Niklas Blomqvist, Philip Johansson, Matteus Laurent, Johan Levinsson,
Oscar Petersson, Erik Peyronson

Group 41 - Design Specification

Högskoleingenjörsutbildning i datateknik, 180 hp

Design Specification - September 29, 2015
System Design - Project, HT15
TSIU03, Linköpings universitet

Supervisor:
Petter Källström
Department of Electrical Engineering (ISY)

Contents

1. Vad dokumentet ska innehålla:

The entire Design Spec should be 5-7 pages

- Now you are the engineers who receive the requirement specification. What you have to do is to propose a solution that fulfills the requirements of the system.
- Contrary to the requirements specifications, now you have to think about HOW you are going to meet the requirements.
- The design specification must describe the entire system. Think about the main blocks that your system will have, the functionality of each block and the interaction between them, i.e., which information the have to send to each other. Explain the functionality of the blocks and their interaction from a signal processing point of view, i.e., how the audio, video, etc. are processed in each block and which infomation is transmitted between blocks. You can provide som equations to show the algorithms that are applied. Note that this is very different from providing the hardware interfaces between the blocks.
- Later, think about the difficulties that you will find in hardware and the hardware limitations (timing, bandwidth, word length, etc.) and check that your design is viable. Some calculations may be necessary. For instance, if a requirrement says that the system must be able to delay the audio signal one second, you will probably think of using a memory in order to meet the requirement. Then you should make some calculations to check how big the memory must be and if it fits in the FPGA or if you need to use an external memory.
- The design specification must be described from the system level. Please, avoid details that are not relevant at that level. Also, make sure that the person who reads the document can get a clear idea of the entire system.
- As a result, the design specification must be a technical proposal that shows that you have analyzed the problem and found the difficulties that you will face, and provides a first approach to the solution. A good approach for writing the design specification is to present a block diagram of the system, provide a high-level description (at signal processing level) about the functionality of each block and how the blocks interact, and show which requirements present challenges and how you will solve them in hardware.

1.1 Layoutstandarder

Läs README:n i ../Projektrapport/ först och främst. Jag har strukturerat det hela genom att ha varje chapter och section i en fil med motsvarande namn, och nya borde inte behövas. Referera till bilder, avsnitt etc. på adekvat sätt med`\ref` (ger kapitelnummer eller figurnummer) och `\pageref` (ger sidnummer). Hänvisningar inom parentes sätts ej kursivt, i löptext anges de enligt (ta gärna en titt i description.tex):

```
\emph{<Typ> \ref{ref:name}: Name}
```

Jag föreslår även att vi använder oss av `\verb+name+` för att markera namn på moduler, signaler, etc.

Förhoppningsvis har ni vid det här laget bekantat er med min LaTeX-guide, men har ni frågor är det bara att hojta. Använd gärna emacs för att redigera dokumenten då det har stöd för uppmärkning av LaTeX-syntax.

Får jag igång SSH:n till burken därhemma via skolans burkar ska jag försöka hålla autokompileringen vid liv under dagen, men jag garanterar inget.

2. Introduction

Project 41 is based around audio signal processing. The audio input and output both go through the WM8731 chip on a DE2 board. Meanwhile, the hardware settings are controlled from a PS/2 keyboard and displayed on a VGA screen. The hardware settings to be implemented are a volume control and a balance control. In addition, an interface consisting of the input and output power level along with appropriate indicators as stated in the requirement specification.

The WM8731 is a stereo codec, which in Project 41 is used as a bridge between the audio source and a class-D amplifier. The custom hardware controls the WM8731 as the analysis of the input controls and encoding the graphical output. The output sound sent to a Class-D amplifier is then allowed further amplification through another instance of pulse width modulation within the amplifier.

3. System Level Description

System Level Description (Block diagram + description of 1 to 2 pages).

(Make sure that your description justifies how the Requirements of your system are met, especially those which are not obvious.)

This chapter will describe the system main blocks, the functionality of each of them, and the interaction between each block and its adjacent modules. Presented below (Figure 3.1) is a graphical overview of the system and its first layer of modules. A high resolution version is also included in *Appendix A: System Overview*.

