

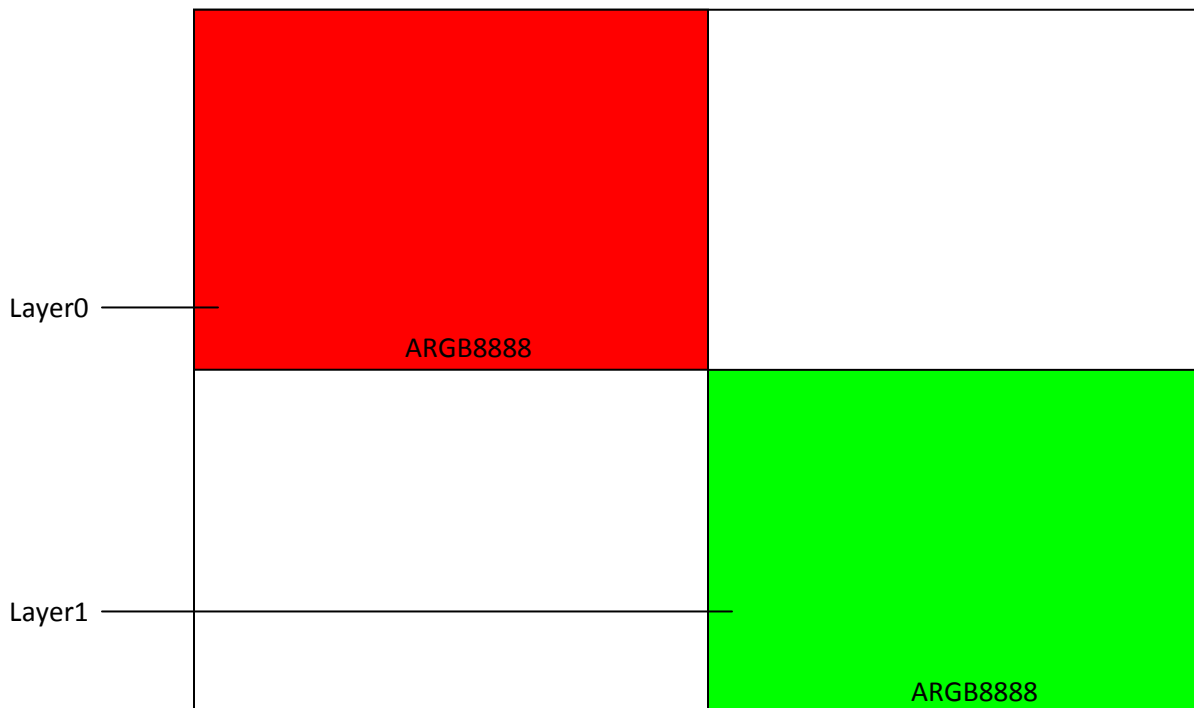
# Freescal MQX RTOS Example Guide

## FBDEV example

This document explains the FBDEV example, what to expect from the example and a brief introduction to the API used.

### The example

FBDEV example can run on svf522revb\_a5 and twrvf65gs10\_a5 board with DDR support. Since it requires big memory block to hold display buffer, IntRam version will fail in running. The example will show two rectangles on screen, each rectangle is controlled by a frame buffer device. Both frame buffer devices are double buffered, FBDEV0 occupies DCU4 layer 0, and locates at up-left of the screen. FBDEV1 occupies DCU4 layer 1, and locates at bottom-right of the screen. Both frame buffer devices contains ARGB8888 data. FBDEV0 has red color in front buffer and green color in back buffer. FBDEV1 has green color in front buffer and red color in back buffer. Firstly the example will show FBDEV0 for 3 seconds, and then FBDEV1 appears for 3 seconds. Next the frame buffers begin swapping, and the user could see yellow (red + green) blinking rectangles on both up-left and bottom-right areas. After 750 frames are swapped, the up-left and bottom-right rectangles will disappear in sequence.



### Running the example

The `BSPCFG_ENABLE_FBDEV` macro must be set to non-zero in the `user_config.h` file prior to compilation of MQX libraries and the example itself.

To run the example the corresponding IDE, compiler, debugger and a terminal program are needed.

## **Explaining the example**

This example demonstrates the usage of FBDEV driver to create simple graphic patterns - rectangles - in memory and to display them over the LCD hardware. The example employs only one task called `main_task` and it does following jobs.

- Open the FBDEV instance 0 and 1 and check the default parameters with command `IO_IOCTL_FBDEV_GET_BUFFER_INFO`.
- Reconfigure the frame buffer devices with command `IO_IOCTL_FBDEV_SET_BUFFER_INFO`, to set double buffered ARGB8888 format, and allocate the buffers and fill in colors.
- Call `IO_IOCTL_FBDEV_ENABLE` to show `fbdev0` and `fbdev1` regions in sequence.
- Now begin swapping frame buffers. First wait swap target buffer to be off screen with `IO_IOCTL_FBDEV_WAIT_OFFSCREEN`, and then make it front buffer with `IO_IOCTL_FBDEV_PAN_DISPLAY`. After 750 frames are swapped, we can get the FBDEV's refresh rate.
- Call `IO_IOCTL_FBDEV_WAIT_VSYNC` for 75 times to get the screen's VSYNC rate. It should be very close to the FBDEV's refresh rate.
- Disable `fbdev0` and `fbdev1` in sequence and close the device handles and free frame buffer memory.
- Terminate the application.