



Exercise 3 Multitasking

The goal of this exercise is to write a graphical program by use of multiple tasks.
For this, every group gets an Olimex STM32-LCD prototype board.

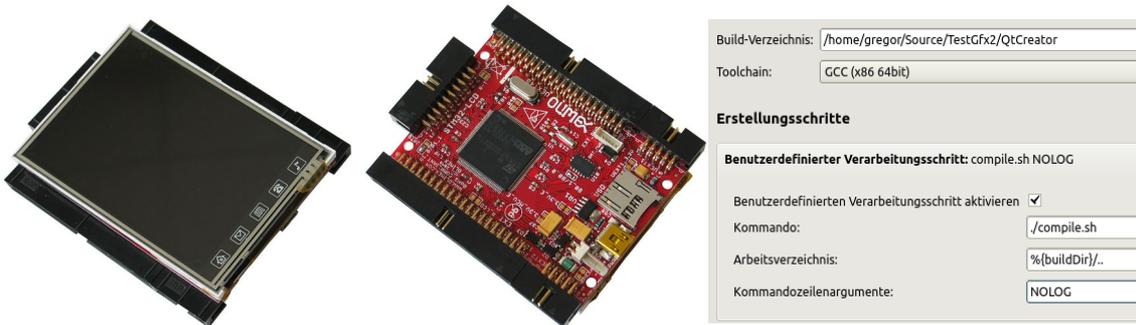


Abbildung 1: Project-Settings

You start by creating a new project with ToolChain v1.0.50 and enabling these lines in your activate_project.sh:

- activate.100_board_olimex_lcd.sh
- activate.600_example_gfx_printf.sh

Don't forget to deactivate other boards and examples. After a first compile run, you should see a counter on the screen.
You shall then modify this example and write your code right into example_gfx_printf.c/.h.

After loading the project into QtCreator, make sure that your project settings are correct for compilation:

The intended application shall start multiple tasks tBox() of the same function. Each task controls one Rectangle by use of these Variables:

X, Y the coordinate of the upper left corner of the rectangle

W, H width and height of the rectangle

C 24 bit color in which the rectangle is drawn (see ../common/ttc_gfx_types.h for predefined colors)

VX, VY velocity vector of the rectangle

The screen geometry is 240x320 pixels in size. Choose datatypes of your variables accordingly!

A graphics library is provided by ../common/ttc_gfx.c. It is activated automatically.

The most important function for this exercise is ttc_gfx_rect_fill() as described in ../common/ttc_gfx.h.

Important: As most graphics libraries, these functions can only be called from one task!

Therefore, the tBox() tasks have to send their draw-data into a queue. This queue is read by a drawing task tDraw().

tDraw() is the only task how is allowed to call ttc_gfx_XXX() functions.

If a box hits the wall, it shall be reflected. The current width and height of the screen can be found inside the global Variable DisplayConfig.

Tip: You can remove a box at its old position by drawing it in background color.

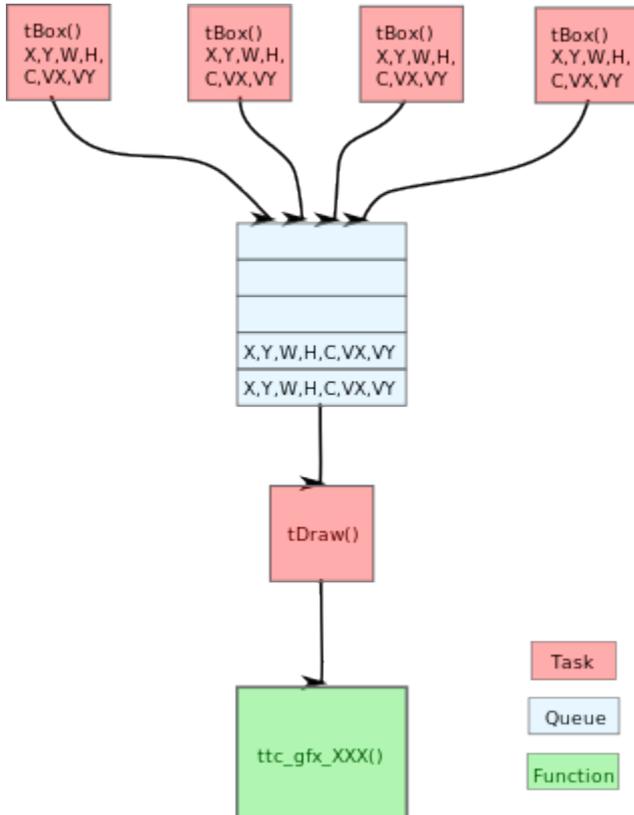


Abbildung 2: Dataflow between Tasks

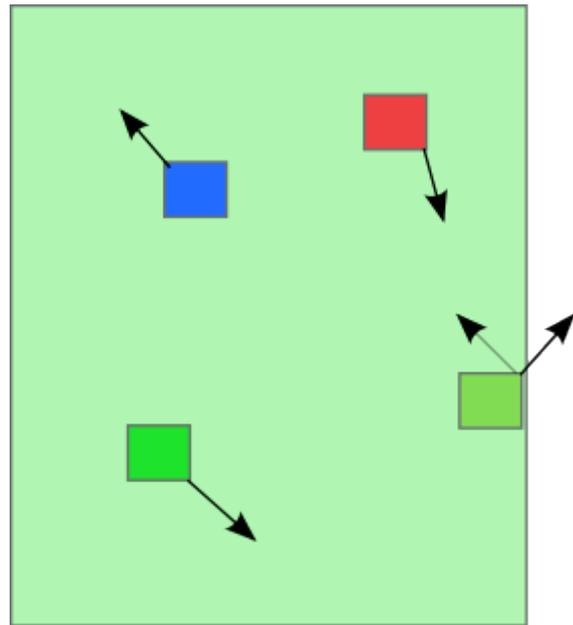


Abbildung 3: Boxes moving on the screen