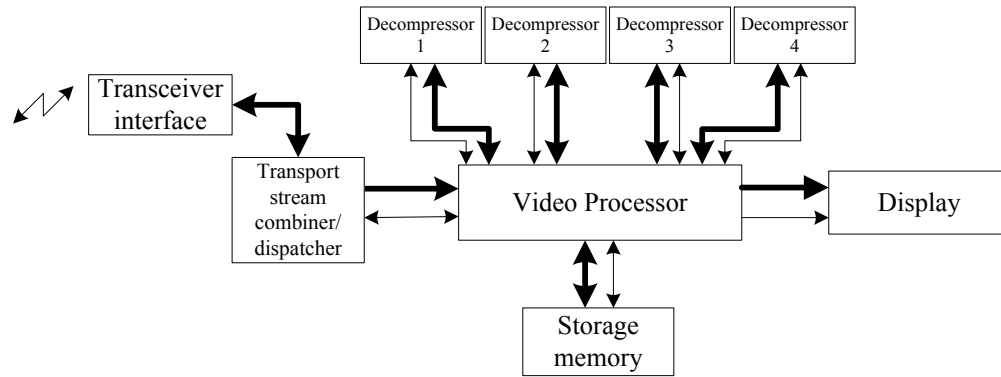


Figure 4.23: Architecture of the Video-Stream Processing and Displaying Subsystem

Above provided functional specifications for both subsystems are associated with the following technical specifications: quality of captured video for each video stream to be 1280x720 pixels (720p standard) at 30 fps and 8 bit per color channel; the data transmission rate between subsystems must not exceed 24Mbit/s (the specification of the transceiver equipment available for the project); live stereo-panoramic video recording up to 5 minutes; stereo-panoramic horizontal field of view to be close to that of a an average human vision providing about 50°-60° in stereoscopic region and about 120°-140° of an overall panorama; baseline to be equal to 68mm.

#### 4.3.1. Selection of the Proposed Elements for the System Implementation

The selection of elements for the second stage of 3D-P Telepresence System implementation is based on the elements analysis as well as results obtained after the first implementation stage.

By taking into account all advantages of reconfigurable logic devices in the considered class of applications, the functional elements such as video pre-processor and video processor are realized on Xilinx FPGA (XC3S1400AN) [40]. Similarly to the processor elements, the transport stream combiner/dispatcher used on both subsystems is realized on Xilinx FPGA (XC3S200AN) [40]. Both selected devices are from the same Spartan-3AN family featuring on-chip flash memory for configuration storage and on-chip configuration loader. The amount of the resources available on each device reflects the functional load of the element expressed by a number of system gates required for the logic circuit implementation as well as a number of I/O ports providing connectivity with an external infrastructure.

The selection of image sensor element is conditioned by the video quality requirements defined in technical specifications. In order to meet the requirements as well as take advantage of the results obtained on the first implementation stage the image sensor by Aptina (MT9T031) is selected for stereo-panoramic vision implementation. The details of image sensor operation are presented in Section 4.2.1.1.2.

By following the design solutions obtained at the first implementation stage the 3D-P Camera manager element performing the image sensors control and settings configuration is implemented using Microchip (PIC24FJ64GA104). Similarly, PC-subsystem controller based on FTDI (FT2232HL) is integrated to the system for verification and testing purposes.

Based on the overview of the existing video compression/decompression standards performed in Chapter 2 as well as analysis and performance comparison amongst the standards accomplished in Section 3.3.2 the JPEG2000 standard is selected for the system design architecture. Considering cost effectiveness of an existing JPEG2000 codecs as well

as their availability on the market at the time of writing this work, the compressor and decompressor IC by Analog Devices (ADV212) is selected for the system implementation. The ADV212 is a single-chip JPEG2000 compression and decompression solution. It incorporates an embedded 32-bit RISC processor which is used for configuration, control and management of the dedicated hardware functions. A single ADV212 can be either in Encode (compression) or Decode (decompression) mode depending on what firmware is loaded into the device. Further details of ADV212 operation can be obtained by consulting associated datasheet.

A part of specification for the project is the use of an external transceiver for communication between the subsystems. In order to provide an interface to RF transceiver, a serial differential link is needed. The Cypress HOTLink II transceiver (CYP15G0201DXB) was specified by the transceiver equipment provider [67], and therefore was incorporated into the system architecture. Further details of CYP15G0201DXB [66] can be obtained by consulting the corresponding datasheet.

In order to fulfill the requirements associated with recording and storage of 5 minute stereo-panoramic video, the two Micron's NAND flash memories (MT29F4G16AAC) each capable of storing 4Gbit are selected for the system implementation. The following calculation demonstrates a maximum possible recording length ( $T_{rec}$ ) of compressed video to both flash memory banks ( $M_{bank1}$  and  $M_{bank2}$ ) in accordance to video specifications and the compression ratio ( $C_{ratio}$ ):

$$T_{rec} = \frac{[M_{bank1} + M_{bank2}] * C_{ratio}}{[Res_{video} * Res_{pixel} * R_{frame} * N_{frames}] * 60sec} =$$

$$= \frac{(4096Mbit + 4096Mbit) * 40}{(1280 * 720)pixel * 8bit * 30fps * 4 * 60sec} \approx 6.47 \text{ min}$$

Similarly to the first implementation stage, to provide the system with the frame buffering capacity two Cypress SRAM (CY7C1460AV33) banks are incorporated into the system. Unlike SRAM banks used one MARS subsystem, CY7C1460AV33 provides 32bit data interface with ability of storage information at 1M locations.

The fact that the selected key functional elements such as video pre-processor, video processor, compressor/decompressor, transport stream combiner/dispatcher and transceiver interface operate in accordance to the configuration firmware they are loaded with influences the architecture organization of the complete system. As a result, two hardware platforms are developed. One of these platforms is designed to be unified, so that the elements on it can perform the operations on slave side and the reverse operations on the master side. The other platform is designed to perform operations on slave side in conjunction with the unified platform. Such an architectural decision is based on the cost of PCB development and manufacturing as well as on the cost of elements required for the implementation.

The following section describes the system hardware architecture organization consisting of the unified platform (which is referred to as 4V-J2C platform) and the video-acquisition and pre-processing platform (which is referred as 4-Vision2 platform).

### 4.3.2. System Hardware Architecture

The 3D-P Telepresence System hardware architecture consists of 4-Vision2 and 4V-J2C subsystems. The associated schematic and layout are attached in Appendix E and Appendix F accordingly.

The 4-Vision subsystem is responsible for acquisition of stereo-panoramic video from four video streams, configuration/control of image sensors, video pre-processing and transmission of data to the 4V-J2C platform.

The implementation of 4-Vision2 requires use of the following main hardware elements and interfaces:

#### **Elements:**

- Four image sensors based on MT9T031
- Video Pre-Processor based on XC3S1400AN
- 3D-P Camera manager based on PIC24FJ64GA104
- Two memory banks based on CY7C1460AV33

#### **Interfaces:**

- Two cross-platform interfaces based on SAMTEC FMC (ASP-134603-01, ASP-134604-01) [60]
- Two 40-pin camera module interfaces
- JTAG configuration interface for FPGA
- ICSP configuration interface for Microcontroller

In addition to the above elements and interfaces the LED indicators and mechanical actuators are incorporated into the platform for testing and control purposes. The block-diagram of 4-Vision platform hardware architecture is illustrated in Figure 4.25 below.

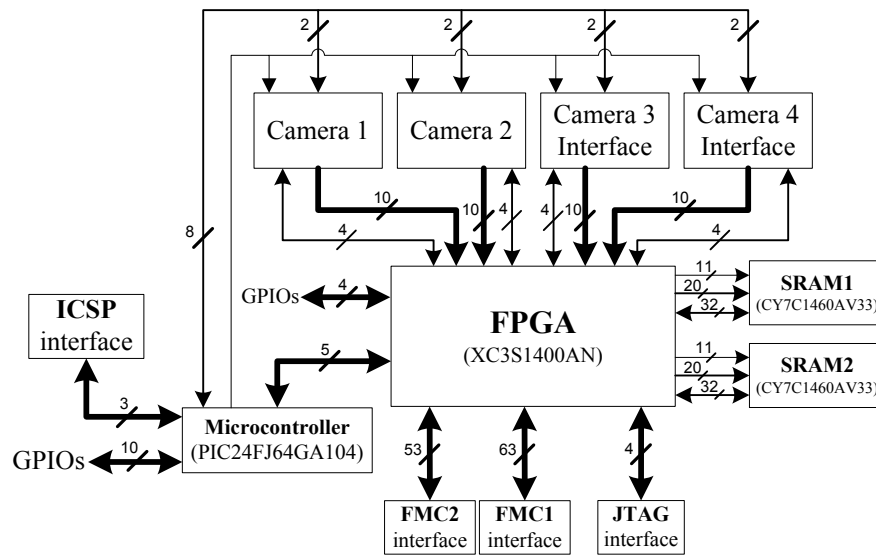


Figure 4.24: 4-Vision2 Hardware Architecture

Depending on the firmware configuration the 4V-J2C subsystem performs operations either on the slave side or on the master side.

In case of slave side configuration, 4V-J2C is responsible for reception of data via high speed FMC connector from the 4-Vision2 subsystem, compression of each of four video streams using dedicated compressors, encapsulation of the compressed data into the payload of the packet (in accordance to the Section 3.3.5) and together with header transmission of packet in a form of a transport stream to the master side via transceiver interface attached to RF transceiver equipment.



In case of master side configuration, 4V-J2C is responsible for reception of a transport stream via RF transceiver equipment attached to the transceiver interface, packet header analysis followed by extraction of the payload, recording of the compressed video data to the storage memory, decompression of video data using dedicated decompressors, video processing and output of stereo-panoramic video data to the high speed FMC connector.

The implementation of 4V-J2C subsystem requires use of the following main hardware elements and interfaces:

**Elements:**

- Video processor based on XC3S1400AN
- Four JPEG2000 video codecs based on ADV212
- Transport stream combiner/dispatcher based on XC3S200AN
- Transceiver interface based on CYP15G0201DXB
- Two flash memory banks based on MT29F4G16AAC
- PC-subsystem controller based on FT2232HL

**Interfaces:**

- Cross-platform interface based on FMC ASP-134603-01
- USB Type-B interface
- BNC interface
- JTAG configuration interface for FPGA
- ICSP configuration interface for Microcontroller

Similarly to the 4-Vision2 subsystem, the mechanical actuators are incorporated into the platform for testing and control purposes.

The block-diagram of 4V-J2C hardware architecture is illustrated in Figure 4.26 below.

