



VGA output using a 36-pin STM32

By Artekit(<https://www.artekit.eu/author/admin/>)

December 8, 2012(<https://www.artekit.eu/2012/12/08/>)

54 Comments(<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comments>)

Thinking about old video game consoles and arcade machines (very old, like those in the 70's/80's) it came to our minds what can be done today using very low-cost microprocessors. Generally, these microprocessors weren't even created to do this task, so the challenge began, and we started to think the way to output video to a screen with few or no external components at all.

We have picked a 36-pin, 72 MHz STM32 (STM32F103T8U6), fast enough to generate monochrome video synchronism and dot signals. We use a couple of timers and the SPI (this way the refresh of the frame buffer is done automatically). And the final result is a pretty decent monochrome VGA output with 400 x 200 dots resolution.

VGA output on a STM32 Cortex M3



**Click here to download the complete source code
(/resources/blog/artekit_vga.zip).**

The project is in KEIL uVision format. You can download the KEIL uVision evaluation version from www.keil.com (<http://www.keil.com>)

The list of materials:

- A board with a STM32F103T8U6 or similar. We use the **AK-STM32-LKIT** (</products/devboards/ak-stm32-lkit/>).
- A female VGA connector (DB15).

Even if the frame buffer is 400×200 pixels length, the output resolution is 800×600 at 56Hz. We will be painting every horizontal dot twice, and every line will be repeated 3 times. This way we fill the entire screen.

Another reason we have chosen 800×600 @ 56Hz is because of the pixel clock: this resolution uses a 36 MHz pixel clock, that is a multiple of 72Mhz, the frequency of the STM32. Since we will be generating the pixels signal with the SPI, we can divide the STM32

clock with the SPI prescaler to get a 18MHz pixel clock, and paint every pixel twice. The SPI MOSI line will stay high or low twice the time needed to output a single pixel for a 800 pixels horizontal resolution.



(<https://www.artekit.eu/wp-content/uploads/vga-screen.png>)

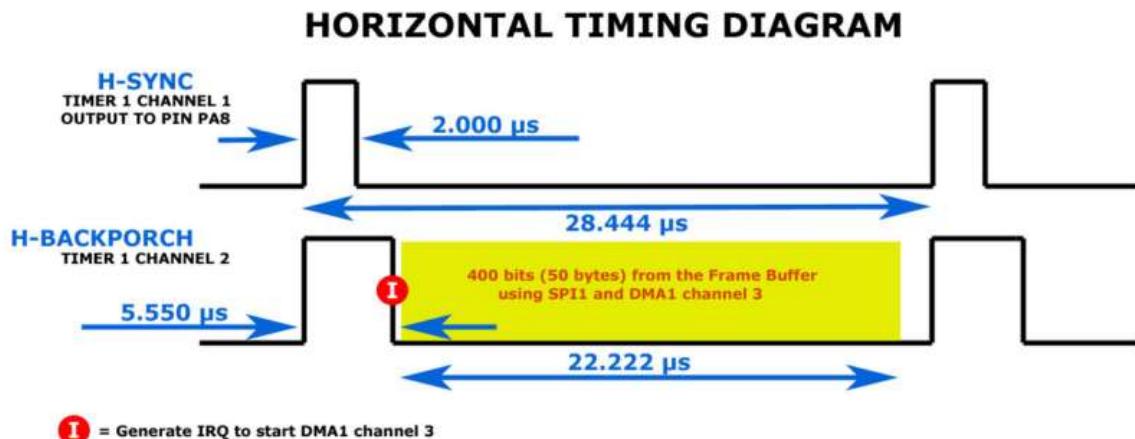
The frame buffer is composed of an array of 52×200 bytes. $50 \times 8 = 400$ pixels (every bit is a pixel). The two remaining bytes will simulate the blanking interval for every line.

```
1 #define VID_VSIZE 200
2 #define VID_HSIZE 50
3
4 __align(4) u8 fb[VID_VSIZE][VID_HSIZE+2];
```

Everything we write in this piece of RAM will be output directly to the screen without intervention of the application: the DMA is set to automatically read from the frame buffer and output the values to the SPI MOSI pin.

The horizontal synchronism

The horizontal synchronism signal and the back porch time are generated using the TIM1 timer, channels 1 and 2 (respectively). The TIM1 channel 1 is connected to the pin PA8.



(<https://www.artekit.eu/wp-content/uploads/VGA-H1.png>)

