The University of Jordan

Faculty of Engineering

Mechatronics Engineering Department

VGA Control Using Rotary Encoder

Instructor : Dr. Musa Al-Yaman

Student's Name	University ID
Muhammad Al-Barham	0162708
Zeina abdelaziz	0156297
Muhammad Arslan	0161049
Saed Al-Nabelse	0154274
Mohannad Atiyeh	0155057

Table of Contents

List of Figures

List of Table

Introduction:

The purpose of project is to modify the VGA code by adding a rotary switch encoder to generate eight different colors on a monitor, where the output color depends on in which direction the rotary switch was rotated. The project is implemented using a Spartan-3E FPGA board, and a 640 by 480 resolution monitor screen, the code is done using Verilog coding language on Xilinx ISE IDE.

The video graphic array (VGA) interface is a standard interface that has been used widely in many applications, brought to the world by IBM back in the 1987. There are two main display modes that the VGA works with; “character display mode and graphics display mode” [1]. It also has three kinds of graphic display modes; the first mode is where CGA (color graphic adapter) and EGA (enhanced graphic adapter) are compatible, secondly the standard VGA graphics display mode, and the expanding VGA graphics display mode, which were named VGA graphics display mode. The VGA offers a wide range of resolutions and refreshing rates, such as; a resolution of 400×600 with 50 Hz refreshing rate, and 360×480 with 60 Hz refreshing rate.

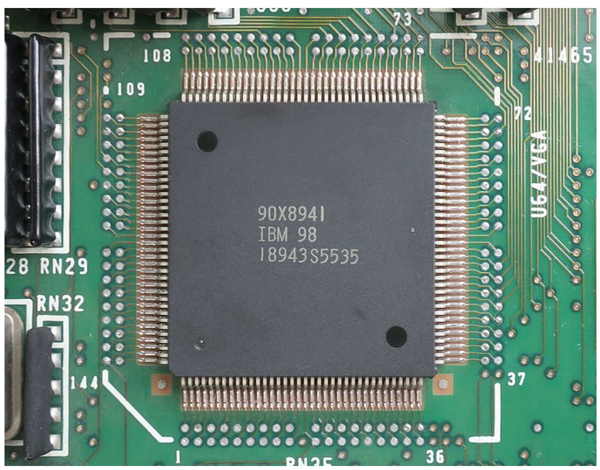

Figure 1: The IBM 90x8941 VGA chip [3]

Lately, in most computers, data is being transported through the analog signals VGA interface to the exterior display equipment, where it will be firstly transformed into tricolor signals (red, green and blue); known as the VGA data signals, and into “horizontal, vertical synchronization signals through D/A switch on the display card” [1]; known as the VGA control signals, from its digital graphic format, and then it will be showed using display equipment.

“The Spartan®-3E FPGA Starter Kit board includes a VGA display port via a DB15 connector” [2]. This port can be directly connected to PC monitors, LCDs, using a standard monitor cable, such as; HD 15.

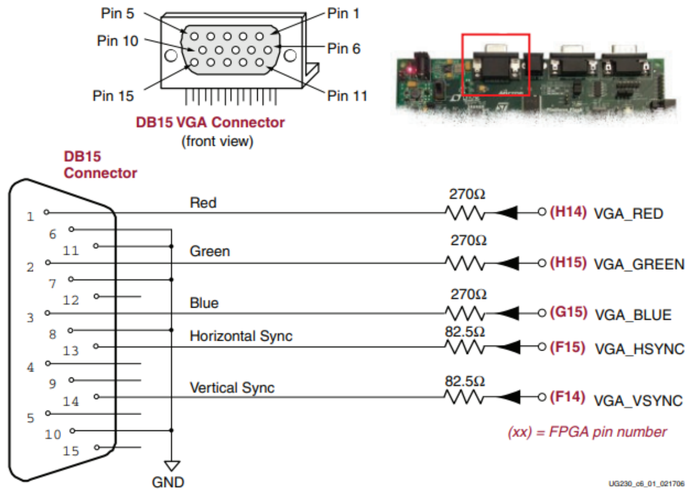

Figure 2: VGA connections from Spartan-3E starter kit board [2]

The Spartan-3E FPGA Starter Kit board also offers different types of switches, push buttons, and a rotary push button switch, that is used in the implementation of this project. This rotary push-button switch is located in between four individual push-button switches, as seen in the following figure:

