rfTextController64

© 2022-2024 Robert Finch

## Description

This is a text or tile mode video display controller that supports color intended for use with a 64 or 32-bit bus. The controller uses several internal dual ported r/w memories to store text, text attributes and character bitmaps. The display memory is sixty-four bits wide. Up to 8192 different simultaneous characters may be displayed along with 16-bit background and foreground colors (RGB565 format). The use of internal dual ported memories means that the text controller does not consume any memory bandwidth from the processor. The text controller may also be used as a tile graphics controller via the programmable character set.

The controller is programmable using only seven registers. Default values are established that should provide a reasonable display for 800x600 VGA mode.

The core respects byte lane selects, and partial updates of registers are possible. This makes it possible for the core to have an optional 32-bit bus slave interface.

The core has a 4kB device discovery black box (DDBB). The DDBB is used to indicate device presence and configure the I/O address the controller responds to.

|  |  |
| --- | --- |
| Address | Description |
| $00000 to $3FFFF | text screen and attribute memory area, currently the controller supports a 256kB memory max, enough for eight 80x50 screens |
| $40000 to $7FEFF | character bitmap memory, number of chars depends on char size |
| $80000 to $800FF | Text controller register area |

Text and Attribute Memory Layout

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 63 58 | 57 48 | 47 32 | 31 16 | 15 13 | 12 0 |
| Plane6 | ~ | Fore Color16 | Back Color16 | ~ | Char code13 |

## Clocks

The text video display controller uses two clocks, a bus timing clock (clk\_i) and a video timing clock (dot\_clk\_i), which can be completely independent.

The core synchronizes the display relative to externally supplied horizontal and vertical synchronization signals.

## Register Description

### Device Discovery Black Box (DDBB)

A 256-byte config space is supported. Most of the config space is unused. The only configuration is for the I/O address of the register set.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Regno | Width | R/W | Moniker | Description |  |  |
| 000 | 32 | RO | REG\_ID | Vendor and device ID |  |  |
| 004 | 32 | R/W |  |  |  |  |
| 008 | 32 | RO |  |  |  |  |
| 00C | 32 | R/W |  |  |  |  |
| 010 | 32 | R/W | REG\_BAR0 | Base Address Register |  |  |
| 014 | 32 | R/W | REG\_BAR1 | Base Address Register |  |  |
| 018 | 32 | R/W | REG\_BAR2 | Base Address Register |  |  |
| 01C | 32 | R/W | REG\_BAR3 | Base Address Register |  |  |
| 020 | 32 | R/W | REG\_BAR4 | Base Address Register |  |  |
| 024 | 32 | R/W | REG\_BAR5 | Base Address Register |  |  |
| 028 | 32 | R/W |  |  |  |  |
| 02C | 32 | RO |  | Subsystem ID |  |  |
| 030 | 32 | R/W |  | Extended ROM address |  |  |
| 034 | 32 | RO |  |  |  |  |
| 038 | 32 | R/W |  | Reserved |  |  |
| 03C | 32 | R/W |  | Interrupt |  |  |
| 040 to  0FF | 32 | R/W |  | Capabilities area |  |  |
| 100 to 1FF | 64 | R/W |  | Scratchpad RAM (256B) |  |  |
| 0200 to FFF | 64 | R |  | Boot ROM |  |  |

REG\_BAR0 defaults to $FEC00001 which is used to specify the address of the controller’s text video ram in the I/O address space. A 256kB region is reserved.

REG\_BAR1 defaults to $FEC40001 which is used to specify the address of the controller’s character bitmap memory in the I/O address space. A 256kB region is reserved.

REG\_BAR2 defaults to $FEC80001 which is used to specify the address of the controller’s registers in the I/O address space. A 256B region is reserved.

The controller will respond with a memory size request of 0MB (0xFFFFFFFF) when BAR0, BAR1, or BAR2 is written with all ones. The controller contains its own dedicated memory and does not require memory allocated from the system.

Parameters

CFG\_BUS defaults to zero

CFG\_DEVICE defaults to one

CFG\_FUNC defaults to zero

Config parameters must be set correctly. CFG device and vendors default to zero.