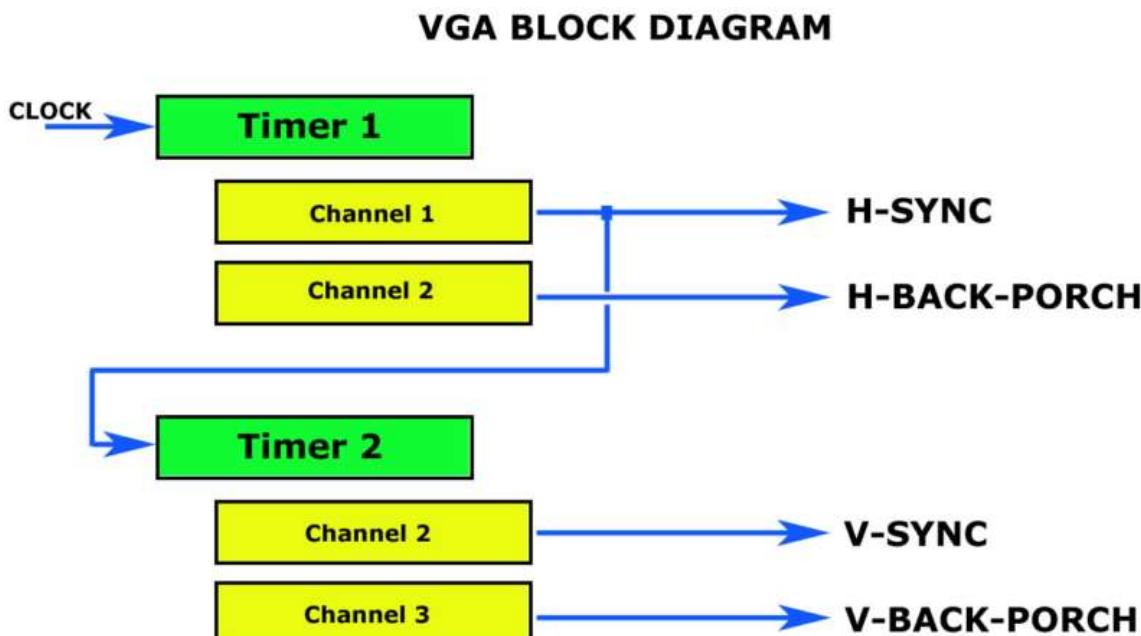
The H-SYNC timer 1 channel 1 (pin PA8) will actually generate the horizontal synchronism that the monitor will receive.

The H-BACKPORCH timer 1 channel 2 signal is calculated from the sum of the horizontal synchronism time and the back porch time. This timer will generate an interrupt that will be used to fire the DMA request to start sending pixels through the SPI.

This is repeated for every line in the frame buffer.

The vertical synchronism

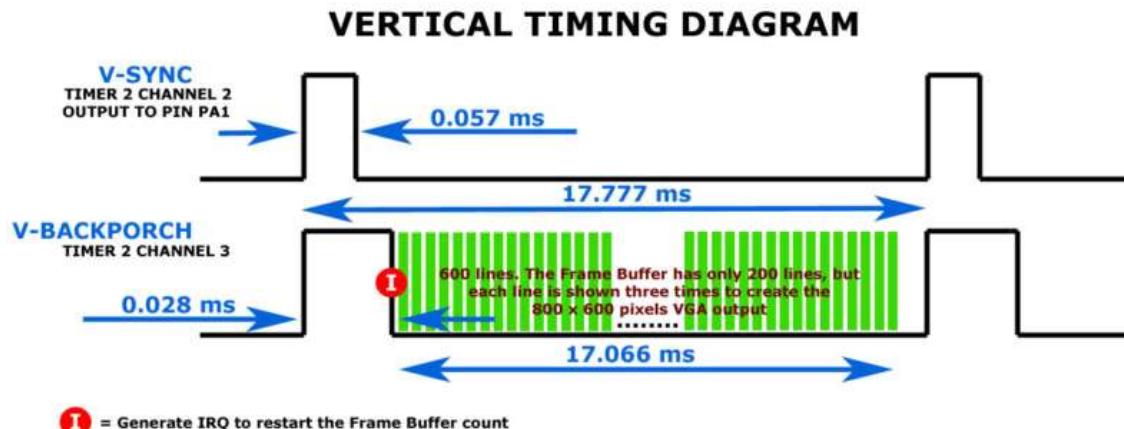
The vertical synchronism is generated using the TIM2 timer, but in slave mode. The TIM2 timer counts the H-SYNC pulses generated by its master, the TIM1 timer.



(<https://www.artekit.eu/wp-content/uploads/vga-timers.png>)

The TIM2 timer channel 2 outputs the V-SYNC pulse through pin PA1.

The TIM2 timer channel 3 will trigger an interrupt when the timer counter reaches the sum of the V-SYNC and vertical back porch time. This interrupt will set a variable indicating that the scanning is within a valid frame and the DMA can start sending pixels to the screen.



(<https://www.artekit.eu/wp-content/uploads/VGA-V.png>)

Pixel generation

Pixels are generated using the SPI MOSI pin (PA7). The timer TIM1 channel 2 generates an interrupt that will enable DMA TX requests to the SPI. The DMA will read a line from the frame buffer and will put the values in the SPI DR register.

The DMA is set to generate an interrupt after a single line is sent, where the line number is incremented. Since we are sending each line three times, we will increment a counter in this interrupt. When the three lines have been sent, we set the DMA pointer to the next line in the frame buffer.

When all the lines have been sent, the DMA cycle is disabled until the next valid frame interrupt (TIM2 channel 3).