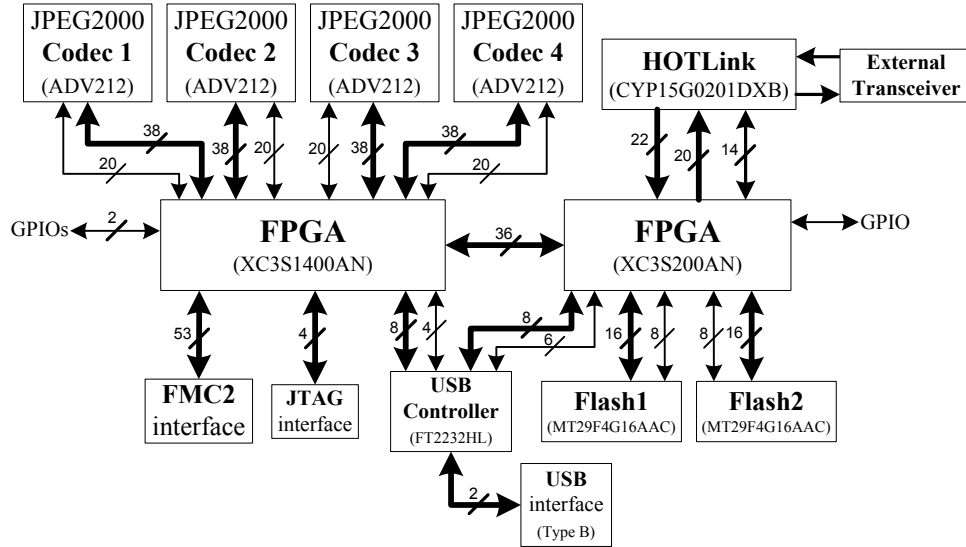


Figure 4.25: 4V-J2C Hardware Architecture

#### 4.4. Conclusion

This Chapter described two versions of the 3D-Panoramic Telepresence System platform organized in accordance to the architecture defined in Chapter 3.

The first implementation stage was oriented for feasibility testing of the proposed platform. During this stage the set of hardware, firmware and software was developed.

As the result, the first live prototype was designed, manufactured and tested. Then, this prototype was used in the experimental setup described in Chapter 5. On the basis of the results gained on the first prototype, the recommendations for modification of the system platform were determined. According to these recommendations, the second version of the platform was designed and prototyped.

## 5. EXPERIMENTS AND RESULTS

### 5.1. Introduction

This chapter presents the analysis of the experimental results gained on the developed 3D-P Telepresence System. The main objective for the experiments was to collect and analyze data of the following performance parameters: start-up time, frame transfer time, quality of images in different compression rates and power consumption for the system in different modes. The following sections will address the above objectives starting with a detailed description of the experimental setup and the associated instrumentation (external and on-chip).

### 5.2. Experimental Setup

In order to perform the data collection needed for the analysis of the performance parameters of the 3D-P Telepresence System, the special experimental setup was created.

The experimental setup consisted of the operational hardware and the configuration /verification tools including the external (to the system) and the internal (on-chip) instrumentation. The block-diagram of the experimental setup is shown on Figure 5.1.

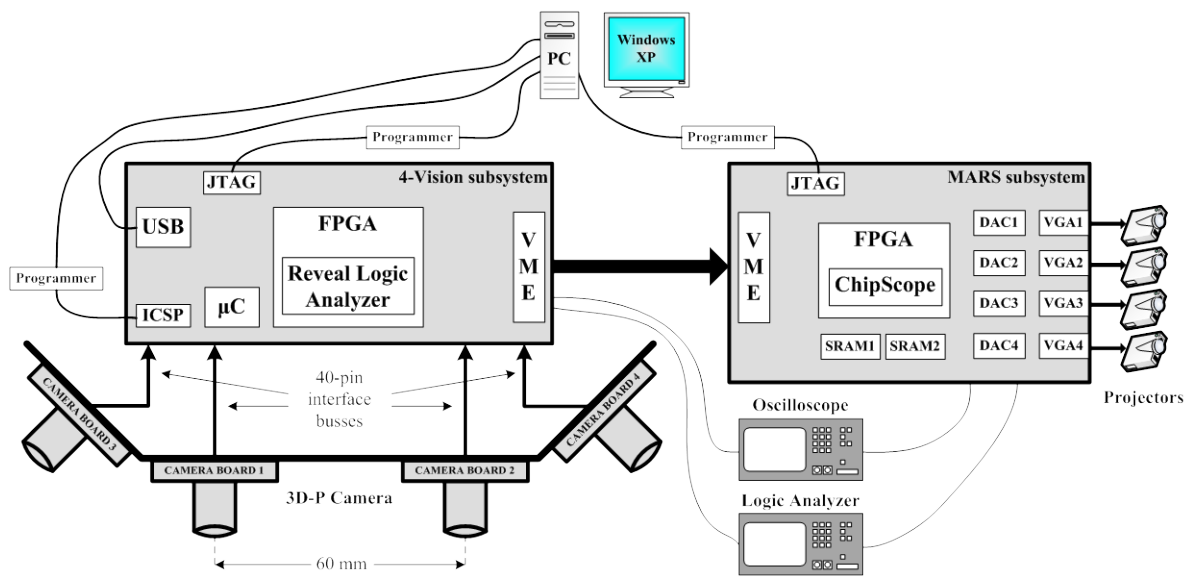


Figure 5.1: Experimental Setup of 3D-Panoramic Telepresence System

The image of the setup is shown in the following Figure 5.2 below.

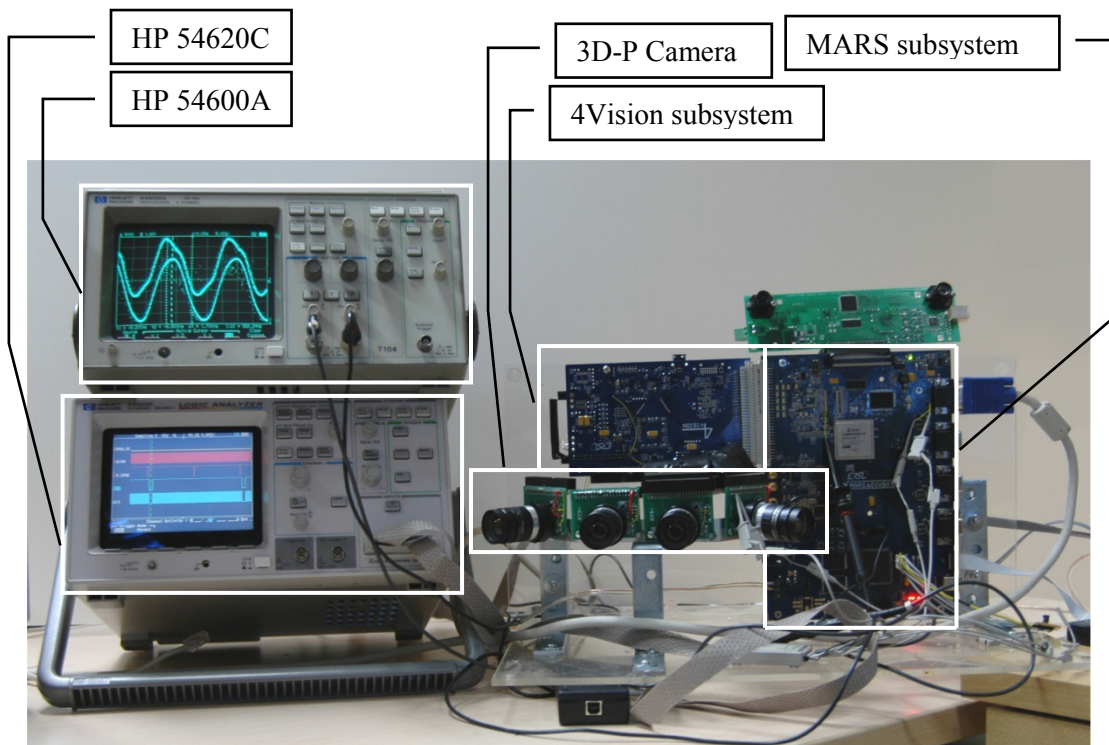


Figure 5.2: Image of the Experimental Setup

The operational hardware consists of the 3D-P Camera, 4-Vision and MARS subsystems described in detail in Section 4.2. Figure 5.3 illustrates the image of the 4-Vision subsystem.

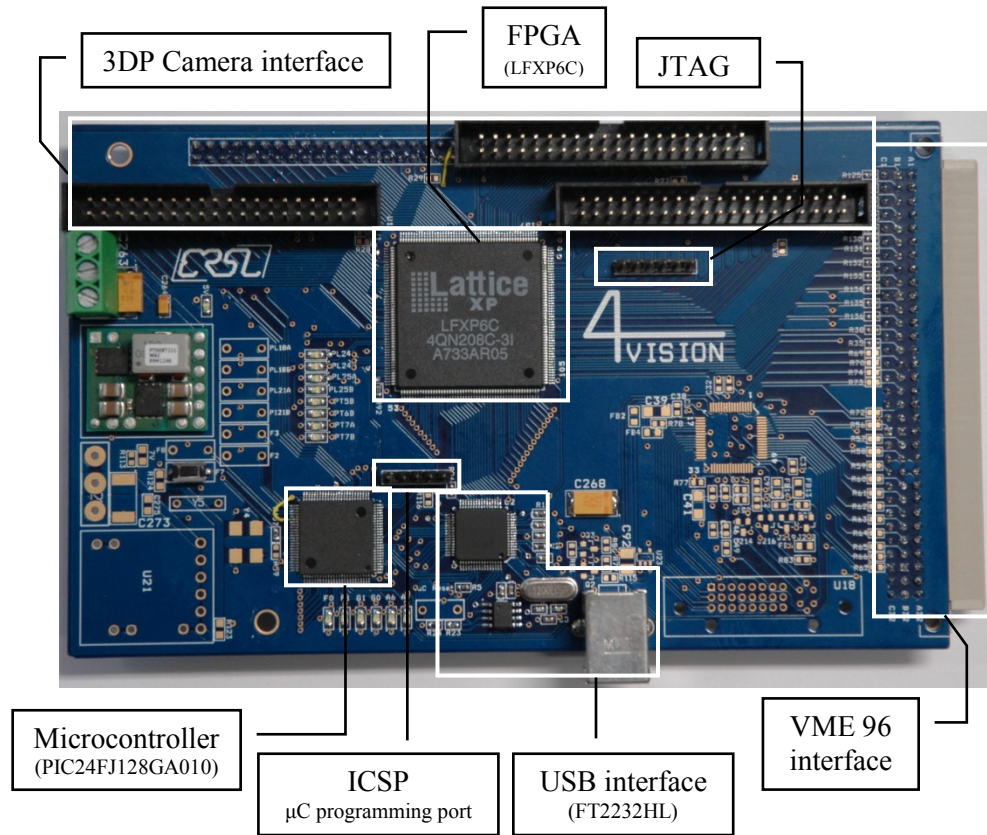


Figure 5.3: Image of the 4-Vision Subsystem

The 4-Vision subsystem incorporates the following hardware (according to the description in Section 4.2.1):

1. Video pre-processor based on Lattice LFXP6C FPGA. This element performs the following functions: i) Synchronous acquisition of four video frames from all camera modules of the 3D-P Camera; ii) Generation of synchronization signals for 3D-P Camera; iii) Combination of the transport stream to be transferred to the MARS

subsystem; iv) Generation of the synchronization signals for the VME 96 communication bus.

2. 3D-P Camera manager based on the Microchip PIC24FJ128GA010 RISC microcontroller. It provides configuration and re-configuration of the 3D-P Camera and therefore, adaptation for different modes of operation.
3. System interfaces consisting of the following: i) USB interface based on FTDI FT2232HL USB controller. It allows the instrumentation PC exchange of configuration data and commands with the 3D-P Camera manager (via low bandwidth asynchronous channel) as well as allowing exchange of video data and control signals with pre-processor (via high bandwidth synchronous channel); ii) ICSP interface port used for the configuration data loading to the 3D-P Camera manager; iii) 3D-P Camera interface ports used for data and control signal exchange between four camera modules and the pre-processor; iv) VME 96 interface port used to transmit the video transport stream to the MARS subsystem; v) JTAG interface used for pre-processor configuration firmware loading as well as for verification and debugging purposes.