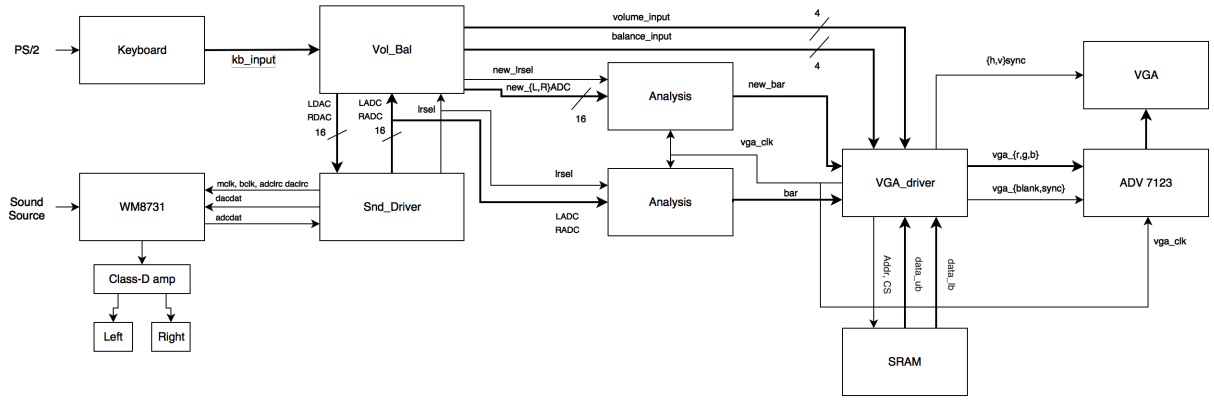


Figure 3.1: A graphical overview of the system's first module layer. Letters inside curly braces indicates multiple signals exlusively including each of the letters.

3.1 Keyboard

The user interacts with the system through a PS/2-connected keyboard. The keyboard is then handled by the module `Keyboard` which reads the scan codes, matches these against a *one hot encoded* preset which makes up the `kb_input` signal passed to `Vol_Bal`.

The module inputs are `PS2_DAT`, `PS2_CLK`, `clk` and `rstn` which are used to shift in the scan code and compare the result with the preset, resulting in `kb_input` — a 2-bit unsigned value indicating if either of the arrow keys have been released. `Vol_Bal` will then use this signal to adjust the volume and balance level. The Up/Down arrow keys controls the volume, and the Left/Right arrow keys controls the stereo channel balance.

3.2 Snd_Driver

3.3 Vol_Bal

3.3.1 Vol_Bal:current_vol_bal

3.4 Analysis

3.5 VGA_driver

The `vga_drive` module exists to handle the rendering of a 640x480 resolution image on the screen. The image that is supposed to be rendered consists of a background image previously stored in the SRAM consisting of pre filled bars that within the module will be blanked out according to the input stimuli, which will give the appearance of bars being filled to different levels.

To render an image on the vga screen you need five main signals. Three analog color channels (red, green and blue) and two signals for synchronization hsync and vsync. The image is rendered pixel by pixel line by line using a horizontal sweep pattern which is reset by the two sync signals. If a color is set when the sweep resets arbitrary patterns can occur and therefore the signal has to be blanked during the reset phase.

The module `vga_drive` accomplishes this using eleven pipelined sub modules viewable in 3.2 it has N inputs: volume_input, balance_inputs