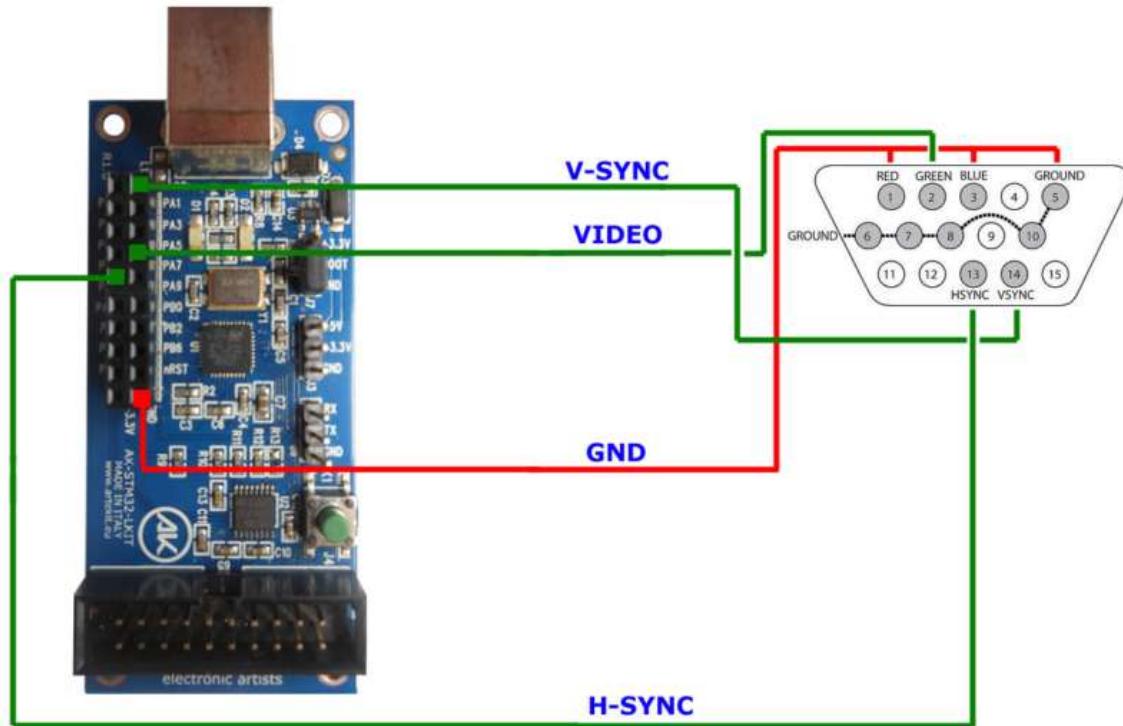
Connections

To use this example you will only need some wires and a female VGA connector.

The VGA standard says that the output signals should be 0,7V to 1V, so you may want to put a voltage divider in the pixel line (serial 68 ohm resistor and a 33 ohm resistor to ground, better with a 47pF in parallel with the 68 ohm resistor). We have tested a couple of LCD monitors without the divider and it went just fine.

Note that the pin layout is referred to the **AK-STM32-LKIT** ([/products/devboards/ak-stm32-lkit/](#)) expansion connector, but the pin names are valid for any STM32. Check your chosen STM32 datasheet to see if the timers and SPI pins matches the design.

Note: we use the green color (pin 2 of the VGA connector) to emulate the old style green phosphor monitors, but you can use another color combination using the RED/GREEN/BLUE DB15 pins. It is possible to create up to 8 color combination.



(<https://www.artekit.eu/wp-content/uploads/vga-connections.png>)

AK-STM32-LKIT pin	VGA connector pin	Description
PA1	Pin 14	Vertical sync
PA7	Pin 2	Green
PA8	Pin 13	Horizontal sync
GND	Pin 5	Ground

Conclusion

We have created a VGA controller using a very low cost microprocessor/development board. The method used it's certainly not the only way to do it, but this one uses no external components besides a VGA connector.

If you are using this example with a bigger STM32, you can try to use double buffering and to write to the frame buffer while the DMA is disabled, to avoid tearing.

You may **download the source code ([/resources/blog/artekit_vga.zip](#))** for this project. There you will also find a utility library to draw lines, points, circles, bitmaps, character generation, bit blit and more.

In the next blog entry we will be implementing a video game using this VGA example.

Have fun!

The Artekit Team.



www.artekit.eu/space-invaders-for-stm32/

54 Responses



Damien

(<https://sourcegate.wordpress.com/2012/11/04/code-to-produce-video-signals-on-the-stm32l-discovery/>)

says:

Did you have any trouble getting a stable image? I tried to make composite output from an STM32L1xx, (<https://sourcegate.wordpress.com/2012/11/04/code-to-produce-video-signals-on-the-stm32l-discovery/>) but the edges of the images wiggled around. I didn't look too much into it, but I'd like to know what causes it.

I notice that the Uzebox and the Arduino TV-Out has code to synchronize the program with a timer to avoid this problem which for the AVR is caused by interrupts being called only after the current

AVOID THIS PROBLEM, WHICH FOR THE AVK1 IS CAUSED BY INTERRUPTS BEING CALLED ONLY AFTER THE CURRENT

instruction runs. It doesn't look like you needed code to do this. This chip was only a 32MHz chip, so maybe your faster CPU was enough to hide this problem.

[Log in to Reply \(\[https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\]\(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)\)](#)



March 13, 2013 at 1:38 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-305>)

says:

Just at the very beginning. But then we realized the interrupt handlers were slow performing the task and was enough to ditch the ST library function calls and go directly to work with the timers and DMA registers. There's still some minor glitch sometimes, I think because we access the RAM when the DMA is fetching the current line.

Changing the "speed" (2, 10, 50 MHz) of the GPIO also played a role because of the slew rate produced. You can check with uVision or any IDE that allows you to change those values at runtime to see the difference.