### Control Register Description

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Reg No. | Bit | R/W |  | Default Value |  |
| 00 | 7 – 0 | RW | Number of character columns | 64 |  |
| 15 - 8 | RW | Number of character row | 32 |  |
| 19 - 16 | RW | Character output delay | 7 |  |
| 31 – 20 | - | These bits are reserved |  |  |
| 43 – 32 | RW | Window left | 3956 (-140) |  |
| 47-44 | - | These bits are reserved |  |  |
| 59-48 | RW | Window top | 4058 (-38) |  |
| 63 – 60 | - | These bits are reserved |  |  |
| 08 | 4 – 0 | RW | Character height in pixels -1 Height < 33 | 17 |  |
| 7 - 5 | - | Reserved | - |  |
| 11 – 8 | RW | Pixel size width -1 (dot clocks) | 0 |  |
| 15 – 12 | RW | Pixel size height -1 (scan lines) | 0 |  |
| 20-16 | RW | Character width in pixels -1 Width must be even < 33 and > 5 for proper operation. | 11 |  |
| 23 – 21 | - | Reserved | - |  |
| 24 | RW | Reset state (auto resets to zero) | 0 |  |
| 32 | RW | Controller enable | 1 |  |
| 40 | RW | Multi-color mode enable | 0 |  |
| 52 – 48 | RW | Y scroll | 0 |  |
| 60 – 56 | RW | X scroll | 0 |  |
| Other | - | reserved | - |  |
| 10 | 30 – 0 | RW | Color for transparent color ZRGB 4-9-9-9 | 511 |  |
| 62 – 32 | RW | Border color ZRGB 4-9-9-9 | FFBF2020h |  |
| 18 | 30 – 0 | RW | Tile color 1 (multi-color mode) | 0 |  |
| 62 - 32 | RW | Tile color 2 (multi-color mode) | 0 |  |
| 20 | 4 – 0 | RW | Cursor end | 31 |  |
| 7 – 5 | RW | Blink Control  |  |  | | --- | --- | | 00 | No blink | | 01 | No display | | 10 | 1/16 field rate | | 11 | 1/32 field rate | | 7 |  |
| 12 - 8 | RW | Cursor start | 0 |  |
| 15 - 14 | RW | Cursor image type | 0 |  |
| 47 - 32 | RW | Cursor location in memory | 3 |  |
| Other | - | reserved |  |  |
| 28 | 15 – 0 | RW | start address – index into display memory | 0 |  |
| Other | - | reserved |  |  |
| 30 | 15 – 0 | RW | Font address in char bitmap memory | 0 |  |
| 63 -32 | RW | “LOCK” or “UNLK” font locking | “LOCK” |  |
| Other | - | reserved |  |  |

## Graphics

The core may be used as a low-resolution graphics controller via the programmable character set. The characters can be programmed for block graphics. For instance, each character could be a two-by-two grid of pixels. Sixteen different characters would be required to represent all the different combinations. It is also possible to program characters to a three-by-three grid of pixels using 512 programmable characters to represent every possible combination of on/off pixels. The default resolution is 64x32 or (768x576 pixels).

Graphics and text may be intermixed by allocating part of the programmable character set for a graphic array. For instance, using 256 programmable characters a 128x128 bitmapped display can be created.

## Fonts

Multiple fonts can be loaded into the character bitmap memory. The controller supports a 64kB font memory. Which font is selected is determined by the contents of the font address register. The font memory may be locked so that it is not inadvertently changed by an errant program. The number of character glyphs that may be stored depends on the size of characters. An 8x8 glyph will use eight bytes of memory, meaning 8192 different characters can be supported. The default pre-loaded font is 12x18 requiring 36 bytes of memory for each character, therefore only 1820 characters of this size can be supported. Fonts with character glyphs up to 64x64 pixels can be used. Horizontally, glyphs are blocked into a size of 8,16,32,or 64 bits. Vertically, glyphs are a multiple of the horizontal size. A 47x56 glyph must be mapped into a 64x56 array of bytes.

The character width of a font must be an even number between six and thirty-two. Character bitmaps are stored contiguously in memory with no wasted space.

## Multi-color Mode

If multi-color mode is enabled, pixels are combined into pairs to select one of four colors, the foreground color, background color, tile color 1 or tile color2. Each character or tile may then display pixels in one of the four colors.

