DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON

ARCHITECTURAL DESIGN SPECIFICATION
CSE 4316: SENIOR DESIGN I
SUMMER 2019

TEAM R.O.B
CAMERA GIMBAL

HUNTER NGHIEM
DAVID CHAVEZ
NHAN LE
DONNIE LAM
MINH TRAM

Revision History

Revision	Date	Author(s)	Description
0.1	10.01.2015	GH	document creation
0.2	10.05.2015	AT, GH	complete draft
0.3	10.12.2015	AT, GH	release candidate 1
1.0	10.20.2015	AT, GH, CB	official release
1.1	10.31.2015	AL	added design review requests

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

The primary concept of the product is to provide remote vision, more specifically a video stream from an off-site camera to local VR headset. This process, at a high level point of view, is composed of three processes and therefore, the general structure of the software will also be divided into the same processes: Determine the user head orientation and motion from the VR headset, translate the updated 3D coordinates and orientation to camera driver motion, and updating the video feed back to the user VR headset.

In between these three process involves both software communication between functional block and software communication with physical hardware, and therefore there are in-between steps that will be discussed later in this document.

2 System Overview

As stated above, the primary functional blocks will be the VR Headset which provides orientation data, the host computer which will handle the majority of the calculation and behaves like a middleman for information delivery, and the real-time motion system which will drive the physical motion of the camera mounting system as provided by the host computer.

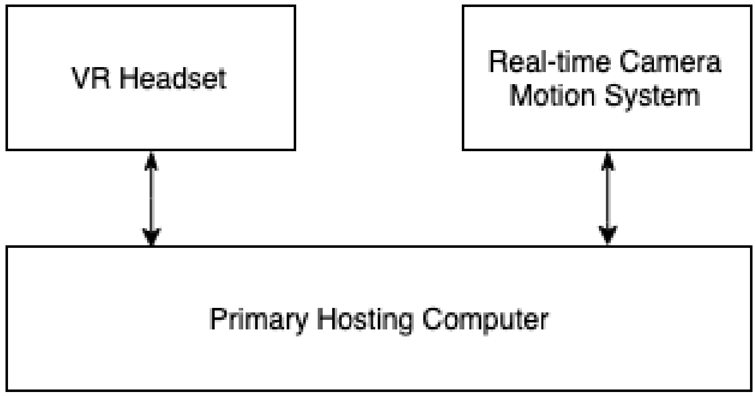

Figure 1: Functional block overview

2.1 VR Headset Subsystem

Since the VR Headset came equipped with internal gyroscope and accelerometer, in which x-y-z motion and orientation can be extracted, the software will poll for these values at an adjustable frequency set by the software, and provides these values for calculation at the host computer. Primary data transfer will be via USB 3.0.

2.2 Primary Hosting Computer Subsystem

When orientation information is received from the headset, the host computer will poll the camera motion system for its current orientation, calculate the difference with respect to the headset, and provide a set of adjustments that needs to be made to the motion system via direct serial communication. It will then wait for the motion to complete while simultaneously polls the off-site camera for an updated video frame via USB 3.0, and refresh the VR headset display. This is where the majority of the heavy calculation and data packing will be done to minimize delays. It is also required that the camera feed is open at all time when the software is operating to minimize setup overhead and simplify software functionality and interface.

2.3 Real-time Camera Motion Subsystem

The motion subsystem serves two purposes: act out necessary motion as provided by the host computer, and frequently updating it's current orientation to be synchronized with the VR headset when information is sent back to the host computer. This is done by on-site microcontroller (8 or 32-bit) via serial communication.

3 Subsystem Definitions & Data Flow

The following diagram breaks the system down into our intended dataflow design. This is for high-level viewing purposes only, to convey the main data streams of the system. Lack of connection here does not mean that there is no information communication between subsystems, as unnecessary details has been removed to maintain primary data stream clarity.