One curious fact was that we had to move the interrupt handlers to a fixed position at the end of the flash because adding code before those functions (in terms of position in flash) made the SPI to start sooner or later, depending on the added code. I believe in some flash fetching delay depending on the function location.

Take in consideration that this project is about VGA and may not apply to your composite video project.

[Log in to Reply \(\[https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\]\(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)\)](#)



Bill Lewis

(<http://www.wrljet.com>) says:

March 28, 2013 at 8:41 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-349>)

The suggested voltage divider (serial 68 ohm resistor and a 33 ohm resistor to ground) will pull 33mA from the STM32 pins. Seems possibly too heavy a load to me.

[Log in to Reply \(\[https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\]\(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)\)](#)

**GipsyRaven**[\(http://www.techstuff.in.ua/\)](http://www.techstuff.in.ua/)

says:

good project, launched on my debug board with STM32F103RBT6 the link you can download the PCB
[\(http://www.techstuff.in.ua/forum/48-ARM/47-%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0%D1%8F-%D0%BF%D0%BB%D0%B0%D1%82%D0%B0-STM32F103RBT6.html\)](http://www.techstuff.in.ua/forum/48-ARM/47-%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0%D1%8F-%D0%BF%D0%BB%D0%B0%D1%82%D0%B0-STM32F103RBT6.html)

%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0%D1%8F-%D0%BF%D0%BB%D0%B0%D1%82%D0%B0-STM32F103RBT6.html
[\(http://www.techstuff.in.ua/forum/48-ARM/47-%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0%D1%8F-%D0%BF%D0%BB%D0%B0%D1%82%D0%B0-STM32F103RBT6.html\)](http://www.techstuff.in.ua/forum/48-ARM/47-%D0%9F%D0%B5%D1%87%D0%B0%D1%82%D0%BD%D0%B0%D1%8F-%D0%BF%D0%BB%D0%B0%D1%82%D0%B0-STM32F103RBT6.html)

based on the detailed design of your plan to make a UART terminal for RTOS

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**decoysmile**[\(https://vk.com/decoysmile\)](https://vk.com/decoysmile)

says:

hi

I have a problem in Keil uVision4 when the controller firmware bug

The console writes

Load "C:\Users\mihail\Desktop\artekit_vga\obj\stm32_vga.axf"

No Algorithm found for: 08000000H – 08001B23H

No Algorithm found for: 08009000H – 08009337H

Erase skipped!

help please

I use your project and your programm. but i have module a STM32vldiscovery. with microcontroller STM32f100RB

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**Artekit**

says:

June 15, 2013 at 11:34 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-506>)

You need a flashing algorithm for your MCU.

Right-click the project and select “Options for target...”. Then in the utilities tab click on your debugger “settings” and select (using the “add” button) a flashing algorithm for your MCU.

BTW, I see your MCU is a 24MHz Cortex M3. I don’t know if this project would run smooth on this speed.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Vincent (http://na)

August 23, 2013 at 1:40 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-710>)

says:

Good job, very nice project !

Love it !

Have you ever been thinking about using another SPI (or maybe 2 others) to add other colors red and/or blue ?

Although I don’t think you can run 2 SPI at full speed with DMA, the data-bus might be the bottleneck.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Artekit
says:

August 27, 2013 at 12:03 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-726>)

Thank you.

This STM32 has only 1 SPI. I think the problem would not be bandwidth, but RAM. Since we are using 10K from the 20K the STM has, just to store the frame buffer (using 1 bit color depth).

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

Pingback:

VGA output using a 36-pin STM32 | 이에스테크놀로지 (<http://www.estech.com/wordpress/?p=255>)

Pingback:

VGA Driver rate using GPIO

(<http://www.edaboard.com/thread303808.html#post1300843>)



Angel G.

(<http://digfor.net>) says:

February 15, 2014 at 8:35 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-1777>)

Impossible!

How did you manage to get SPI to join the databytes in scan-lines without sending inter-byte gaps/stop bits ?

Each SPI protocol diagram I found, shows that after a byte or a frame is sent, there is a **pause**. So either the diagrams I saw were incorrect or you did kind of a magic ? for example, I read:

http://www.eng.auburn.edu/~nelson/courses/elec5260_6260/SPI%20STM32.pdf

(http://www.eng.auburn.edu/~nelson/courses/elec5260_6260/SPI%20STM32.pdf)

Is it possible to create longer scanlines (1680 bits for example) ?

I plan to use STM32F4 with LQFP-100, because there are more SPI and UART modules.

[Log in to Reply](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F) (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/>)



Artekit
says:

February 15, 2014 at 10:04 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-1778>)

Hi.

It's possible. SPI is used with DMA so there are no gaps (if you mean gap as the consequence of , for example, a busy loop or interrupts happening in the middle of a transmission). Also, the STM32 is quite precise with timings.

Perhaps the pause you saw on diagrams is because the peripheral you are talking to needs a pause to process whatever command you have sent to it. Since we are emulating VGA, we produce the pauses we want for sync intervals (by disabling/enabling the SPI/DMA).

You can send as many bytes as you want, as long as you have enough memory to hold an entire frame buffer. You will need to check if the frequency of the SPI matches the frequencies needed for the resolution you want to show.