## Output Planes

A four-bit output plane number may be supplied as part of the character attributes. The plane number controls the display priority when multiple video display devices are present in the video pipeline. Higher numbered planes will appear in front of lower numbered ones.

## Smooth Scrolling

Scrolling the screen in a smooth fashion is supported with the x and y scroll registers which allow the screen to be scrolled pixel by pixel.

## Power On Screen Randomizer

The controller features an automatic screen randomizer that causes random characters to be displayed when the controller is reset. Video display memory is loaded with random values. This is a visual aid that the controller is working properly.

## Display Input / Output Bus

The controller inputs 32-bit ZRGB data and outputs 32-bit ZRGB video data. ZRGB is RGB data with plane number indicator bits tacked on. Two bits are reserved for the plane number and 10 bits are reserved for each RGB color component.

## Core Parameters

The amount of memory used for the controller depends on the number of text cells supported. The default is 8192 cells or 64kB of memory. A maximum of 32768 cells or 256kB of memory is supported. An 80x50 text screen would use 4,000 cells.

|  |  |  |
| --- | --- | --- |
| Name | Default Value | Description |
| COLS | 64 | default number of columns of text |
| ROWS | 32 | default number of rows of text |
| BUSWID | 64 | Slave bus width may be 64 or 32 |
| TEXT\_CELL\_COUNT | 8192 | Number of supported text cells. Power of 2 |
| CFG\_BUS | 8 | config bus number (default 0) |
| CFG\_DEVICE | 5 | config device number (default 1) |
| CFG\_FUNC | 3 | config function number (default 0) |
| CFG\_VENDORID | 16 | Default 0 |
| CFG\_DEVICEID | 16 | Default 0 |
| INTERNAL\_SYNCGEN | 1 | Use internal sync generator |

## Internal Sync Generator

The internal sync generator has its own set of parameters for generating sync signals. These are defined in the core with default generation of VGA 800x600 mode. To use a different display mode parameters must be modified appropriately. (phTotal, pvTotal, etc).

## Module Interface Description

rfTextController

module rfTextController(rst\_i, clk\_i, rst\_busy\_o, config\_cs\_i, req\_i, resp\_o, dot\_clk\_i, hsync\_i, vsync\_i, blank\_i, border\_i, zrgb\_i, zrgb\_o, xonoff\_i);

|  |  |
| --- | --- |
| **System** | **Description** |
| rst\_i | This signal is normally connected to the system reset signal. It resets the text controller interface forcing it to the reset state. |
| clk\_i | This is usually connected to the system clock and is used as a base timing clock for I/O operations. |
| rst\_busy\_o | Indicates the controller is currently performing a reset operation. The reset last for 10 video frames during which the screen is randomized. |
| Slave Port | |
| cs\_config\_i | circuit select input – active for the DDBB config space (256MB) |
| req\_i | This is the FTA request bus. Please see the FTA bus documentation for details. |
| resp\_o | This is the FTA response bus. Please see the FTA bus documentation for details. |
| Video Ports | |
| dot\_clk\_i | This input is the video clock input. Pixel timing is derived from it. |
| hsync\_i | Horizontal sync. This input signal signals the start/end of a video scanline (end-of-line). This input is ignored if internal sync generation is used. |
| vsync\_i | Vertical sync. This input signal indicates the end of the video frame. This input is ignored if internal sync generation is used. |
| blank\_i | This input signal indicates that the display should be blanked. It is active during the video blanking period. This input is ignored if internal sync generation is used. |
| border\_i | This input signal indicates that a border area is active. This input is ignored if internal sync generation is used. |
| zrgb\_i | This 40-bit input bus can be connected to an external RGB input. (The text controller may display on top of the external input). |
| zrgb\_o | This output signal bus contains the 40-bit RGB display data. |
| hsync\_o | Horizontal sync output. Will be the same as the input if internal sync generator is not used. Otherwise, will be generated by the core. |
| vsync\_o | Vertical sync output. Will be the same as the input if internal sync generator is not used. Otherwise, will be generated by the core. |
| blank\_o | Blank video output. Indicates to blank the video output. Will be the same as the input if internal sync generator is not used. Otherwise, will be generated by the core. |
| border\_o | This output signal indicates that a border area is active. Will be the same as the input if internal sync generator is not used. Otherwise, will be generated by the core. |