The maximum on-board clock frequency is 96MHz which is distributed within the subsystem in accordance to the clock distribution scheme described in Section 4.2.1.2.2.

The image of the MARS subsystem is illustrated in Figure 5.4 below.

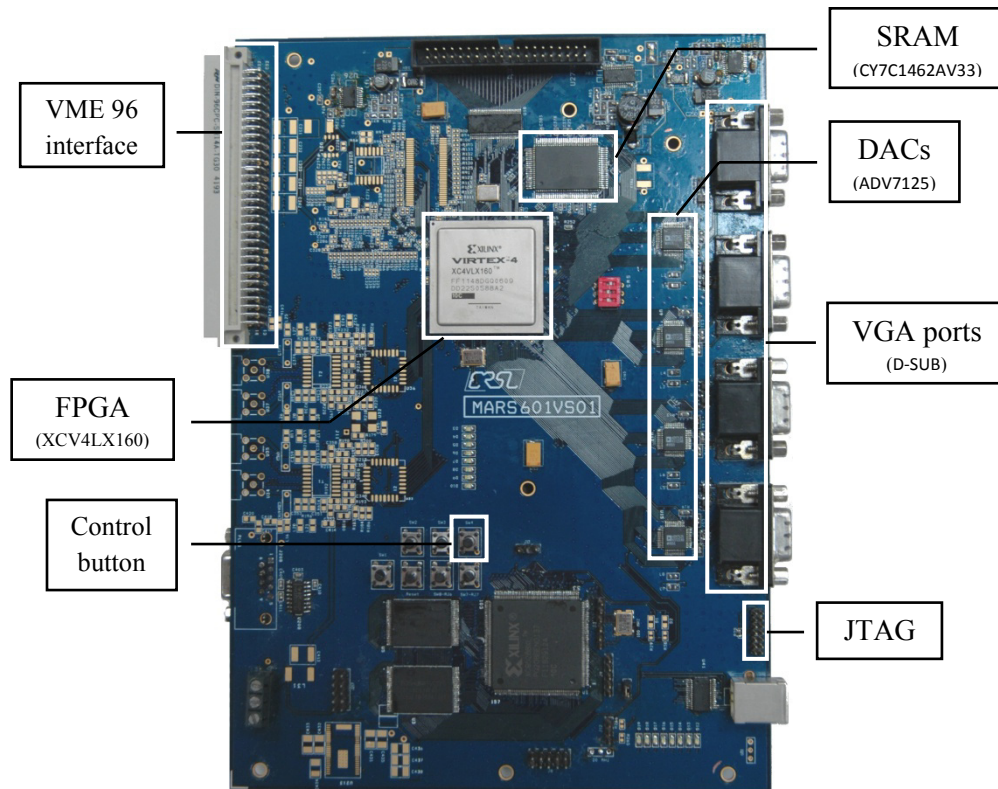


Figure 5.4: Image of the MARS Subsystem

The highlighted elements are those which are enabled for the project's needs. This includes the following (according to description in Section 4.2.2):

1. Video processor based on Xilinx XCV4LX160 FPGA. This element performs the following functions: i) Synchronous reception of the video transport stream from the 4-Vision subsystem; ii) Color interpolation for four real-time video streams iii) Control signal generation for real-time 3D-Panoramic video output.
2. Two memory banks based on Cypress CY7C1462AV33 SRAM used for the video frame buffering.

3. Four DACs (Digital-to-Analog Converters) based on Analog Devices ADV7125.

These elements perform conversion of digital to analog signals required for the four VGA interfaces.

4. System interfaces consisting of the following: i) VME 96 interface port used to receive the video transport stream from the 4-Vision subsystem; ii) Four VGA D-SUB interfaces used for video streams output to four video projectors; iii) JTAG interface used for configuration, verification and debugging purposes for the video processor.

The maximum on-board clock frequency equals 130MHz and is generated in accordance to the clock distribution scheme described in Section 4.2.2.1.

The system configuration/verification tools consist of the following:

**Programmers:**

- Xilinx Platform Cable USB
- Lattice ispDOWNLOAD Cable
- Microchip MPLAB ICD 3

**Software:**

- Xilinx ISE 9.2i [40]
- Xilinx ChipScope Pro 9.2i [40]
- Lattice ispLEVER Starter 7.2 [43]
- Microchip MPLAB IDE 8.5 [63]

**Equipment:**

- HP 54620C Logic Analyzer



- IBM PC

All the programmers specified in the above list provide the configuration and verification interface between the PC running the corresponding software application and the hardware system.

The configuration of Lattice FPGA is performed using the ispDOWNLOAD Cable attached to the Parallel port of a PC running the ispVM System 7.2 application and to the JTAG port of the 4-Vision subsystem. The verification of Lattice FPGA is performed using the same connectivity scheme and the Reveal Logic Analyzer 7.2 tool. Both of these applications are the part of the ispLEVER Starter 7.2 software suite.

The configuration of Xilinx FPGA is performed using the Xilinx Platform Cable USB programmer attached to the USB port of a PC running iMPACT application and to the JTAG port of the MARS subsystem. The verification of Xilinx FPGA is performed using the same connectivity scheme and the Xilinx ChipScope Pro 9.2i tool.

The configuration of the microcontroller is performed using Microchip MPLAB ICD 3 programmer attached to the USB port of the PC running MPLAB ICD 8.5 application and to the ICSP port of the 4-Vision subsystem.

As already mentioned in Section 4.2.1.2, once being programmed, the Lattice FPGA and the microcontroller don't require reconfiguration on the subsequent power ON/OFF cycles. Therefore, on every power ON of the system, the only device which is required to be programmed from the PC is the Xilinx FPGA.

### 5.3. Experimental Results Analysis

#### 5.3.1. Startup Time Analysis

The experiments performed in this section are aimed for the initiation time determination of the 3D-P Telepresence System. This is due to the importance of the initial reaction time of the system in many time critical applications.

The experiments were conducted on both: 4-Vision and MARS subsystems. On power up of the system from the dual-channel 5.5V DC power supply, the on-board DC-DC converter on 4-Vision and MARS subsystems stabilize their maximum 3.3V rail output in 17.5ms and 20ms accordingly. These timings are shown on the images in Figure 5.5 which were taken from the Oscilloscope HP-54600A.

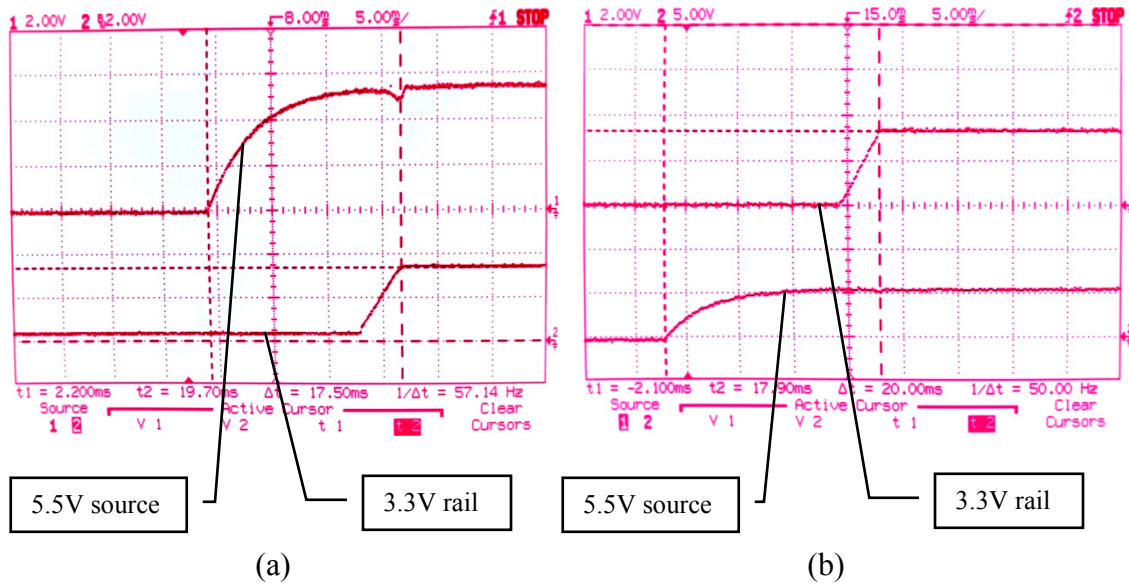


Figure 5.5: DC-DC Converters Startup Time: (a) 4-Vision Subsystem, (b) MARS Subsystem

By reaching the sufficient voltage level, the Lattice FPGA and the Microcontroller perform self-configuration. Overall this takes 20.1ms and 16.7ms accordingly. Following self-configuration, the Microcontroller by means of 3D-P Camera configuration software component described in Section 4.2.1.4 performs a synchronous reset of all camera modules and a video settings configuration.

Execution of both of these operations including the time it takes 3D-P Camera to provide a first frame valid HIGH signal takes 3.241s on a first power up cycle and 240ms on a consequent reconfiguration cycles. Being configured all camera module operate synchronously and output pixel bytes and associated frame and line valid signals in respect to the 48MHz pixel clock. The image of the Logic Analyzer HP-5620C readings corresponding to the 3D-P Camera operation after the configuration cycle is shown on the Figure 5.6 below.

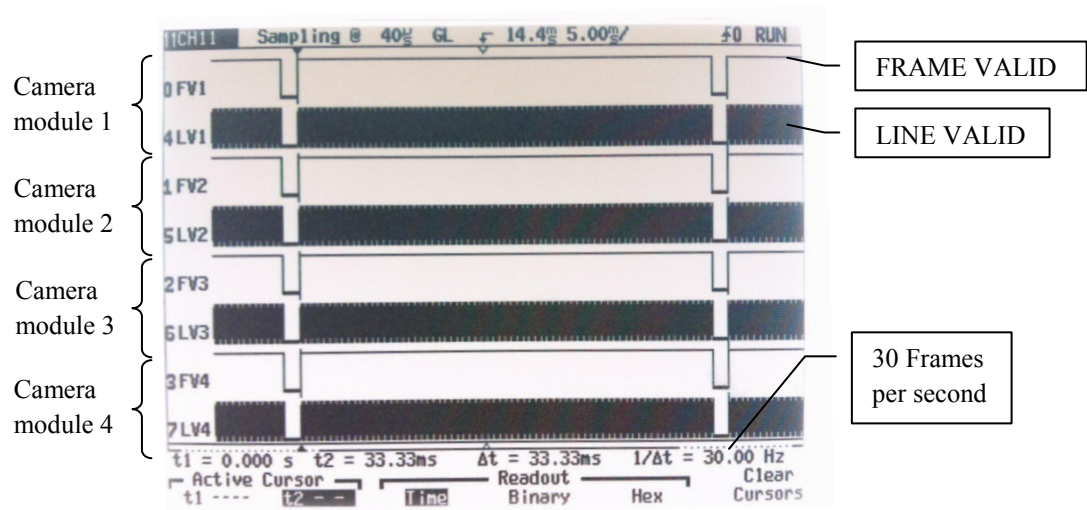


Figure 5.6: 3D-P Camera Synchronous Operation

Additional images of Logic Analyzer and Oscilloscope readings corresponding to all above provided timing measurements are attached in Appendix D.

Based on the JTAG configuration clock set to 6MHz in the iMPACT application, the complete configuration procedure of Xilinx FPGA takes  $\sim 7$ s. However, in the case of SelectMAP (32-bit) configuration, Xilinx FPGA requires only 52ms for a complete configuration cycle. The following Table 5.1 summarizes the startup timing information of the system.