Try it yourself! There are people who made their own boards to try the example:

<http://mikrokontroler.pl/content/space-invaders-na-stm32-projekt-mini-konsoli-dla-oldschoolowych-graczy> (<http://mikrokontroler.pl/content/space-invaders-na-stm32-projekt-mini-konsoli-dla-oldschoolowych-graczy>)

projekt-mimimi-konsoli-dla-olasczowowyci-graczy)

<https://www.youtube.com/watch?v=9rbvH5T-Hw4> (<https://www.youtube.com/watch?v=9rbvH5T-Hw4>)

This one on an LCD: **<https://www.youtube.com/watch?v=iwMcIFJg3hQ> (<https://www.youtube.com/watch?v=iwMcIFJg3hQ>)**

Log in to Reply (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



luftek

December 6, 2014 at 9:37 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-3237>)

(<http://luftek.si>) says:

Hi Artekit,

THANK YOU for sharing you project with us.

Your project was a starting point to our project. I used your logic, rewritten whole code, expanded to 2 output SPI, synchronized them in slave mode using two more timers. One to be same as TIM1, to trigger 10Mhz timer for SPI CLK.

I did additional logic with input part and viola an adapter on a MCU.

https://www.youtube.com/watch?v=HGje7a6_1Jk (https://www.youtube.com/watch?v=HGje7a6_1Jk)

Best regards from Slovenia

Log in to Reply (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



JeremyBai

January 21, 2015 at 2:50 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-3630>)

says:

hi, I see the method of solving the pixel clock is to make SPI MOSI line stays high or low twice the time needed to output a single pixel for a 800 pixels horizontal resolution, but I read the code and can not find the operation to make SPI MOSI line stays high or low twice the time needed , Could you tell me how to make SPI MOSI line stays high or low twice the time needed?with code or something else? thanks very much!

Log in to Reply (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan
[\(https://www.artekit.eu\)](https://www.artekit.eu)

January 21, 2015 at 8:52 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-3634>)

says:

Hi,

There is no specific code for setting the pin high/low twice the time. The pixel clock for this 800×600 VGA is 36MHz, and we use the SPI at 18MHz, so the MOSI line stays high/low twice the time needed to draw a pixel.

The monitor interprets every pixel twice, hence drawing two pixels.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



JeremyBai
 says:

January 22, 2015 at 3:26 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-3637>)

I got it ! because the cycle of the SPI is twice of the pixel clock! Thank you~^^.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



phuong says: June 16, 2015 at 6:26 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5456>)

Thanks for your project. I have read the code many times and still don't know how we determine x,y in these function like :gdiDrawTextEx(i16 x, i16 y, pu8 ptext, u16 rop)where code show me how to determine x,y on screen to create pixel?.thanks alot

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan
[\(https://www.artekit.eu\)](https://www.artekit.eu)

June 16, 2015 at 11:32 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5457>)

says:

Hi! The fb variable points to an array of 50×200 bytes. Every bit in that array is a pixel. Whenever you write into this array it will be shown in the screen.

Functions like gdiDrawTextEx accept x and y with a value between VID_PIXELS_X and

VID_PIXELS_Y (see the video.h file)

VID_F1A13_1 (see the video on the

https://github.com/Artekit/arteikit_vga/blob/master/video.h
(https://github.com/Artekit/arteikit_vga/blob/master/video.h)).

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



phuong

June 20, 2015 at 3:48 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5498>)

says:

thank you very much.I've got it.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



BobLynas

June 30, 2015 at 10:01 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5635>)

says:

Hi, is the STM32F103T8U6 similar to the STM32F103T8C6 (C6) device ? as I have lots of them here to play with and would like to try your code out. Regards Bob

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan

(<https://www.artekit.eu>)

June 30, 2015 at 10:41 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5639>)

says:

That "C" may be about a different package (that I cannot find in the datasheet), but should be compatible because it's the same STM32F103x8 family.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Iran Espinoza

July 9, 2015 at 7:08 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5701>)

says:

Artekit Hello! first of all, let me congratulate you on your project, that is what I was looking for! previously did something similar with a pic 18F2550, but actually want to do with stm32f0 (discovery

card), which works 48 mhz, I just want to make some of the text. and in the second step, an OSD, a 640×480 VGA source. I'm currently using cubemx, and Keil uVision 4. Could you help please? I tried to follow your project settings, adapting to stm32f0, jumping from 72 MHz to 48 MHz, using cubemx. but there are parts I do not understand how to adapt them. in advance I appreciate your time.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪  **Ivan**
<https://www.artekit.eu>

July 9, 2015 at 8:26 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5703>)

says:

Hi. Thank you.

I don't know if it will work @ 48MHz. Usually max. SPI speed is Fpclk/2, and I think there are not 640×480 VGA resolutions with that pixel clock or a multiple of it.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪  **Iran Espinoza** says:

July 9, 2015 at 11:15 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5705>)

Sorry bad writting:

Thank you for your response Ivan, certainly I can tell you, do not worry, I did it before, and it worked! It was not necessary to be exact pixel clock, you can check here:

<https://www.youtube.com/watch?v=fp5e6lgMYG0> (<https://www.youtube.com/watch?v=fp5e6lgMYG0>), the 1:09 minute, ES 640×480, it was with a pic 18F2550, but not in C. the problem is that someone stole my laptop and my projects, so the jump to arm. What do you think?