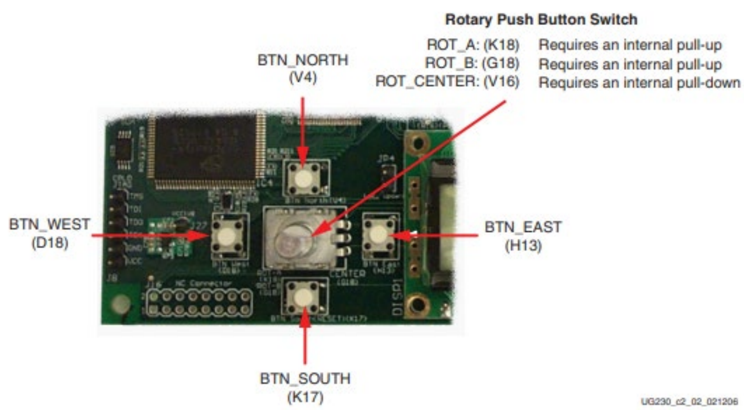

Figure 3: Four push-button switches surround rotary push-button switch [2]

This switch has three outputs, the shaft encoder outputs, ROT_A, and ROT_B, as well as the center push-button, ROT_CENTER. The switch can be rotated to either turn on ROT_A, or ROT_B, and it can be pressed to produce ROT_CENTER, acting as a simple push-button. In this project, the focus is rotating the switch to change the colors on the screen, using ROT_A, and ROT_B.

Block Diagrams and ASM charts:

This section provides a description of how the system works, and ASM charts of the modules, block diagrams of each module, and results of the behavioral simulation.

Block diagrams:

Mechanical switches and push buttons are prone to bounce, the rotary switch in the Spartan-3E kit also is, which is described by chattering. A switch's chatter is an interrupt of an extra rotation (click) in either directions of rotation. This problem has been solved through programming. To eliminate this effect completely, only the first change of signal is detected as seen in the following truth table:

Table 1: Truth table of the rotary switch filter

Rotary_A	Rotary_B	Rotary_q1	Rotary_q2
0	0	0	No change
0	1	No change	1
1	0	No change	0
1	1	1	No change

Where Rotary_A, and Rotary_B represent the two states of the rotary switch, and Rotary_q1 indicates the change from low to high, and Rotary_q2 represents the direction of rotation. This can be achieved using simple programming and implemented using D-flip flops. The block diagram of the module is as seen below:

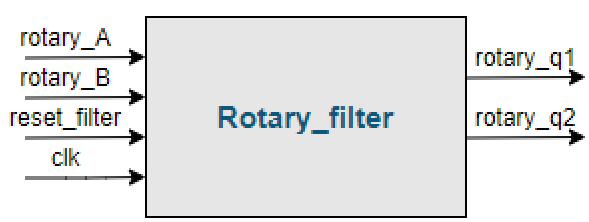

Figure 4: Block diagram of the rotary filter

After implementing the filter, the direction of rotation is specified through a different module that uses the filter as an instance inside it. The main idea is that if rotated to the right then Rotary_A takes the value of high, and Rotary_q2 takes the value of low, giving the shift_right register the value of 1 , on the other hand, if rotated to the left then Rotary_B takes the value of high, as well as Rotary_q2, giving the shift_left register the value of 1, but for the registers to take action it has to go through the sequence shown in figure 6, figure 5 represents the block diagram of the module:

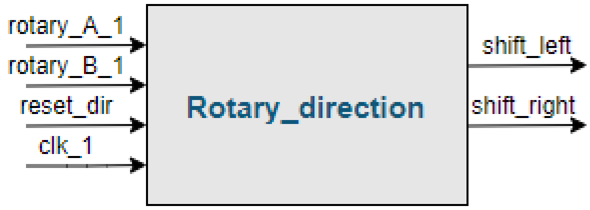

Figure 5: the direction of the rotation block diagram

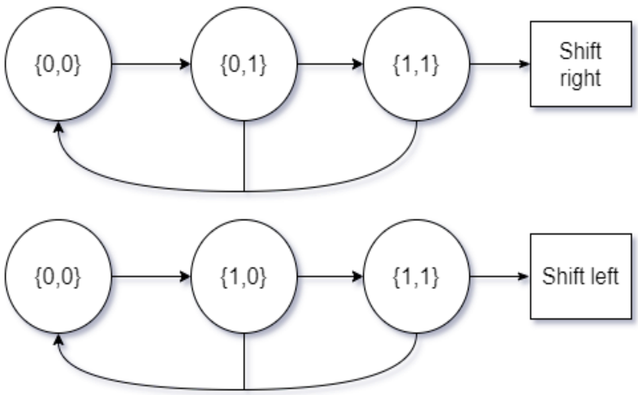

Figure 6: Right and Left Sequence Gray Code

The main module which combines the VGA code with the rotary switch to control the pixel generator is as shown in the figure below:

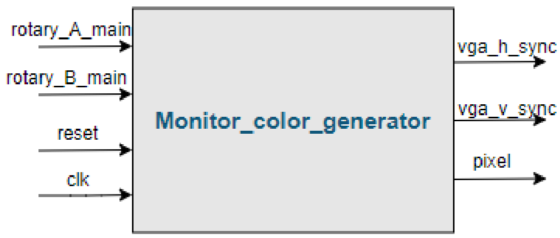

Figure 7: Main module block diagram

After specifying in which direction the rotary switch was turned, and producing the shift_left and shift_right registers, this is used to implement two different counters, one to count how many times the switch was rotated to the right, and the other counts how many times it was rotated to the left, then compared, and the output is taken and saved in the pixels, where it generates the different eight colors on the monitor.

The full block diagram of the module with all connections is:

