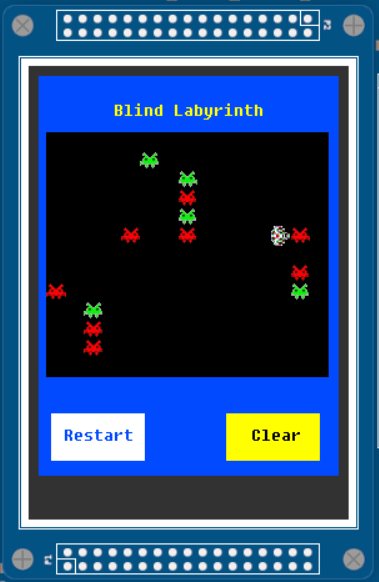
The application implements the game “Blind Labyrinth” and makes use of the LCD display on board the LandTiger. The project includes some pre-created libraries for managing the LCD and Touch Panel.

In particular, the LCD library is exposed by the GLCD.h header file, but I extended it by adding another source file (GLCD\_POWER.c) that adds some functions. In particular, we have:

* LCD\_DrawRectangle: this is capable of drawing rectangle given a starting point and a length and a width. In order to make the drawing faster (the use of the predefined LCD\_DrawLine function makes the drawing too slow), I had to expose from GLCD.c some functions (initially declared as static so no global available) like LCD\_SetPoint, LCD\_WriteIndex and LCD\_WriteData. This let me write directly inside the GRAM without using external functions. In particular, using LCD\_SetPoint is possible to set the starting “y” and “x” of a line of the rectangle to draw. With LCD\_WriteIndex I ask to the LCD that I want to send data to the GRAM (addess 0x0022) then with LCD\_WriteData I can send all the new pixels of my rectangle in a row.
* LCD\_DrawSpaceship: this is capable to draw the space ship from Space Invaders in the four directions (N, S, W, E). This represents the robot and is possible to draw it with two different background (to differentiate the EXPLORE and MOVE modes).
* LCD\_DrawSpacemonster: like the space ship, draws the monster from Space Invaders representing an obstacle of the labyrinth. The functions is capable of drawing both the red version and green version using a parameter.

The display shows the labyrinth map. Each cell has a resolution of 15x15 pixel. Some macros are available (inside the file lab\_map.h) useful to map a position of the labyrinth to the relative position on the LCD, and this is useful especially with functions like LCD\_DrawSpaceship.

Both the RIT and Timer0 are used. The RIT is used with a period of 20 ms and it pools the joystick and the Touch Screen. When a direction is selected in MOVE, it enables Timer0 that moves the robot along its direction with a period of 200 ms.

So, the RIT manages the Clear and Reset buttons on the display too. For the Clear operation, it loops through all the labyrinth and for each obstacle found (except for the one in front of the robot) it draws a void rectangle at its position on the LCD (even if there is no obstacle shown).