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪  **Ivan**
<https://www.artekit.eu>
says:

July 10, 2015 at 12:44 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-5706>)

Well, I guess with lower resolutions you can be more permissive with timings. You'll lose some quality though. Create a thread in the support forum and we'll discuss over there. Tell us what parts you don't understand.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Nard says: May 29, 2016 at 2:48 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-8838>)

Hi, would it be possible to make a LoRes color display (say 320x320 or 320x160) and would it be possible with the cheap STM32F103C8T6 (<\$4 on ebay)

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



→ **Ivan**
(<https://www.artekit.eu>)

May 29, 2016 at 4:48 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-8840>)

says:

Hi.

It's possible to drive an LCD using similar timings but I can't answer without doing some research first, which is what I suggest you to do.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Thiago Kuerten January 30, 2017 at 12:10 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10694>)

says:

Hello.

I'm doing a similar project to show a digital clock in a monitor using VGA connection. My ARM is STM32F103RB, and I adapted your code to configure the clocks in VGA in a 800x600 at 36 MHz. Now, I have to learn how to draw in the screen.

However, I didn't understand two things:

- 1) Why are you using Systick functions?
- 2) Where is located (in video.c) the command that enables to draw in the screen? I'm not seeing the connections between demoInit() and vidlInit().

Can you help me? Thanks.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪  **Ivan** [January 30, 2017 at 1:27 pm \(https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10698\)](https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10698)
[https://www.artekit.eu\)](https://www.artekit.eu)
says:

Hi Thiago,

- 1) for delays like the sysDelayMs() function, for example
- 2) there are not commands to enable drawing. Once vidInit() is called, you can draw in the screen by writing to the frame buffer (the fb variable). There are some helper functions to draw text, lines, circles, etc. You can find them in gdi.h.

demolInit() is the main loop for the demo you can see in the video.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪  **Thiago Kuerten** [January 30, 2017 at 11:46 pm \(https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10712\)](https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10712)
says:

Ivan,

What locations of your project that I can change to configure a 800 x 600 resolution? Just changing:

```
#define VID_HSIZE 50 // Horizontal resolution (in bytes)
#define VID_VSIZE 200 // Vertical resolution (in lines)
```

to:

```
#define VID_HSIZE 100 // Horizontal resolution (in bytes)
#define VID_VSIZE 600 // Vertical resolution (in lines)
```

results in this error:

```
.... ex1.elf section `.co_stack' will not fit in region `ram'
.... region ram overflowed with stack
[cc] collect2.exe: error: Id returned 1 exit status
```

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan

(<https://www.artekit.e>

January 31, 2017 at 12:10 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10713>)

u) says:

You can't use those values. 600×100 turns into 60KB of RAM. The STM32F103RB has only 20KB. The effective resolution of this project is 400×200 – the 800×600 output resolution is just to satisfy the VGA timing.

If you want a real 800×600 resolution you have to move to a bigger uC. Check the work of Karl Lunt using an STM32F4: <http://www.seanet.com/~karllunt/S4DVGA.html> (<http://www.seanet.com/~karllunt/S4DVGA.html>)

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Thiago Kuerten

January 31, 2017 at 2:45 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10714>)

says:

Thanks a lot! I won't realize with this important detail!

I have to use STM32F103RB, so I decided to use the same resolution as your source code. However, using the same parameters that you used to configure the VGA, and testing just the red color in the screen:

- 1) Drawing a full red screen by modifying vidClearScreen() setting `fb[y][x] = 127`, appears in my monitor this image: <https://www.dropbox.com/s/48uihd72xqud9cs/img1.jpg?dl=0> (<https://www.dropbox.com/s/48uihd72xqud9cs/img1.jpg?dl=0>). It has 52 black vertical lines, like VTOTAL value.

- 2) Drawing a rectangle by modifying vidClearScreen(), appears this image: <https://www.dropbox.com/s/js9ggx7lp82ef5c/img2.jpg?dl=0> (<https://www.dropbox.com/s/js9ggx7lp82ef5c/img2.jpg?dl=0>)

Do you have an idea of what's happening? If you like to see my code, contact me at my e-mail.

Excuse me for a lot of questions, and thanks a lot for always helping me!

NOTE: The code for the second (2) situation:

```
void vidClearScreen(void) {
    uint16_t x, y;
    for (y = 0; y < VID_VSIZE; y++)
    {
        for (x = 0; x < 69 && y < 20 && x < 40) {
            fb[y][x] = 127;
        }
    } else {
        fb[y][x] = 0;
    }
}
```

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

↪ **Ivan (https://www.artekit.eu)** [January 31, 2017 at 1:46 pm \(https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10719\)](https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-10719)



says:

Hi. Every bit on the frame buffer is a pixel. Please re-read the blog post. It states that "...The frame buffer is composed of an array of 52×200 bytes. $50 \times 8 = 400$ pixels (**every bit is a pixel**)...".

A value of $127 = 01111111_2$ will draw one blank column followed by 7 colored columns.
Use the gdiPoint() function to place pixels in the screen. It will do all the calculations for you.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ashin says: [April 21, 2017 at 6:15 am \(https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-11228\)](https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-11228)

Hi,

First of all , Great Project.

I wasn't able to display any lower case characters.