Table 5.1: Startup Timing Summary

<b>HW element</b>	<b>Startup time (ms) 4Vision</b>	<b>Startup time (ms) MARS</b>
DC-DC converter	17.5	20
FPGA configuration	20.1	52
Microcontroller	16.7	—

The above experimental results demonstrated that the maximum start-up time for 4-Vision subsystem does not exceed 20.1 ms, whereas the start-up time for the MARS subsystem can be reduced to 52 ms. By taking into account that the start-up processes in the 4-Vision and MARS subsystems are occurring parallel, it is possible to determine that the 3D-P Telepresence System can start operations after 52 ms from the starting point. For the requested frame rate equal to 30fps, it is less than a period of two video frames ( $2 \times 33.33\text{ms} = 66.66\text{ms}$ ). However, the 3D-P Camera configuration brings a major delay equal to 3.241s on the first power up cycle and 240 ms for consequent re-configuration cycles. Therefore, to reduce the reaction time of the system, it is possible to keep the 3D-P Camera in “sleep”

mode when the rest of the system is powered off. In this case, the power consumption of 3D-P Camera will be minimal and therefore can keep the configuration data in cameras using a back-up battery solution. This recommendation will be implemented in the next version of 3D-P Camera.

### 5.3.2. Video Acquisition and Packet Transfer Time Analysis

The experiments performed in this section are aimed for the determination of the actual timing characteristics of the 4-Vision subsystem. In particular, timing associated with video pre-processing operations performed in a pipeline manner and consisting of latency and cycle time were the major parameters in these experiments.

The timing characteristics were obtained from the 4-Vision subsystem using Reveal Logic Analyzer tool. The diagram screenshot in Figure 5.7, demonstrates the initial stage of pixel data acquisition from four camera modules, a combination of data into 30-bit packets and (together with the control signals) transmission to the MARS subsystem via VME 96 interface. The details of the subsystem behavior defined by the video pre-processor firmware component are provided in Section 4.2.1.3.

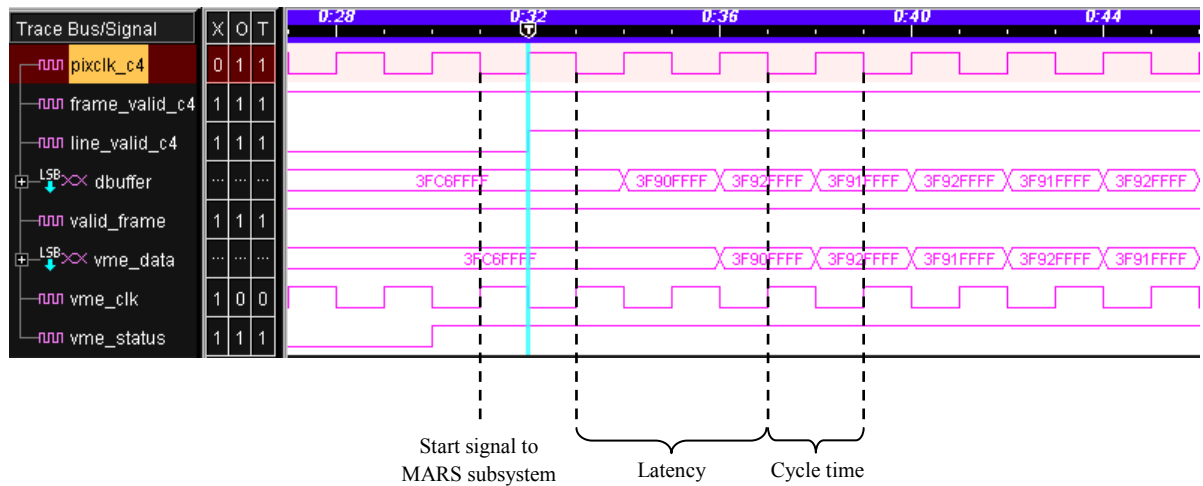


Figure 5.7: 4-Vision Operation Timing Diagram

The timing points on the above figure demonstrate the important moments of the subsystem operation including the start of transmission to the MARS subsystem indicated by the vme\_status signal, initial transmission latency for the first packet and the transmission cycle time for the subsequent packets. As it can be seen, operation of the subsystem fully conforms to the principles of FPGA based implementation described in Section 4.2.1.2.1. Here the expected latency equals to 2 pixel clock cycles and the cycle time is equal to 1 pixel clock cycle. No other delays or additional signals for synchronization were required.

Therefore, there is no recommendation for modification of the video-acquisition and packet combination circuit for the next version of the system.

### 5.3.3. Packet Reception and Frame Buffering Time Analysis

Following the experiments on packet transfer from the 4-Vision subsystem, the set of experiments on packet reception and video frame(s) buffering is needed to be done on the MARS subsystem. The experiments associated with the timing characteristics of the MARS subsystem were conducted to determine the performance of the reception of the data packets and their buffering to the memory banks (SRAM banks). These processes are illustrated in Figure 5.8, obtained from the Xilinx ChipScope Pro tool.

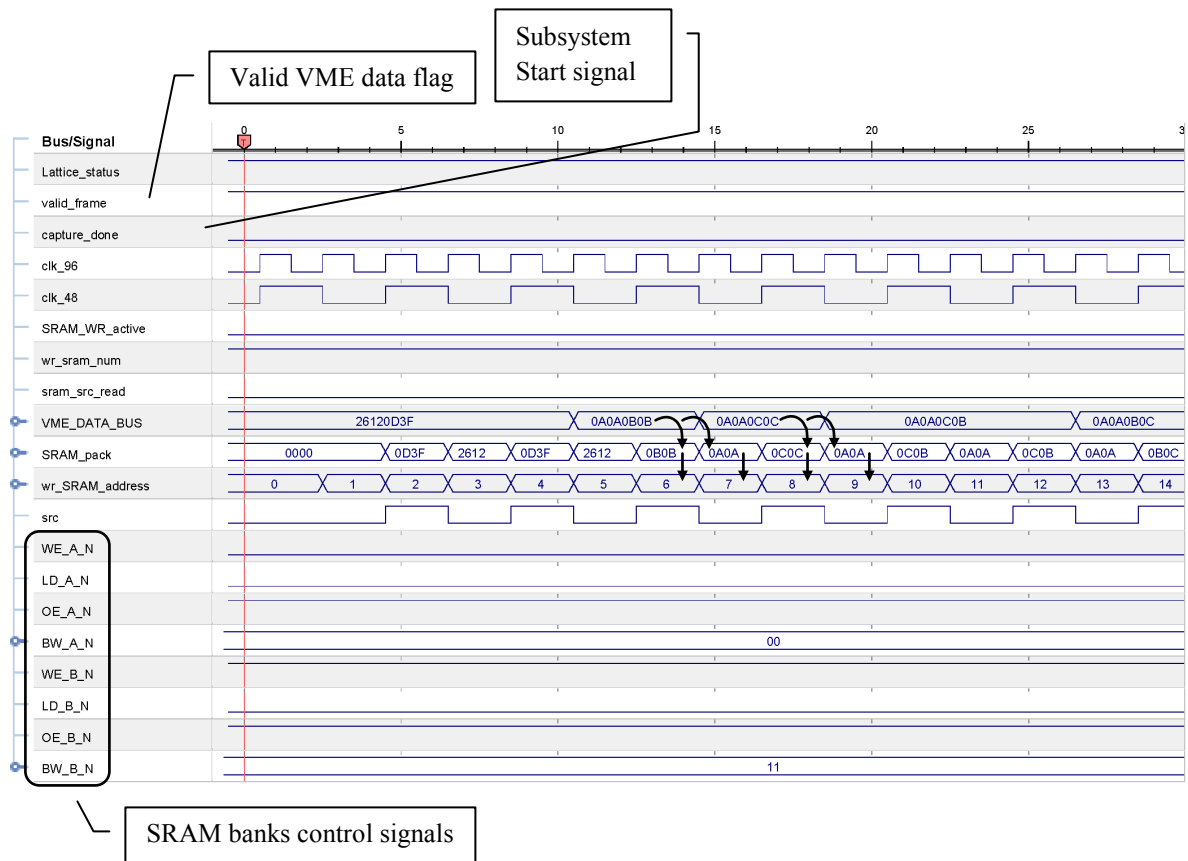


Figure 5.8: Packet Reception and Frame Buffering Processes

As it can be seen, the control signal `valid_frame` indicating validity of VME data is at logic HIGH. Together with `capture_done` signal at logic LOW, this signal activates reception of packets and then activates the storage of the video data in the SRAM bank. The arrows on the diagram illustrate the distribution of 30-bit packet to the data input of SRAM bank in accordance to the scheme presented in the Section 4.2.2.1.1.

As for the actual measurements from the ChipScope, the data from the VME bus being received, according to cycle time equal to pixel clock, is then divided into two words and is sequentially stored in SRAM memory according to 96 MHz internal clock. There was no need for timing adjustment or further improvement of this process.

#### 5.3.4. Analysis of Bayer Color Decoding and RGB Pixels Output Timing

The experimental testing for the timing associated with color interpolation process and process of parallel synchronous RGB pixel output to VGA display channels were needed for accurate synthesis of pixel colors and synchronous image outputting to the VGA ports. The timing was obtained from the Xilinx ChipScope Pro tool similar to the previously performed packet reception and frame buffering timing analysis.

Firstly, the timing analysis for pixels distribution from the SRAM bank at 130MHz to the pixel row buffers was conducted. At the same time, distribution of previously buffered rows at 65MHz to Bayer color decoders dedicated for each video stream was recorded. The detailed description of these operations is provided in Section 4.2.2.1.1 and Section 4.2.2.2.2. The actual timing recorded by Xilinx ChipScope is shown in Figure 5.9.