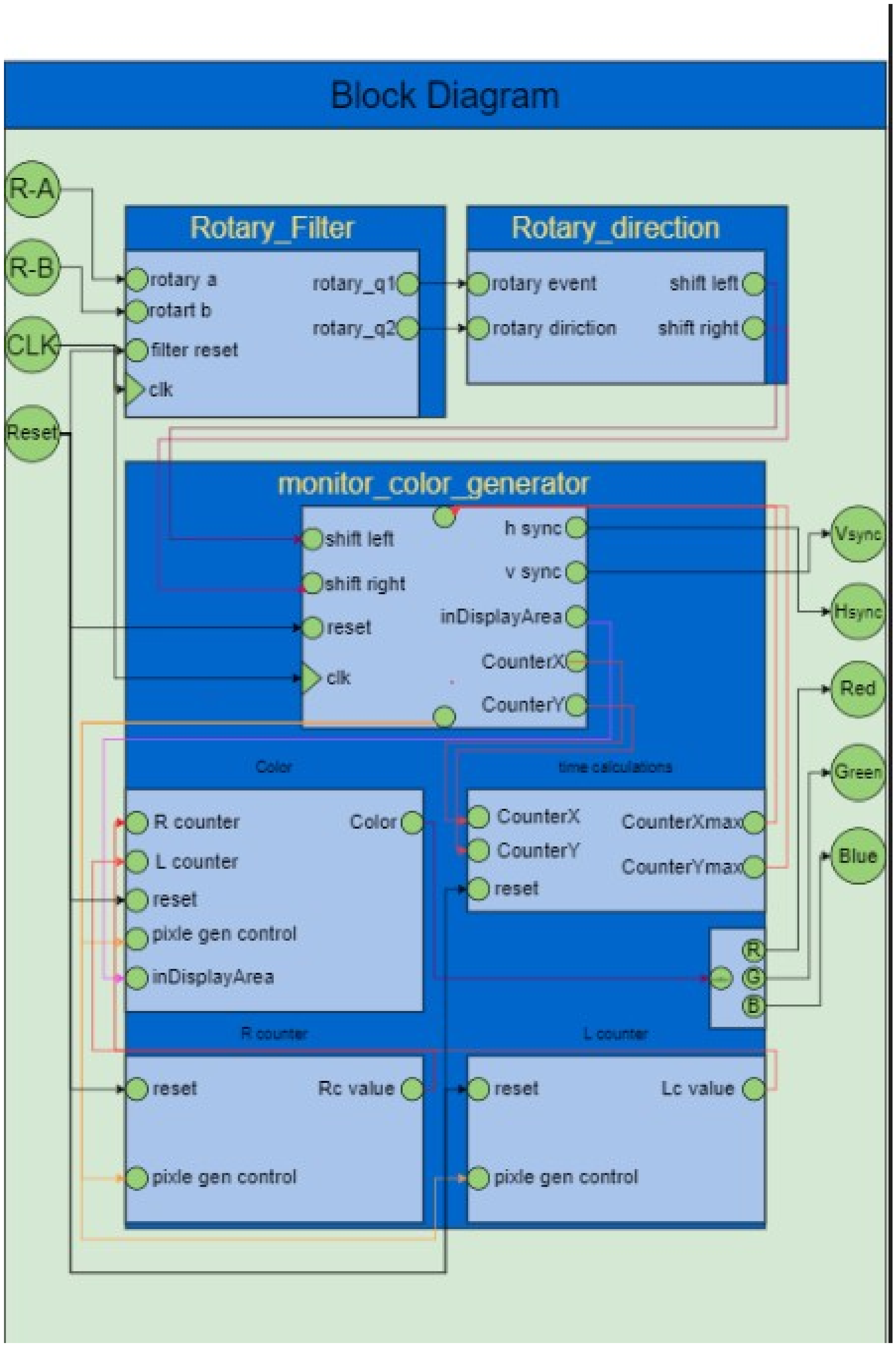

Figure 8: Block diagram of all connections of the main module

ASM charts:

Tracing down the behavior of generating colors on the monitor through rotating the rotary switch, the achieved ASM chart of the implemented module is in the figure below:

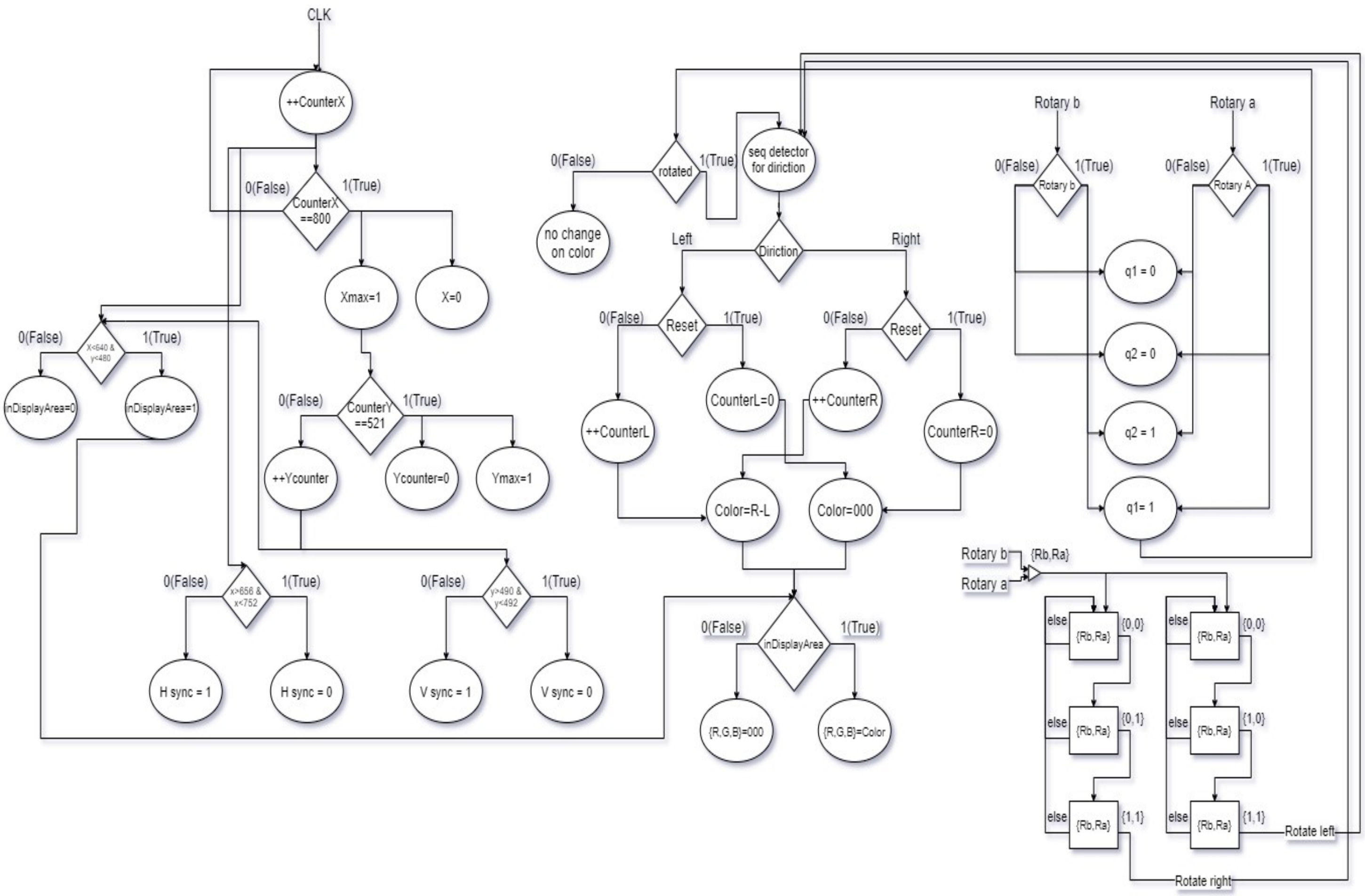

Figure 9: ASM chart of monitor generator

VGA IBM PS/2 Protocol:

This protocol is used for the 640×480 resolution characteristic of the VGA hardware.

Today, the VGA analog interface is used for high-definition video, including resolutions of 1080p and higher , While the transmission bandwidth of VGA is high enough to support even higher resolution playback

The 640×480 16-color and 320×200 256-color modes had fully redefinable palettes, with each entry selectable from within an 18-bit (262,144-color) RGB table, although the high resolution mode is most commonly familiar from its use with a fixed palette under Microsoft Windows. The other color modes defaulted to standard EGA or CGA compatible palettes (including the ability for programs to redefine the 16-color EGA palette from a master 64-color table), but could still be redefined if desired using VGA-specific programming.

Standard text modes

- 80×43 or 80×50 (8×8 font grid) in 16-colors, with an effective resolution of 640×344 (EGA-compatible) or 640×400 pixels.

Specifications

The original VGA specifications are as follows (supporting e.g. no hardware sprites):

- 256 KB Video RAM (The very first cards could be ordered with 64 KB or 128 KB of RAM, at the cost of losing some or all high-resolution 16-color modes.)

- 16-color and 256-color palette display modes.
- 262,144-color global palette (6 bits, and therefore 64 possible levels, for each of the red, green, and blue channels via the RAMDAC)
- Selectable 25.175 MHZ or 28.322 MHz master pixel clock
- Usual line rate fixed at 31.46875 kHz
- Maximum of 640 horizontal pixels
- Maximum of 480 lines
- Refresh rates at up to 70 Hz
- Vertical blank interrupt (Not all clone cards support this.)
- Planar mode: up to 16 colors (4 bit planes)
- Packed-pixel mode: 256 colors (Mode 13h)
- Hardware smooth scrolling support
- No Blitter, but supports very fast data transfers via "VGA latch" registers
- Some "Raster Ops" support
- Barrel shifter
- Split screen support
- 0.7 V peak-to-peak
- 75 ohm double-terminated impedance (18.7 mA, 13 mW)

Signal timings of VGA

All derived VGA timings (i.e. those which still use the master 25.175 and 28.322 MHz crystals and, to a lesser extent, the nominal 31.469 kHz line rate) can be varied widely by software that bypasses the VGA firmware interface and communicates directly with the VGA hardware, as many MS-DOS based games did. However, only the standard modes, or modes that at least use almost exactly the same H-sync and V-sync timings as one of the standard modes, can be expected to work with the original late-1980s and early-1990s VGA monitors. The use of other timings may in fact damage such monitors and thus was usually avoided by software publishers. Third-party "multisync" CRT monitors were usually much more flexible, and in combination with "super EGA", VGA, and later SVGA graphics cards using extended modes, could display a much wider range of resolutions and refresh rates at wholly arbitrary sync frequencies and pixel clock rates (within a particular lower/upper range, depending on model, typically encompassing a range spanning CGA's 15.7 kHz to SVGA and XGA's 36 kHz at a minimum, and including the 18.4, 21.8, 24.8 and 31.5 kHz of MDA/Hercules, EGA, "25kHz medium resolution" (a standard popularised by several Japanese domestic-market computers) and VGA along the way), commonly reaching 640x400 to 720x480 (at 56 to 72 Hz), 752x410, 800x560 and 800x600 as standard or near-standard settings.