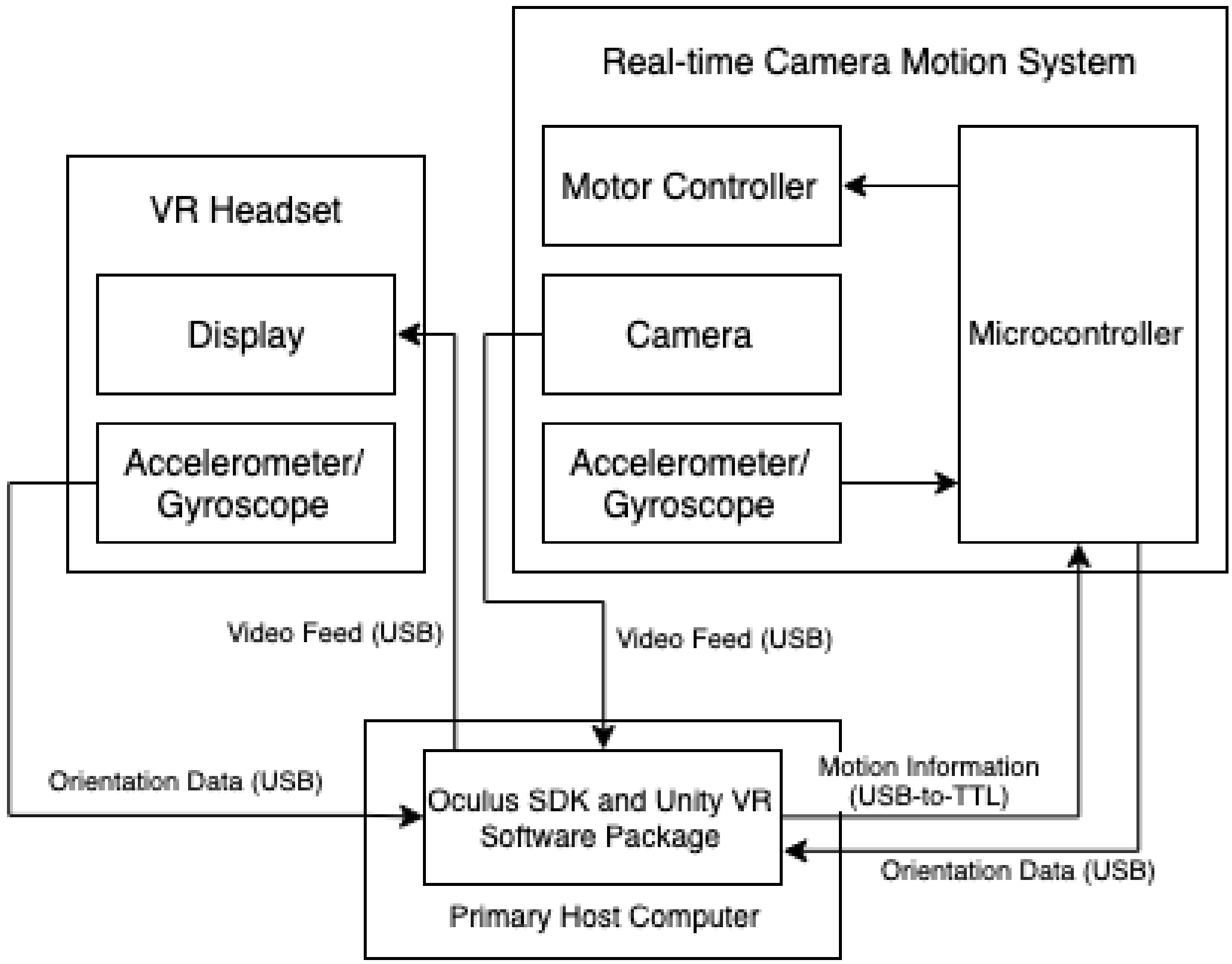

Figure 2: Detailed subsystems of primary functional systems

4 VR Headset Subsystems

4.1 Display Output

This component will only be waiting to be updated by the latest video feed provided by the primary hosting computer.

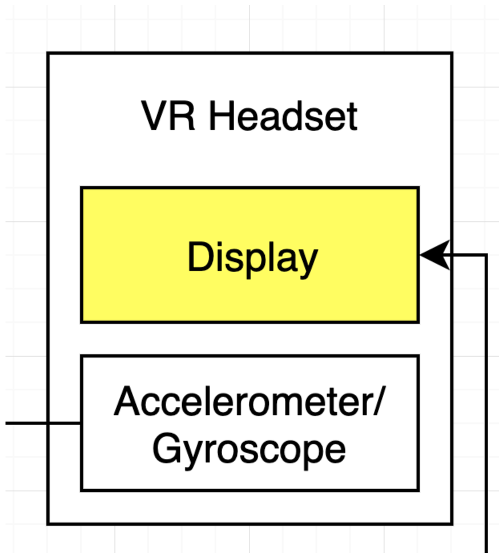

Figure 3: Display with no software output and one input

4.1.1 Assumptions

The VR headset is correctly configured and the display is calibrated to unique user's preferences.