The lower case characters where missing and only the Upper case characters were displayed.

How do i display lower case characters?

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

→  **Ivan**
[\(https://www.artekit.eu\)](https://www.artekit.eu)

April 22, 2017 at 12:04 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-11235>)

says:

Hi,

Yes. I see the font is not complete (file font5x7.c).

You can complete it to show lower case characters, but you may be dealing with font descent and other issues that are beyond the scope of this project.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



sunil says: April 22, 2017 at 12:01 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-11234>)

How can I display a color image ?

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

→  **Ivan**
[\(https://www.artekit.eu\)](https://www.artekit.eu)

April 22, 2017 at 12:06 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-11236>)

says:

You can't, at least with the proposed method. It's monochrome.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Phil says: January 3, 2018 at 11:23 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12688>)

Thank you for your great work but depending on the code size compile I see jitter on the third or second raster line. I am confused. I have disabled all interups in my code and still does happen. Do

you have any idea. By the way I have ported your code to gcc tool chain.

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Phil says: January 4, 2018 at 5:09 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12689>)

I think I have resolved my issue by doing a slighth modification to your code:

I have moved the vdraw++ instruction at the beginning of the IRQHandler and now the timing seems to be spot on no matter what code I add... this is weird. I have not looked at the assembly insturctions though...

Phil

```
__irq void DMA1_Channel3_IRQHandler(void)
{
DMA1->IFCR = DMA1_IT_TC3;
DMA1_Channel3->CCR = 0x92;
DMA1_Channel3->CNDTR = VTOTAL;

vdraw++;
```

```
if (vdraw == 3)
{
vdraw = 0;

vline++;

if (vline == VID_VSIZE)
{
vdraw = vline = vflag = 0;
DMA1_Channel3->CMAR = (u32) &fb[0][0];
} else {
DMA1_Channel3->CMAR += VTOTAL;
}
}
```

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan
(<https://www.artekit.eu>)

January 4, 2018 at 5:37 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12690>)

says:

Hi,

It may depend on the optimization done by the different compiler options. I don't recall having glitches due to the differences in the generated code, but a look into the assembly instructions for every case may worth the effort

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Phil
says:

January 5, 2018 at 4:01 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12694>)

I just saw in one of your post:

"One curious fact was that we had to move the interrupt handlers to a fixed position at the end of the flash because adding code before those functions (in terms of position in flash) made the SPI to start sooner or later, depending on the added code. I believe in some flash fetching delay depending on the function location."

Could that be my issue?

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan
(<https://www.artekit.e>)

January 6, 2018 at 2:34 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12705>)

u) says:

It may be. But if I recall correctly, the behavior didn't change by changing optimization flags, that is your case. I have to say that we only tested the code compiling with KEIL only and not with GCC.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Rodrigo Corbera January 29, 2018 at 11:25 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12788>)
says:

Hello Ivan,

I really enjoyed your project a lot!

Thanks for sharing and for answering all the questions sent to you!

I've few questions:

- 1) There is some concurrency of CPU and DMA accessing RAM. Does it affect somehow CPU performance?
- 2) How different would be using DMA to memory GPIO instead of SPI? Would both work the same way and perform the same?
- 3) Please evaluate feasibility of this modification on your original code:
 - Reduce resolution to 200×150 making it 1/4 of 800×600 – pixel aspect 4:4 instead of 2:3
 - 200×150 is 3,750 bytes only. Now let's create 3 buffers of 3,750 bytes, for Red, Green and Blue colors. It's a total of 11,250 bytes, but now we have 8 colors in 3 bpp model (bits per pixel).
 - It's still full screen in VGA 800×600@56Hz mode
 - But now we must run DMA for 3 blocks from RAM to GPIO synchronously.

Thanks!

Best Regards,

Rodrigo Corbera

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



→ **Ivan** January 29, 2018 at 2:20 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12789>)
<https://www.artekit.eu>)

says:

Hi,

1) The bus is shared, so yes, there is concurrency. The bus matrix should prioritize CPU accesses.

Read http://www.st.com/resource/en/application_note/cd00160362.pdf
(http://www.st.com/resource/en/application_note/cd00160362.pdf)

2) It'll be very different as implementation. Never tried it. I'm not sure, but since the GPIO registers are accessed word-wide you'll need to dedicate an entire 16 pin port just to move N<=16 pins of the port with DMA.

3) I didn't do the math, but I think it's the wrong MCU for bit-banged RGB VGA. This is because:

a). the points mentioned in 2).

b). The GPIO peripheral has the ODR register, that would have to be written with a word containing the 3 pixels (you should use a timer interrupt to write 3 bits on a GPIO ODR. This will slow things a bit) so no DMA for this. And there are the BSRR and BRR registers, but you should know whether a pixel is a one or a zero before writing to it, so no DMA in this case either.

And, the functions that draw on screen would have to be modified to place bits in 3 different buffers.

[Log in to Reply \(\[https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\]\(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)\)](#)



Rodrigo Corbera January 29, 2018 at 6:38 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12795>)
says:

Thanks again Ivan for your clear and quick response.

As you said there is no DMA to GPIO pin port as a way of emulating a SPI or even sending just 8 bits to a GPIO port, thus this idea is dead.

I just looked at some STM32F103 datasheets and thought.... what about SDIO parallel 8 bits with DMA...