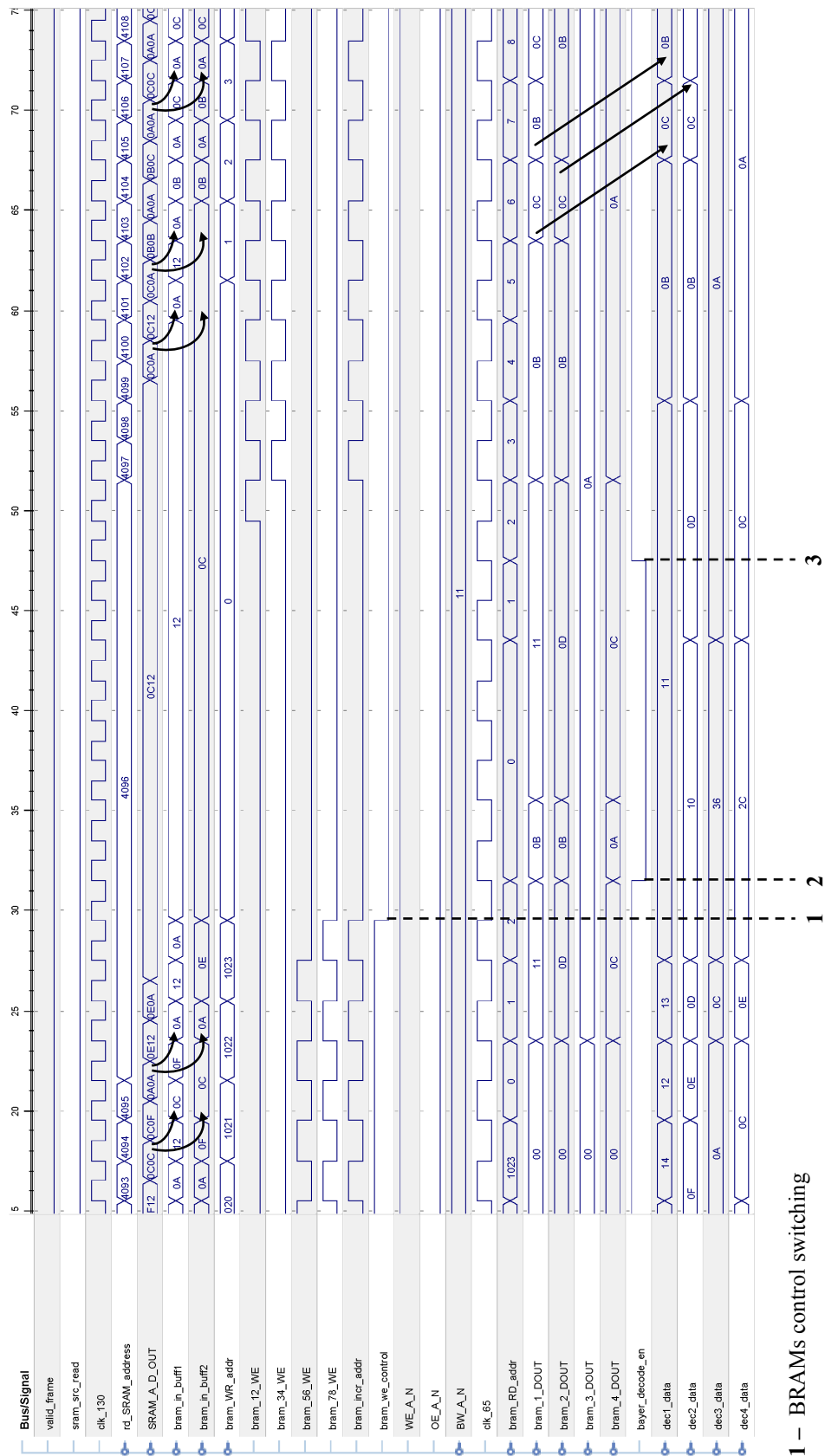


Figure 5.9: Pixel Distribution to the Bayer Color Decoders

Then, the process of the RGB output from Bayer color decoders attached to the inputs of VGA channels was analyzed. This process is shown in Figure 5.10 below. As it can be seen from the image, the enable signal for Bayer color decoders is applied 5 c.c. before the vertical and horizontal signals transition to logic HIGH. This is done to adjust the arrival of the first valid color pixel to the active line window (see the VGA timing diagram in Figure 4.19).

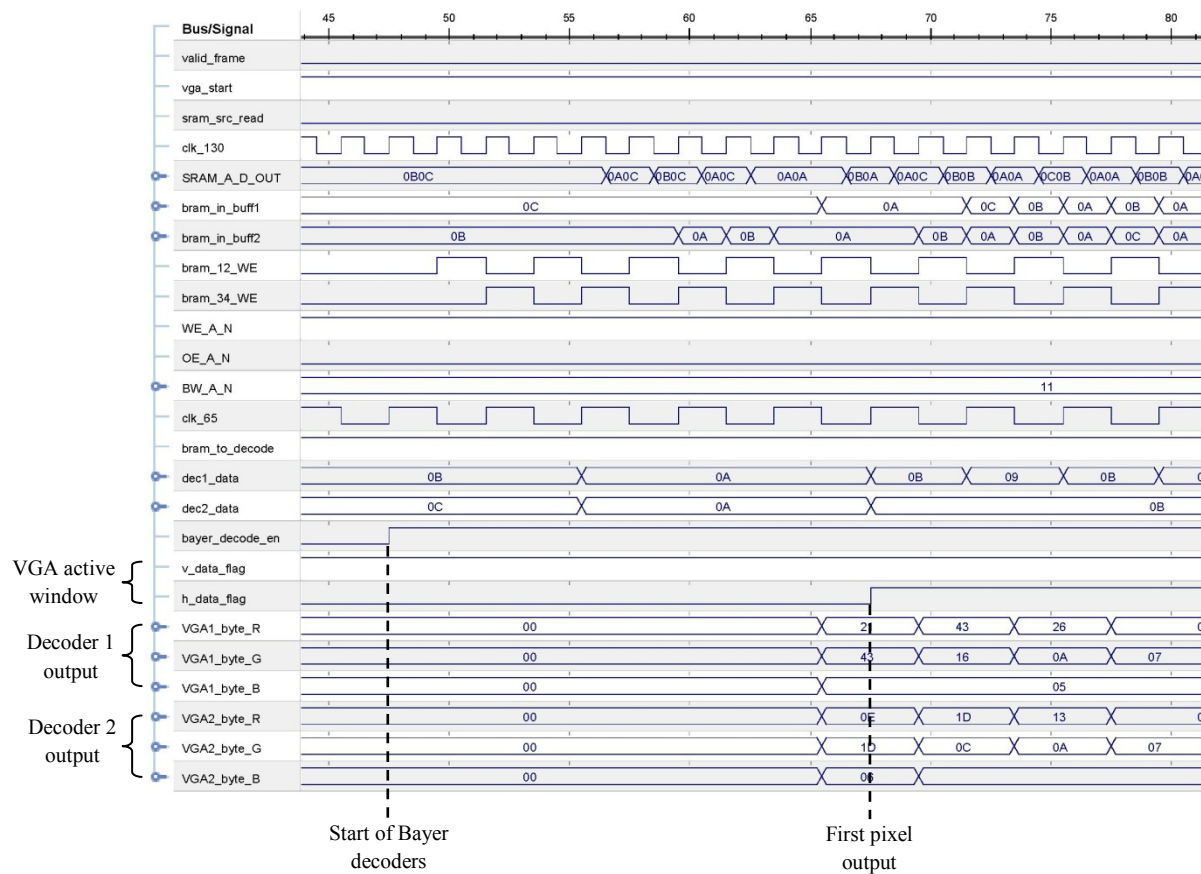


Figure 5.10: RGB Components Output to VGA

In addition to the RGB pixel outputs, Figure 5.10 shows operation stages described in the previous two sections, thus demonstrating the parallelizable nature of the FPGA based

implementation. All the requested timing was accurately adjusted. The latency of 5 c.c. = 77ns is acceptable for VGA output timing as well as the cycle time equal to 1 c.c. = 15.38 ns.

#### 5.4. Compression Rate Analysis

In order to estimate the system performance involving JPEG2000 compression, the following set of tests were performed using open source JPEG2000 software codec [59].

The tests involved compression of the RAW frame taken by the camera module from the 3D-P Camera setup. The sample frame is illustrated in Figure 5.11. The distance from the camera module to the image under evaluation (numerical test table) is equal to 65cm. This distance was selected based on an average human arms length defining an operational area.

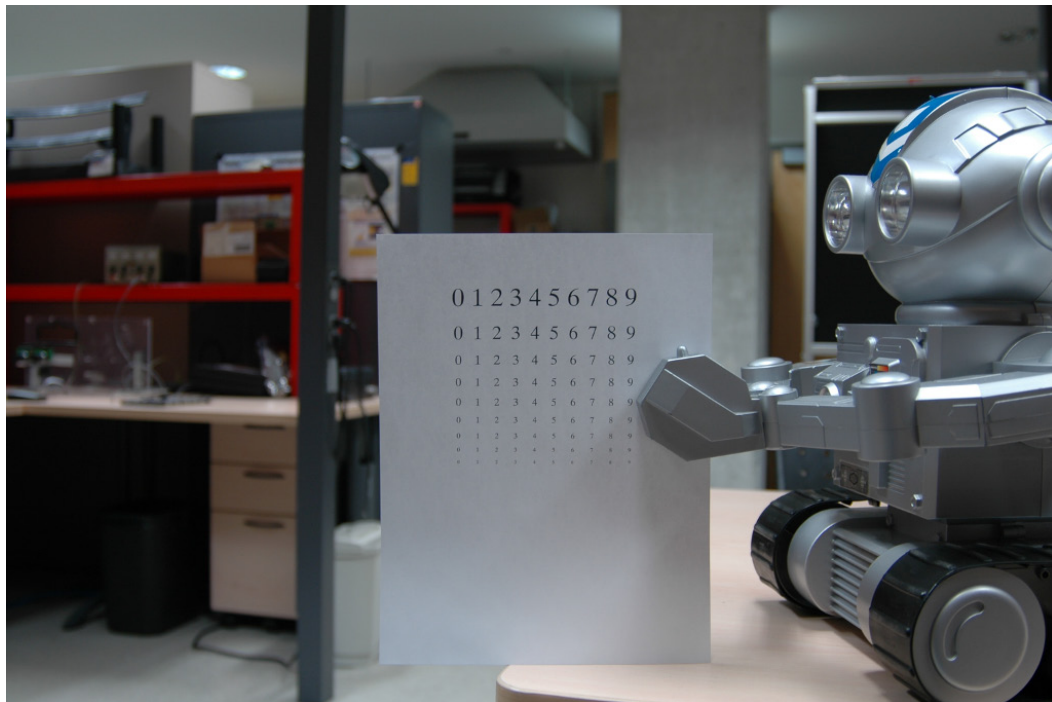
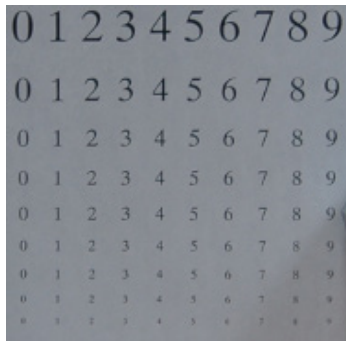
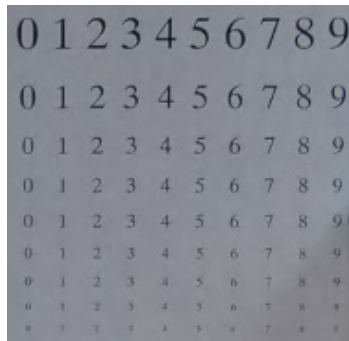


Figure 5.11: Original Sample RAW Frame

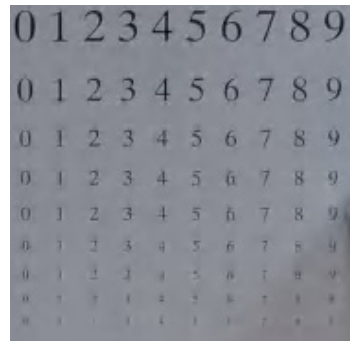
The JPEG2000 compression of the RAW frame was performed in lossless mode, without frame tiling, at 50, 100, 150, 200, 250, 300, 400, 500 compression rates. After the compression, the decompression was performed using the same JPEG2000 software decompressor [59]. Figures 5.12 (a) to (i) demonstrate the image quality degradation of the numerical rows with the increase of the compression rate.