4.1.2 Responsibilities

Output the video feed received from the primary hosting computer as soon as data is received. This task is prioritized at the highest level to ensure minimal delay between camera feed and display output.

4.1.3 Subsystem Interfaces

Table 2: Display interfaces

ID	Description	Inputs	Outputs
#1	USB 3.0	Video feed from host computer	N/A

4.2 VR Headset Internal Accelerometer and Gyroscope

This component will provide the headset orientation and relative 3D (x-y-z) coordinate to the primary hosting computer.

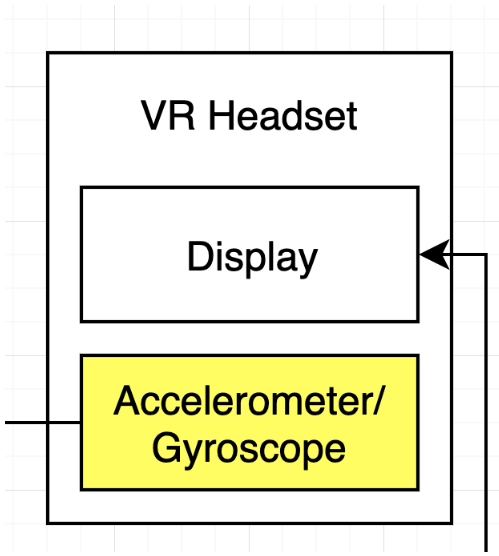

Figure 4: IMUs with single output and no input

4.2.1 Assumptions

The internal Inertial Measurement Unit(s) or IMUs are well calibrated and must be active at all time.

4.2.2 Responsibilities

Continuously (or upon request via interrupts, requests, etc.) transmit the headset current orientation and coordinates to the hosting computer for camera position update.

4.2.3 Subsystem Interfaces

Table 3: Headset internal IMUs interfaces

ID	Description	Inputs	Outputs
#1	USB 3.0	N/A	Current headset spatial information

5 Primary Hosting Computer Subsystems

5.1 Primary Software Package

This subsystem is build with the appropriate SDKs and interfaces to have bidirectional communication with both the VR headset and motion control subsystem. The entire software package is considered to be one whole unit, with all the necessary functionality build into it.

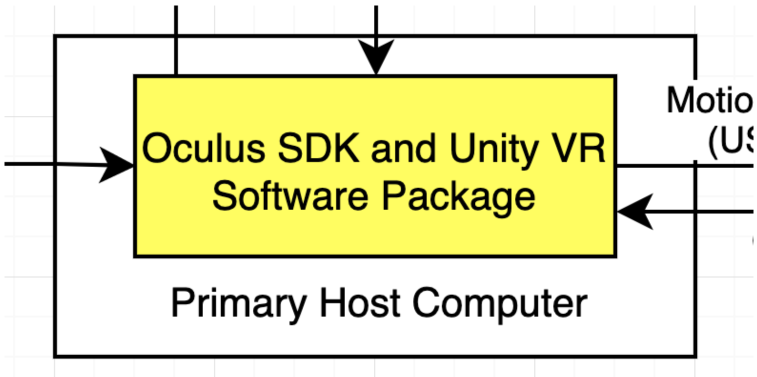

Figure 5: Internally self-contained software package on host computer

5.1.1 Assumptions

All connections to other subsystems are correct.

5.1.2 Responsibilities

Take the data received from both the headset and the motion control system, compute all necessary adjustments and updates then transfer it back to each appropriate system.

5.1.3 Subsystem Interfaces

Table 4: Software package interfaces

ID	Description	Inputs	Outputs
#1	Dedicated USB 3.0	VR headset orientation	N/A
#2	Dedicated USB 3.0	Camera video feed	Display video feed
#3	USB to TTL	Camera orientation	Camera adjustments

6 Real-time Camera Motion Subsystems

6.1 Microcontroller

This subsystem will be responsible for handling communication with the primary hosting computer and carry out the necessary orientation adjustment to synchronize the camera with the VR headset in real-time.

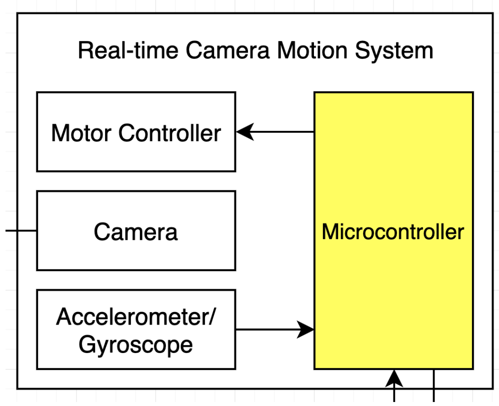

Figure 6: "Single" input and output stream MCU unit

6.1.1 Assumptions

All communication in and out of this subsystem follows standardized format as a data package to minimize transfer time and data corruption.