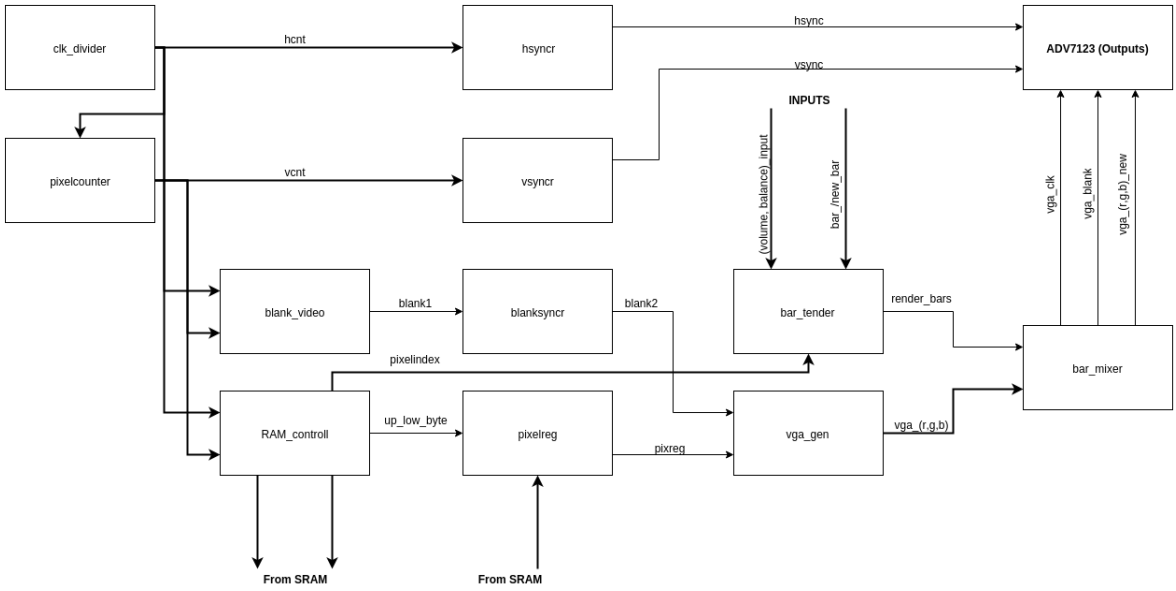


Figure 3.2: Block diagram of vga_drive

3.5.1 VGA_driver:bartender

3.5.2 VGA_driver:barmixer

4. Challenges in the Design and Proposed Approach

Challenges in the Design and Proposed Approach (Main 2 or 3 challenges and proposed solutions, 1/2 to 1 page per challenge).

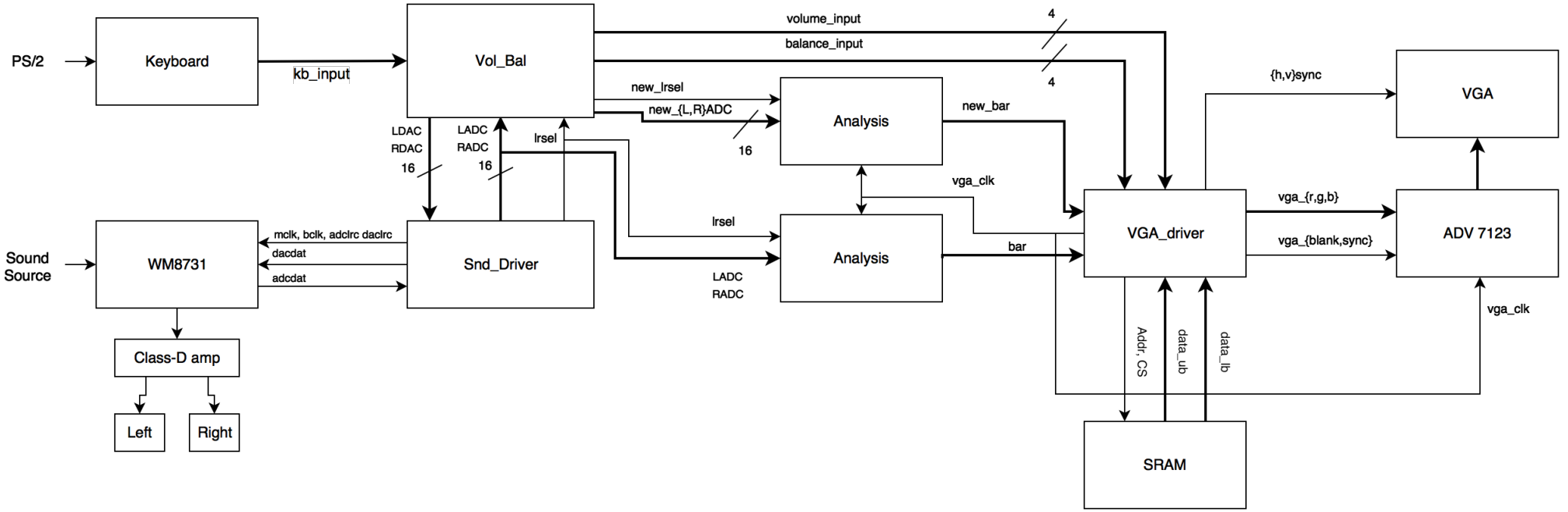
The immediate challenge the project is facing is the `Analysis` module. Since the update rate of the VGA screen is far much lower than the polling rate of the `Snd_Driver`, the analysis have to extract average of a set of samples.

Further, there might be a need for the user interface to be further evened out over a longer period of time for a more fluent visual experience.

5. User Interface

User Interface (How to control the system + image visualized on the screen).

A. Appendix: System Overview


PS/2
Keyboard
kb_input
Vol_Bal
volume_input
balance_input
4
4
new_lrsel
new_{L,R}ADC
16
Analysis
new_bar
{h,v}sync
VGA
LDAC
RDAC
16
LADC
RADC
16
lrsel
vga_clk
vga_{r,g,b}
vga_{blank,sync}
ADV 7123
Sound Source
WM8731
mclk, bclk, adclrc daclrc
dacdat
adcdat
Snd_Driver
lrsel
LADC
RADC
Analysis
bar
VGA_driver
vga_clk
Class-D amp
Left
Right
Addr, CS
data_ub
data_lb
SRAM