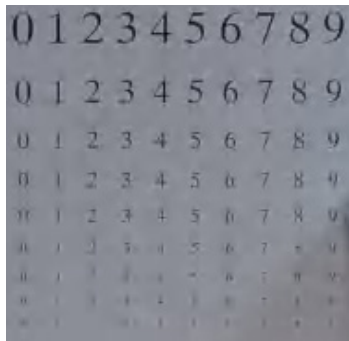
(a)



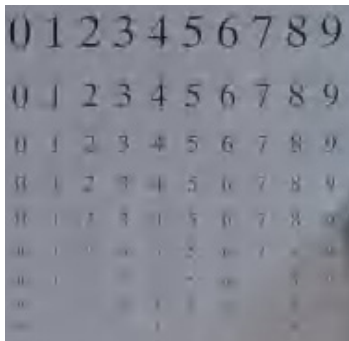
(b)



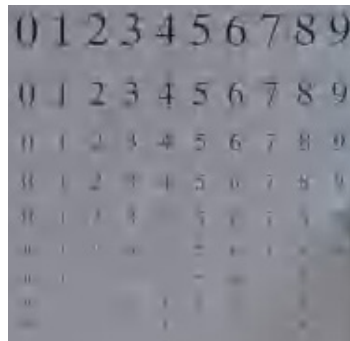
(c)



(d)



(e)



(f)

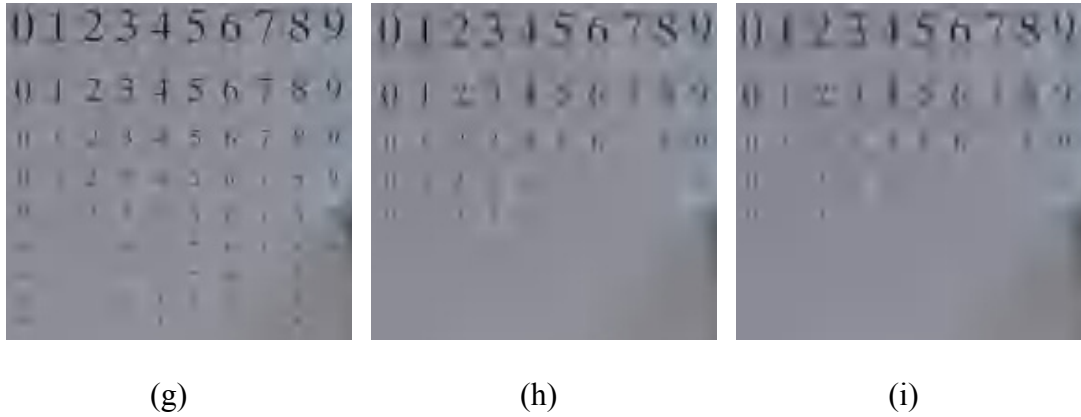


Figure 5.12: Compression performance. Samples: (a) – original, (b) – 50, (c) – 100, (d) – 150, (e) – 200, (f) – 250, (g) – 300, (h) – 400, (i) – 500

As it can be seen from the above samples, the clarity of image decreases according to the increase of the compression rate. The quantitative analysis of this clarity was done similarly to optometric tests. The size of the symbols (numbers) in the test table was measured according to the standard font dimensions. Table 5.2 shows the dimensions of symbols (numbers) arranged from top to bottom from the smallest font size (row 1) to the biggest font size (row 9).

Table 5.2: Row Number to Font Size Correspondence

Row number	Font size	Dimension (mm x mm)
1	9	2x1
2	11	2.5x1
3	14	3x1.5
4	16	3.5x2
5	20	4.5x2.5
6	22	5x3
7	26	6x4
8	36	8x5
9	48	11x7

The results of JPEG2000 compression analysis are presented on diagram in Figure 5.13.

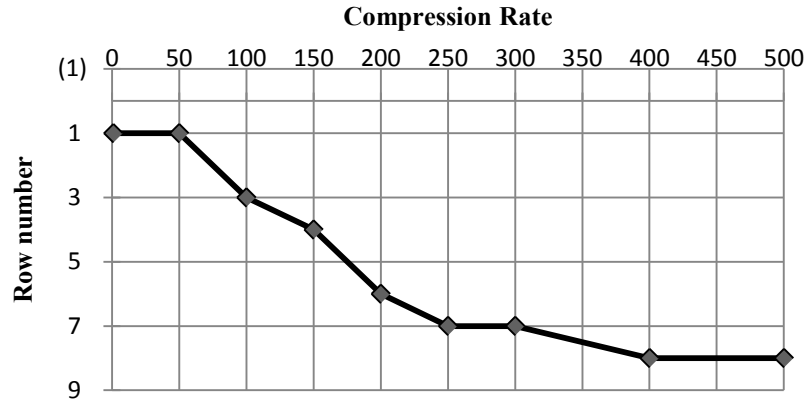


Figure 5.13: Image Quality in Accordance to JPEG2000 Compression Rate

The X axis of this diagram represents the compression rate, whereas the Y axis represents the row number. The diagram shows the minimal font size that can be recognized during operation at the specific compression rate. As it can be seen on the diagram, even the smallest font can be recognized up to compression rate equal to 50. Then, the degradation of the image quality is quite linear from the compression rate of 50 to 250. The quality of the image stays almost the same until the compression rate equals 300, where the 3 top rows are still readable. The 9<sup>th</sup> row can still be recognized until the compression rate equals 500.

The above experiments showed that: i) the compression rate for the stereoscopic images of the operational area should not exceed 50 and ii) the images of the right and left peripheral areas can be compressed up to the rate of 250.

Therefore, the minimum transmission bandwidth ( $BW_{3D-P}$ ) for the 3D-P transport stream can be estimated as follows:

$$\begin{aligned}
BW_{3D-P} = & \left( \frac{2 * [Res_{video} * Res_{pixel} * R_{frame}]}{C_{rate}} \right)_{stereo} + \\
& + \left( \frac{2 * [Res_{video} * Res_{pixel} * R_{frame}]}{C_{rate}} \right)_{peripheral},
\end{aligned}
\tag{5.1}$$

where  $Res_{video}$  is a video frame resolution,  $Res_{pixel}$  is a number of bits used to represent every pixel,  $R_{frame}$  is a video frame rate per second and  $C_{rate}$  is the compression rate.

By using the above Equation 5.1, the required transmission bandwidth for the 720p30 and for the 1080p30 standards are calculated as:

$$\begin{aligned}
BW_{3D-P_{720p30}} = & \left( \frac{2 * [(1280 * 720)pixel * 10bit * 30fps]}{50} \right) + \\
& + \left( \frac{2 * [(1280 * 720)pixel * 10bit * 30fps]}{250} \right) = 12.6Mbps
\end{aligned}$$

$$\begin{aligned}
BW_{3D-P_{1080p30}} = & \left( \frac{2 * [(1920 * 1080)pixel * 10bit * 30fps]}{50} \right) + \\
& + \left( \frac{2 * [(1920 * 1080)pixel * 10bit * 30fps]}{250} \right) = 28.4Mbps
\end{aligned}$$

Thus, in the next version of 3D-P 4-Vision subsystem a serial transceiver with the bandwidth not exceeding 30 Mbps can be utilized instead of parallel VME bus.

### 5.5. Power Consumption Analysis

The experiments associated with timing, conducted on the prototype of 3D-P Telepresence System, have shown that the requested performance can be reached with relatively low clock frequencies from 48MHz on the 4-Vision subsystem to 130MHz on the MARS subsystem. This was possible due to a wide parallelization and pipelining of the video-acquisition, video-processing and video-outputting processes. In turn, the low range of clock frequencies allows relatively low power consumption. This, however, needs to be proved by experiments associated with power consumption in all modes of operation.

In order to analyze the power consumption of the 3D-P Telepresence System in the operation modes described in Section 3.2, the set of tests involving measurements of the static and dynamic power was performed. The static measurements were performed without loading the configuration bitstreams to the FPGAs, whereas dynamic measurements were performed during normal system operation in every possible mode. The input voltage according to specification of DC-DC converters used in the prototype was set to 5.5 VDC. Thus, the current-  $I$  was measured for all modes of operation and the consumed power was calculated accordingly:  $P = 5.5 \text{ VDC} * I$ . The results are collected in the Table 5.3 below.



Table 5.3: Power Consumption Chart

Mode	4-Vision subsystem		MARS subsystem		Total	
	Static (mA)	Dynamic (mA)	Static (A)	Dynamic (mA)	Amps	Watts
3D-P mode	160	330	1.02	230	1.74	9.57
Panoramic mode	160	290	1.02	220	1.69	9.295
3D mode	160	260	1.02	220	1.66	9.13
Telescopic mode	160	220	1.02	180	1.58	8.69

As for Table 5.3, the static current consumption of the 4-Vision and MARS subsystems doesn't depend on the operational mode. However, dynamic current measurements reflected the utilization of FPGA logic as well as utilization of camera modules in the particular operational mode. Furthermore, the variation of total power for the 4-Vision subsystem deployed at the remote location is in range from 2W (in telescopic mode) to 2.7W (in full 3D-P mode). Thus, the variation of power consumption between modes was equal to 35%. However, in "sleep" mode the actual power consumption is close to zero. It is necessary to mention that power consumption of the 4-Vision subsystem is most critical due to dependence on an autonomous power supply (e.g. on-board batteries). The power consumption of the MARS subsystem and associated video monitors and video-projectors is not so critical due to deployment at a control centre site.

The above experiments show that the 4-Vision subsystem can be powered by batteries with relatively low capacity. For example, 6VDC x 10Ah rechargeable battery will allow non-stop operation of the 4-Vision subsystem for almost twenty-four hours.

On the other hand, variation of power consumption for the MARS subsystem is much less; from 6.6 W (in telescopic mode) to 6.875 W (for 3D-P mode) and is equal to 4.2%.

This variation is actually negligible if power consumption of the video displays is taken into account. Therefore, standard power supply can be used for the MARS subsystem. However, relatively low power dissipation (not exceeding 7W) makes possible avoidance of any kind of coolers (e.g. fans) or even heat sinks in the design of the MARS subsystem.

## 5.6. Utilization of Experimental Results in 4-Vision2 Subsystem Design

The analysis of the experimental results associated with different aspects of the 3D-P Telepresence System performance allowed certain modifications in the 4-Vision subsystem design. The following modifications were made according to experimental results:

1. Two video sensors for stereoscopic image capture were located on the 4-Vision2 board. This solution allows minimization of signal cross talk and simplifies implementation of stereo-rectification procedure;
2. Video-sensors have a “sleep” mode for minimization of power consumption in the idle stage;
3. New subsystem 4V-J2C was added to the 4-Vision2 module to provide four parallel channels of run-time & synchronous JPEG2000 compression;
4. The 4V-J2C subsystem also allows combining the compressed video-streams to the serial transport stream (e.g. ASI transport stream) using an FPGA based combiner;
5. Different serial interfaces (ports) are added to the design of 4V-J2C: USB, HotLink;

Therefore, the operational hardware in the second version of the 3D-P Telepresence System consists of 4-Vision2 and 4V-J2C subsystems. This part of the system is described in detail in Section 4.3. Figure 5.14 illustrates the image of the 4-Vision2 subsystem.