I don't have the specific knowledge about this MCU, but maybe you have and can tell me if that would be possible for achieving RGB output of RAM somehow.... at any divisor of 36MHz.

If SDIO with 8 bits out could render 256 colors in a resistor ladder, such as RRRGGBB for VGA. If Bluepill could send byte as a parallel 8 bits stream of 400 bytes, I could try to think about a tile system instead of Video RAM.

Please let me know,

Thanks again,

Rodrigo.

[Log in to Reply \(\[https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\]\(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)\)](#)



Ivan
(<https://www.artekit.eu>)

January 29, 2018 at 7:30 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-12796>)

says:

I don't think you can send whatever you want using SDIO as with SPI. It has an internal state machine and is command based, so, I hardly believe it can be done with SDIO.

As said before I don't think you can easily do RGB VGA with this particular MCU. Perhaps there is some trick to do it, but as the contrary of the current project, I doubt it would leave free computing power for whatever application it'll run. I would look into an STM32 with LCD controller and check if the timings can be adapted to those of a VGA. Perhaps you can select the color for every line (full R, G or B), by moving an external switch between line blanking times.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



nimagechouhai

March 28, 2018 at 4:30 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13134>)

says:

Hi, Ivan. When I apply your code for my board , it works successfully. But I can see double images on the screen. The whole screen seems to move a little distance towards right. My board's MCU is F103RBT6. Can you help me solve this problem? Looking forward your reply ! Thanks a lot !

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)



Ivan
(<https://www.artekit.eu>)

March 31, 2018 at 12:01 am (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13137>)

says:

Hi,

If you have not done any modification to the code, you can measure h-sync and v-sync signals with an oscilloscope and check if they respect the timings. Also check your quartz/resonator frequency, and if you are using something different to 8MHz, you'll need to adjust the initialization code.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**nimagechouhai**

says:

April 19, 2018 at 3:41 pm (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13193>)**Hi Ivan!**

Thanks a lot for your sharing and help. I still confuse about your code. You described that the resolution was 800*600@60Hz, but I find that only 400 pixels every line and 200 lines on the screen practically. Why isn't 800 pixels per line and 600 lines? Could you tell me why not. Looking forward your reply.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**Ivan**<https://www.artekit.eu>**April 21, 2018 at 10:36 am** (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13197>)

says:

Hi,

A framebuffer for 800×600 pixels (1 bit per pixel) is 60KB. The STM32 has only 20KB of RAM. So it's used a 400×200 framebuffer, and every pixel in a line is sent twice and every line is sent three times. Other reasons are described from the third paragraph.

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**Andrey Belykh****June 10, 2018 at 4:44 am** (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13370>)

says:

If anyone interested, I created a similar project that displays 256×192 with 64 colors.

<https://github.com/abelykh0/VGA-demo-on-bluepill> (<https://github.com/abelykh0/VGA-demo-on-bluepill>).

[Log in to Reply](#) (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**chydysz****June 15, 2018 at 5:10 pm** (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-13377>)

says:

Hi, should be the TIM1 stopped during V-SYNC pulse ? Is H-sync occurs during H-synch?

Sorry for my english i am still learning 😊

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

**Anonymous****April 25, 2021 at 11:56 am** (<https://www.artekit.eu/vga-output-using-a-36-pin-stm32/#comment-18350>)

says:

Hi Ivan,

MANY THANKS for sharing this great and clever project with us !! I'm planing to use it as a terminal for my 8bit hombrew computer.

1- Why the last 2 bytes are needed in fb[200][50 + 2], it works with only 50??

2- Can we run the SPI @32MHz? If yes, how to let the uControler read bytes from external 64KByte SRAM to get a true 800x600 resolution?

PS: please give your email

[Log in to Reply \(https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F\)](https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F)

Pingback:

Experiments in hard realtime - My Blog (<https://light-up.co.uk/blog/adventures-with-a-stm32/>)

Leave a Reply

You must be **logged in** (https://www.artekit.eu/my-account/?redirect_to=https%3A%2F%2Fwww.artekit.eu%2Fvga-output-using-a-36-pin-stm32%2F) to post a comment.



FAST SHIPPING WORLDWIDE

We usually ship in 24 hours with fast and economic couriers, and also with tracked and insured DHL Express.

From Italy to every country in the world.



FROM ENGINEERS TO ENGINEERS

We share the same passion for electronics and embedded systems with our customers.

We design and build our products just as we were the users.



FROM ITALY WITH LOVE

All our boards are designed and assembled in Italy and we carefully curate the entire sourcing, manufacturing and testing processes.

Artekit Labs is a growing Italian start-up created and builds electronics, breakout boards, development boards and tools for hobbyists, players, makers and professionals.



Forum(<https://forum.artekit.eu/>)

Blog(/blog/)

ABOUT

About Artekit Labs(/about-us/)

Privacy Policy(/privacy-policy/)

Online Dispute Resolution(/online-dispute-resolution/)

Terms of Use(/legal/)

Cookie Policy(/cookie-policy/)

Delivery and Payment FAQ(/delivery-and-payment/)

CONTACT US

Artekit Labs

Via Pacinotti 2/D

Porto Recanati

62017 MC, Italy

PI/VAT Number: IT02072930437

Numero REA: MC-281229