Table 2 : Signal timings

Parameter	Value for model 320/640 pixel graphic	Unit
Pixel clock frequency	25.175	MHz
Horizontal frequency	31468.75	Hz
Horizontal pixels visible	640	
Horizontal sync polarity	negative	
Total time for each line	31.778	µs
Front porch (A)	0.636	µs
Sync pulse length (B)	3.813	µs
Back porch (C)	1.907	µs
Active video (D)	25.422	µs

Typical uses of selected modes

640×480 @ 60 Hz is the default Windows graphics mode (usually with 16 colors), up to Windows 2000. It remains an option in XP and later versions via the boot menu "low resolution video" option and per-application compatibility mode settings, despite Windows now defaulting to 1024x768 (and then only when running the default universal video driver and/or unable to detect the connected display's native resolution) and generally not allowing any resolution below 800x600 to be set. The need for such a low-quality, universally compatible fallback has diminished since the turn of the millennium, as VGA-signaling-standard screens or adaptors unable to show anything beyond the original resolutions have become increasingly rare, and are now most likely to be employed for specialist uses by enthusiasts - for example, arcade cabinet gaming PC conversions, or driving pico-projectors from wearable computers - instead of blithely connected up as a main desktop display by an unsuspecting casual user; however, as it retains such niche utility, the ability to deliberately set it (and indexed 16 or 256 color modes besides, instead of now-normal 16/24/32-bit direct color), nowadays as a special rather than default mode, has been preserved.

Table 3 : VGA connector (DE-15/HD-15)

A female DE15 socket		
Pin 1	RED	Red video
Pin 2	GREEN	Green video
Pin 3	BLUE	Blue video
Pin 4	ID2/RES	Reserved since E-DDC, formerly monitor id. bit 2
Pin 5	GND	Ground (HSync)
Pin 6	RED_RTN	Red return
Pin 7	GREEN_RTN	Green return
Pin 8	BLUE_RTN	Blue return
Pin 9	KEY/PWR	+5 V DC (powers EDID EEPROM chip on some monitors), formerly key
Pin 10	GND	Ground (VSync, DDC)
Pin 11	ID0/RES	Reserved since E-DDC, formerly monitor id. bit 0
Pin 12	ID1/SDA	I^2C data since DDC2, formerly monitor id. bit 1
Pin 13	HSync	Horizontal sync
Pin 14	VSync	Vertical sync
Pin 15	ID3/SCL	I^2C clock since DDC2, formerly monitor id. bit 3

Conclusion:

Modifying the VGA module in order to add a rotary switch available in the Spartan-3E started kit board is a simple job achieved by the understanding of the nature of the switch, the understanding of the VGA module and how it flows, and understanding how a pixel generator works. Block diagrams and ASM charts were developed to ease the job and get a better understanding of the system. One of the main qualities of programming using FPGA is it being flexible and doesn’t take as much space as other processors do. The VGA chip is easily controlled, with the disadvantage of it having “bad real-time ability” [1], but has different resolutions, at different refreshing rates. The showcased rotary switch has the disadvantage of chattering, known as bouncing, which most mechanical push-buttons are prone to, but can be solved through programming.

References:

[1] Wang, G., Guan, Y. and Zhang, Y., 2009, May. Designing of VGA character string display module based on FPGA. In *2009 International Symposium on Intelligent Ubiquitous Computing and Education* (pp. 499-502). IEEE. [Accessed 7 Aug. 2020].

[2] partan-3E FPGA Starter Kit Board User Guide UG230 (v1.2). (2011). [online] Available at: https://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf [Accessed 7 Aug. 2020].

[3] en.m.wikipedia.org. (n.d.). *Video Graphics Array - Wikipedia*. [online] Available at: https://en.m.wikipedia.org/wiki/Video_Graphics_Array?fbclid=IwAR0YWS0c5fZ0jkfMbKHJo23fS1FNWct9HMODq9qIZPxwMTxpHpQyWXwK7_E [Accessed 7 Aug. 2020].

[4] Chapman, K., 2006. Rotary encoder interface for Spartan-3E starter kit. *Application Note, Xilinx Ltd*.

[5] the self-generated visuals were done using: https://www.draw.io/