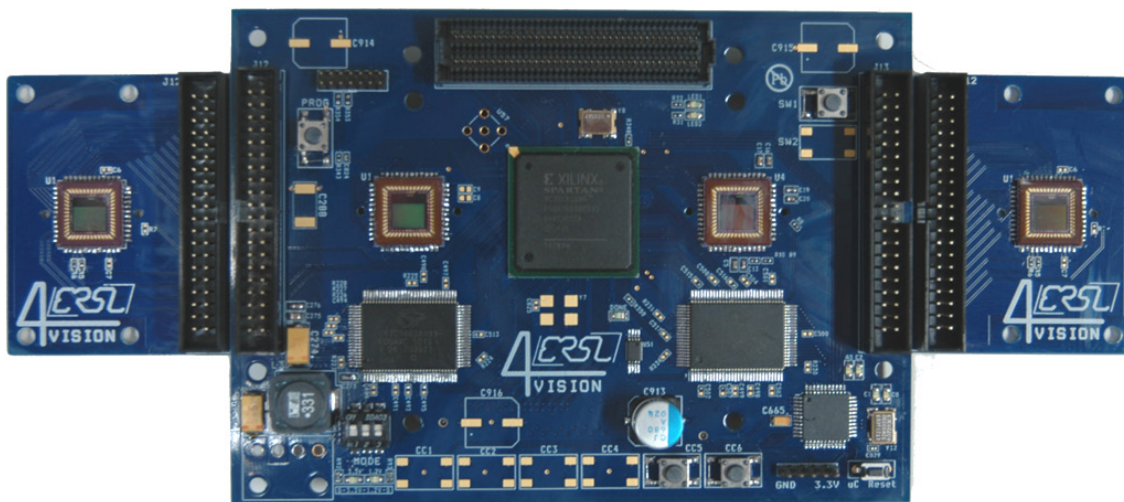


Figure 5.14: Image of the 4-Vision2 Subsystem

The 4-Vision2 subsystem consists of the Xilinx XC3S1400AN FPGA, Microchip PIC24FJ64GA104 microcontroller, Cypress CY7C1460AV33, four Aptina MT9T031 CMOS image sensors and two cross-platform SAMTEC FMC interfaces ASP-134603-01, ASP-134604-01. The on-board clock frequency is estimated to be up to 200MHz.

The image of the 4V-J2C subsystem is illustrated on the Figure 5.15.

It incorporates the same Xilinx XC3S1400AN FPGA used in 4-Vision2 platform, group of four JPEG2000 codecs - ADV212, Xilinx XC3S200AN FPGA, two flash memory modules: Micron MT29F4G16AAC, the HotLink transceiver Cypress CYP15G0201DXB, USB controller & FIFO buffer: FTDI FT2232HL and cross-platform SAMTEC FMC interface ASP-134603-01. The on-board clock frequency is estimated to be up to 200MHz.

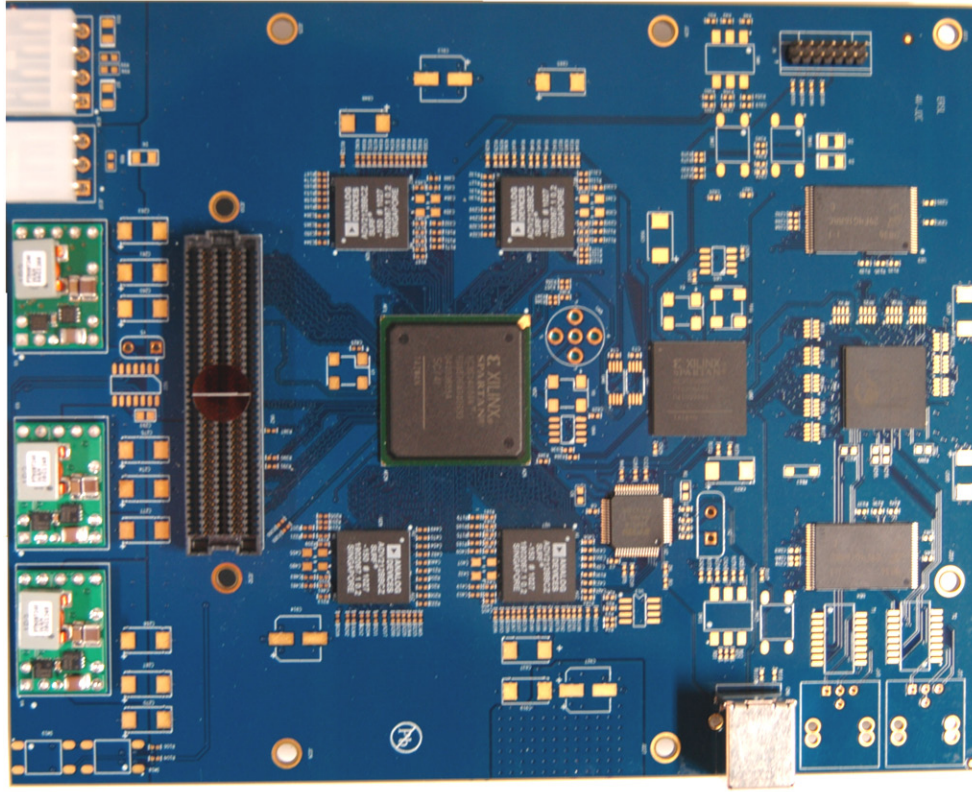


Figure 5.15: Image of 4V-J2C subsystem

The 4-Vision2 subsystem is integrated with the 4V-J2C subsystem via FMC interface in accordance to Section 4.3.

## 5.7. Conclusion

In this Chapter the results of the experiments were analyzed, and recommendations for further modifications of the 3D-P Telepresence System were made.

For this purpose, the special experimental setup was created on the basis of the first prototype of the 3D-P Telepresence System which consisted of the 3D-Panoramic video-

acquisition subsystem – 4-Vision and the video-stream processing and displaying subsystem – Multi-stream Adaptive Reconfigurable System (MARS).

The set of experiments consisted of the following evaluations: start-up time analysis; important for any real-time embedded system, timing characteristics analysis of on-chip operations, power consumption analysis and finally the analysis of the JPEG2000 compression performance. According to the gained results and their analyses, a set of recommendations were made for the next version of the system.

Finally, implementation of the next version of the 3D-P Telepresence System was presented with mention of the main reasons for the modifications in the design.

## 6. CONCLUSION AND FUTURE WORK

The goal of this work was to develop the conceptually new platform architecture for the wide class of telepresence systems specialized for remotely controlled mobile robotic applications. The main motivation for this was a very high demand for the controlled robotic systems which can eliminate the need of human exposure in various hazardous environments such as nuclear power plants, outer space and underwater operations, and mining etc. On the other hand, the replacement of humans by a robotic system requires existence of natural sensory feedback from the remote environment. First of all this refers to visual information consisting of front binocular (3D) vision and associated peripheral (2D) vision. This approach gives the operator immersive telepresence sensation and therefore simplifies remote control and robotic manipulations in distant areas. The comprehensive search of existing teleoperation systems has shown several aspects which reduce the effectiveness of remote control of complex systems in real-time. These aspects are as follows:

- Reduced resolution of real-time video-information and/or frame rate mainly caused by: i) limited computational performance provided by a sequential type of processors; ii) electromechanical devices used in many systems for panoramic or stereo-panoramic area observation and iii) limited communication bandwidth or video-compression mechanisms with limited compression rate;
- Relatively high power consumption which limits implementation of 3D-Panoramic video-acquisition, pre-processing and video-transmission systems on mobile objects

with limited power budget. The reason for this is also utilization of conventional CPU / GPU platforms for video-processing and compression;

- Limited ability of real-time adaptation for different operational modes due to dedicated hardware circuits used for video-acquisition, transport stream combining and other stream processing functions.

Therefore, in this thesis, a novel approach was proposed to mitigate the above problems using recent technological advancements in reconfigurable logic devices, digital HD-video sensors and compression/ decompression mechanisms.

As the result, a concept of run-time reconfigurable computing platform was proposed for multi-mode and multi-stream video-processing. This made possible the utilization of parallel acquisition, parallel video-processing and parallel video-displaying of video-frames in HD format and a relatively high frame rate. Furthermore, utilization of Field Programmable Gate Array (FPGA) devices allowed the reduction of power consumption in one or two orders of magnitude due to a wide parallelization of computational processes and thus, a dramatic reduction of operational clock frequency. On the other hand, utilization of recent FPGA devices allowed seamless mode switching by run-time FPGA reconfiguration (during the period of one video-frame).

Analysis of performance characteristics of different video-compression mechanisms demonstrated big advantages of wavelet based compression (e.g. JPEG2000) for stereo-vision applications comparing to DCT based algorithms (e.g. MPEG-2 or MPEG-4). This was another aspect added to the proposed concept of the platform for 3D-Panoramic Telepresence Systems developed in capacity of this project.

The proposed concept was implemented in the prototype of the platform of 3D-P Telepresence System and performance parameters were evaluated. The analysis of the experimental result became a basis for recommendations for further modification of the developed platform architecture for this class of telepresence systems. Furthermore, these recommendations were used in revision of platform architecture and the next version of the platform was designed and manufactured.

During the project presented in this thesis the following contributions were made:

- Extended search and analysis of the state-of-the-art in the area of 3D-P Telepresence Systems for mobile robotic applications was conducted to find out pros and cons of existing systems. This analysis allowed for the determination of the state of the problem and possible approaches for problem mitigation;
- The concept of the framework of the reconfigurable platform for 3D-P Telepresence System was proposed and developed. This concept assumed multi-stream video-acquisition, parallel video-stream processing and parallel displaying subsystems;
- On the basis of the proposed framework, the platform architecture was developed in all levels of architecture: system level, on-board and on-chip levels;
- The architecture of 3D-P Telepresence System was implemented in the first prototype for further evaluation of feasibility of the proposed approach and collection of experimental data. This allowed for the analysis of the system performance based on the obtained data;
- Performance analysis of the first prototype was used to generate the set of modifications in platform architecture. These recommendations were used for the



design of the next version of the platform for 3D-P Telepresence Systems. This platform also was prototyped.

The first system prototype being demonstrated in the annual conference SVAR-2010: Space Vision and Advanced Robotics held at MDA Space Missions in Brampton, Ontario in June 2010 was awarded with the first prize for the best presentation and demo.

The updated version of the platform prototype was recently demonstrated at the exhibition “Discovery 11” organized by Ontario Centres of Excellence (OCE) and held at the Metro Toronto Conventional Centre in May 2011.

The future work regarding further development of the platforms for 3D-P Telepresence Systems may include but not limited to the following aspects for research and development:

- Detailed analysis of HD video compression / decompression mechanisms. This aspect will need to compare the effectiveness of implementation of wavelet based algorithms in dedicated circuits (e.g. ASICs) and reconfigurable cores to be implemented in FPGA devices;
- Evaluation of platform effectiveness for all spectrums of algorithms used for 3D panoramic vision (e.g. different schemes for stereo-rectification, image merging, object tracking algorithms etc.);
- Implementation and analysis of Natural User Interface (NUI) elements including 3D pointers and elements of augmented reality. This aspect may bring more complexity to the hardware and firmware parts of the platform architecture and therefore need to be investigated in detail.

Finally, it is necessary to say that the R&D efforts in the area of 3D-P Telepresence Systems may bring very high economical effect by replacing humans by machines in hazardous areas or areas where physical presence of a human operator is not effective or even possible. Therefore, a future development in this area seems very promising.

## BIBLIOGRAPHY

1. M. Minsky, "Telepresence," Omni, New York, pp. 45–51, 1980.
2. Feldmann I., Waizenegger W., Atzpadin N., Schreer O., "Real-time depth estimation for immersive 3D videoconferencing," *Proceedings of the 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, pp.1-4, 7-9 June 2010.
3. Kauff P., Schreer O., "An immersive 3D video-conferencing system using shared virtual team user environments," *In Proceedings of CVE'02*, pp. 105–112., 2002.
4. Sen Ma, Yuanyuan Shang, Junchan Du, "Design of Panoramic Mosaic Camera Based on FPGA," *In Proceedings of the International Conference on Information Engineering and Computer Science, ICIECS 2009*, pp.1-4, 19-20 Dec. 2009.
5. Marescaux J., Leroy J., Rubino F., Smith M., Vix M., Simone M. et al., "Transcontinental robot-assisted remote telesurgery: feasibility and potential applications," *Annals of Surgery*, vol. 235, no. 4, pp. 487–492, 2002.
6. Falk V., Diegler A., Walther T., Autschbach R., Mohr FW., "Developments in robotic cardiac surgery," *Current Opinion in Cardiology*, vol.15, no.6, 378–387, 2000.
7. Anwari M. "Remote Telepresence Surgery: The Canadian Experience," *Surgical Endoscopy*, vol. 21, no. 4, pp. 537–541, 2007.
8. Ferre M., Aracil R., Sanchez-Uran M., "Stereoscopic human interfaces," *IEEE Robotics & Automation Magazine*, vol.15, no.4, pp.50-57, Dec. 2008.
9. Kandel E. R., Schwartz, Jessell T., "Principles of Neural Science," *McGraw-Hill*, New York, ed. 4, pp. 558-560, 2000.