6.1.2 Responsibilities

Read orientation data provided by the hosting computer and drive the physical devices to correctly synchronize the camera with the VR headset. After corrections are made, relay new orientation data to the hosting computer as a signal to poll for new camera video feed.

6.1.3 Subsystem Interfaces

Table 5: Microcontroller interfaces

ID	Description	Inputs	Outputs
#1	USB to TTL	Camera adjustments	Post-adjustment orientation
#2	Direct connection	Current camera orientation	N/A
#3	Direct connection	N/A	Step count and direction to motor controller

6.2 Inertial Measurement Unit

This subsystem will provide current camera orientation data to the MCU unit to be transfer to the primary hosting computer.

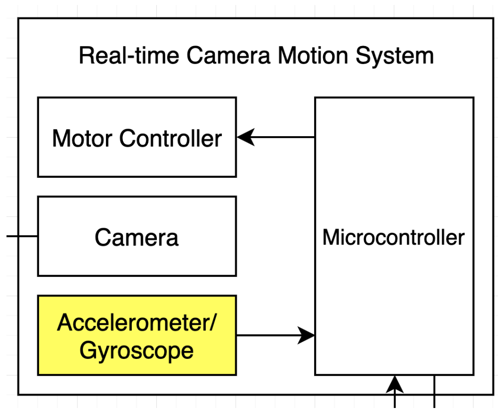

Figure 7: 3D information directly to MCU

6.2.1 Assumptions

This subsystem is properly calibrated and accounted for deviations.

6.2.2 Responsibilities

Computer current camera orientation and provide it to the MCU.

6.2.3 Subsystem Interfaces

Table 6: Camera IMU interfaces

ID	Description	Inputs	Outputs
#1	Direct connection	N/A	Current orientation

6.3 Live feed Camera Subsystem

This subsystem will provide the real-time video feed from the real world to the software package running on the hosting computer.

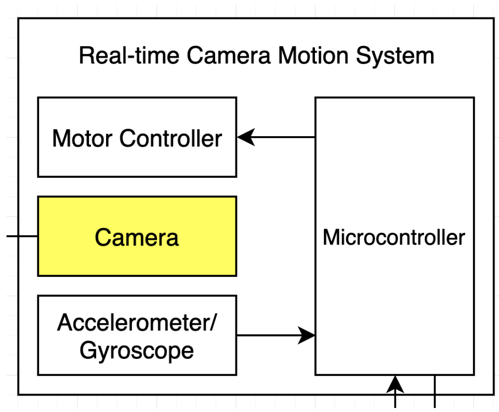

Figure 8: Single video feed output - stereo or mono

6.3.1 Assumptions

Camera orientation is correct and is rigidly mounted to the motor system.

6.3.2 Responsibilities

Provide video feed that will be handover to the VR headset via the hosting computer.

6.3.3 Subsystem Interfaces

Table 7: Camera interfaces

ID	Description	Inputs	Outputs
#1	USB 3.0	N/A	Live video feed

6.4 Motor Controller Subsystem

This system will carry out the necessary motor motion to correctly synchronize the camera and the VR headset.

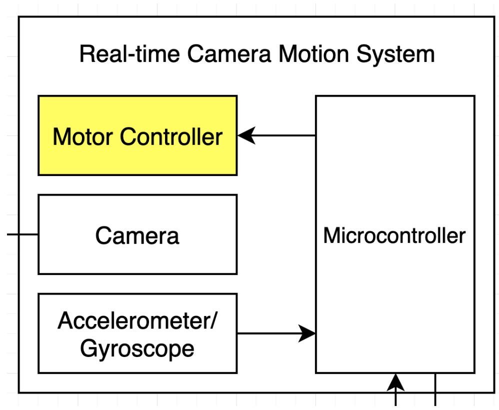

Figure 9: Direct data streaming to the MCU

6.4.1 Assumptions

Initial motor position are correctly homed and is in real-time sync with the VR headset.

6.4.2 Responsibilities

Move individual physical motors for each of the axis (roll-pitch-yaw) to simulate the provided VR headset orientation and coordinates.

6.4.3 Subsystem Interfaces

Table 8: Motor controller interfaces

ID	Description	Inputs	Outputs
#1	Direct connection	roll-pitch-yaw adjustment	N/A

REFERENCES