10. Kron, A., Schmidt G., Petzold B., Zah M.I., Hinterseer P., Steinbach E., "Disposal of explosive ordnances by use of a bimanual haptic telepresence system," *In Proceedings of the IEEE International Conference on Robotics and Automation ICRA '04*, vol.2, pp. 1968- 1973, April 26-May 1, 2004.
11. Chen J.Y.C., Haas E.C., Barnes M.J., "Human Performance Issues and User Interface Design for Teleoperated Robots," *IEEE Transactions on Systems, Management, and Cybernetics, Part C: Applications and Reviews*, vol.37, no.6, pp.1231-1245, Nov. 2007.
12. Amanatiadis A., Gasteratos A., Georgoulas C., Kotoulas L., Andreadis I., "Development of a stereo vision system for remotely operated robots: A control and video streaming architecture," *IEEE Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pp.14-19, 14-16 July 2008.
13. Khaleghi B., Ahuja S., Wu Q., "An improved real-time miniaturized embedded stereo vision system (MESVS-II)," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp.1-8, 23-28 June 2008.
14. Pohanka J., Pribula O., Fischer J., "An embedded stereovision system: Aspects of measurement precision," *In Proceedings of the 12th Biennial Baltic Electronics Conference (BEC)*, pp.157-160, 4-6 Oct. 2010.
15. Seunghun Jin, Junguk Cho, Xuan Dai Pham, Kyoung Mu Lee, Sung-Kee Park, Munsang Kim, Jae Wook Jeon, "FPGA Design and Implementation of a Real-Time Stereo Vision System," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.20, no.1, pp.15-26, Jan. 2010.

16. Pachidis T.P., Lygouras J.N., "Pseudostereo-Vision System: A Monocular Stereo-Vision System as a Sensor for Real-Time Robot Applications," *IEEE Transactions on Instrumentation and Measurement*, vol.56, no.6, pp.2547-2560, Dec. 2007.
17. Petty R., Robinson M., Evans J., "3D measurement using rotating line-scan sensors," *Measurement Science & Technology*, vol. 9, no. 3, pp. 339–346, March 1998.
18. Barth, M., Barrows C., "A fast panoramic imaging system and intelligent imaging technique for mobile robots," *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp.626-633 4-8 Nov 1996.
19. Gijeong Jang, Sungho Kim, Inso Kweon, "Single-camera panoramic stereo system with single-viewpoint optics," *Optics Letters* vol. 31, no.1, pp. 41-43, 2006.
20. Peleg S., Pritch Y., Ben-Ezra M., "Cameras for stereo panoramic imaging," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp.208-214, 2000.
21. Chahl J.S., Srinivasan M.V., "A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation," *In Proceedings of the IEEE Workshop on Omnidirectional Vision*, pp.104-111, 2000.
22. Nakao T., Kashitani A., "Panoramic camera using a mirror rotation mechanism and a fast image mosaicing," *In Proceedings of the International Conference on Image Processing*, vol. 2, pp.1045-1048, 7-10 Oct 2001.
23. Tzavidas S., Katsaggelos A.K., "A multicamera setup for generating stereo panoramic video," *IEEE Transactions on multimedia*, vol.7, no.5, pp. 880- 890, Oct. 2005.
24. Howard I. P., Rogers B. J., "Binocular Vision and Stereopsis," Oxford University Press, London, U.K.,1995.

25. [www.analog.com](http://www.analog.com), accessed on June 9, 2011.
26. Rolland J. P., Hua H., "Head-mounted display systems," Encyclopedia of Optical Engineering, R. G. Driggers, ed., Taylor & Francis, 2003.
27. Delaney B., "Forget the funny glasses (autostereoscopic display systems)," *IEEE Computer Graphics and Applications*, vol.25, no.3, pp.14-19, May-June 2005.
28. Iastrebov V., Seet G., Asokan T., Chui Y., Lau M.W.S., "Vision enhancement using stereoscopic telepresence for remotely operated underwater robotic vehicles," *Journal of Intelligent & Robotic Systems*, vol. 52, no.1, pp. 139-154, 2008.
29. Symes Peter, "Digital video compression," McGraw-Hill Professional, New York, pp. 131-136, 2004.
30. Hanzo L., Cherriman P., Streit J., "Video Compression and Communications: From Basics To H.261, H.263, H.264, MPEG4 For DVB and HSDPA-Style Adaptive Turbo-Transceivers," John Wiley and Sons Inc, 2<sup>nd</sup> ed., 2007.
31. ISO/IEC 15444-1, JPEG2000 Image Coding System, Part 1, 2004.
32. ISO/IEC 14496-2, Coding of audio-visual objects, Part 2: Visual, 2004.
33. ISO/IEC 14496-10, Coding of audio-visual objects, Part 10, 2005.
34. Taubman D., Marcellin M., "JPEG2000 Image Compression Fundamentals, Standards and Practice", Kluwer Academic Publishers, 2002.
35. Eyadat M., Muhi I., "Compression standards roles in image processing: case study," *In Proceedings of the Information Technology International Conference on Coding and Computing*, vol.2, pp. 135- 140, 4-6 April 2005.

36. Boxin Shi, Lin Liu, Chao Xu, "Comparison between JPEG2000 and H.264 for digital cinema," *In Proceedings of IEEE International Conference on Multimedia and Expo*, pp.725-728, June 23 2008-April 26 2008.
37. Samet A., Benn Ayed M.A., Bouhlef M.S., Loulou M., Masmoudi N., Kamoun, L., "JPEG 2000: performance and evaluation," *In Proceedings of IEEE International Conference on Systems Management and Cybernetics*, vol.5, pp. 6-9 Oct. 2002.
38. Hennessy J. L., Patterson D. A., Goldberg D., Asanovic K., "Computer architecture: A quantitative approach," Morgan Kaufmann Publishers, San Francisco, CA. 3rd ed., 2003.
39. Kuon I., Tessier R., Rose J., "FPGA architecture," Now Publishers, 2008.
40. www.xilinx.com, accessed on June 9, 2011.
41. www.altera.com, accessed on June 9, 2011.
42. www.actel.com, accessed on June 9, 2011.
43. www.latticesemi.com, accessed on June 9, 2011.
44. Asano S., Maruyama T., Yamaguchi Y., "Performance comparison of FPGA, GPU and CPU in image processing," *In Proceedings of International Conference on Field Programmable Logic and Applications*, pp.126-131, Aug. 31 2009-Sept. 2 2009.
45. Nan Zhang, Yun-shan Chen, Jian-li Wang, "Image parallel processing based on GPU," *In Proceedings of 2<sup>nd</sup> International Conference on Advanced Computer Control*, vol. 3, pp.367-370, 27-29 March 2010.
46. Kalarot R., Morris, J., "Comparison of FPGA and GPU implementations of real-time stereo vision," *In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 9-15, 13-18 June 2010.

47. Grasczew G., Roelofs T.A., Rakowsky S., Schlag P.M., "E-health and telemedicine," International journal of computer assisted radiology and surgery, vol.1, Springer, Berlin, 2006.
48. Anvari M., "Reaching the rural world through robotic surgical programs," *European Surgery*, vol.37, no.5, Springer, Wien, 2005.
49. Eadie L.H., Seifalian A.M., Davidson B.R., "Telemedicine in surgery," *British Journal of Surgery*, John Wiley & Sons, vol.90, no. 6, pp. 647–658, 2003.
50. Trevelyan J. P., Kang S., Hamel W.R., "Robotics in hazardous applications," *Springer handbook of robotics*, Springer, Berlin Heidelberg, 2008.
51. ISO/IEC 13818-1, Generic coding of moving pictures and associated audio information: Systems, 2007.
52. Bayer B. E., "Color imaging array," US Patent No. 3971065, 1976.
53. IEEE Computer Society. "IEEE Standard Test Access Port and Boundary-Scan Architecture - IEEE Std. 1149.1-2001", IEEE, New York, 2001.
54. Saakov A., Marcantonio D., Dumitriu V., Kirischian L., "3D Panoramic Naturally Interactive Real-Time Telepresence System," *Presentation at workshop SVAR-2010: Space Vision and Advanced Robotics*, MDA Space Missions, Brampton, Canada, 2010.
55. Lukin A., Kubasov D., "High-quality algorithm for bayer interpolation," *Programming and Computer Software*, vol. 30, no.6, 2004.
56. Hsia S., "Fast high-quality color-filter-array interpolation method for digital camera systems," *IS&T Journal Electronic Imaging*, vol. 13, no. 1, pp. 244-247, Jan. 2004.
57. Tsai P.-S., Acharya T., Ray A.K., "Adaptive fuzzy color interpolation," *Journal of Electronic Imaging*, vol. 11, no. 3, pp. 293–305, July 2002.



58. Gunturk B.K., Glotzbach J., Altunbasak Y., Schafer R.W., Mersereau R.M.,  
"Demosaicking: color filter array interpolation," *IEEE Signal Processing Magazine*,  
vol.22, no.1, pp. 44- 54, Jan. 2005.
59. [www.openjpeg.org](http://www.openjpeg.org), accessed on June 9, 2011.
60. [www.samtec.com](http://www.samtec.com), accessed on June 9, 2011.
61. [www.fujifilmusa.com/products/optical\\_devices/machine-vision/index.html](http://www.fujifilmusa.com/products/optical_devices/machine-vision/index.html), accessed on  
June 9, 2011.
62. [www.ptgrey.com/products/accessories/index.asp?type=optic\\_mounts](http://www.ptgrey.com/products/accessories/index.asp?type=optic_mounts), accessed on June  
9, 2011.
63. [www.microchip.com](http://www.microchip.com), accessed on June 9, 2011.
64. [www.ccsinfo.com/content.php?page=compilers](http://www.ccsinfo.com/content.php?page=compilers), accessed on June 9, 2011.
65. [www.ftdichip.com](http://www.ftdichip.com), accessed on June 9, 2011.
66. [www.cypress.com](http://www.cypress.com), accessed on June 9, 2011.
67. [www.uniquesys.com](http://www.uniquesys.com), accessed on June 9, 2011.
68. [www.cisco.com/en/US/products/ps7060/index.html](http://www.cisco.com/en/US/products/ps7060/index.html), accessed on June 9, 2011.
69. [www8.hp.com/us/en/business-solutions/visual-collaboration/index.html](http://www8.hp.com/us/en/business-solutions/visual-collaboration/index.html), accessed on  
June 9, 2011.
70. [www.inition.co.uk/inition/products.php?CatID\\_=8](http://www.inition.co.uk/inition/products.php?CatID_=8), accessed on June 9, 2011.
71. [www.apioptics.com/circular-polarizers.html](http://www.apioptics.com/circular-polarizers.html), accessed on June 9, 2011.