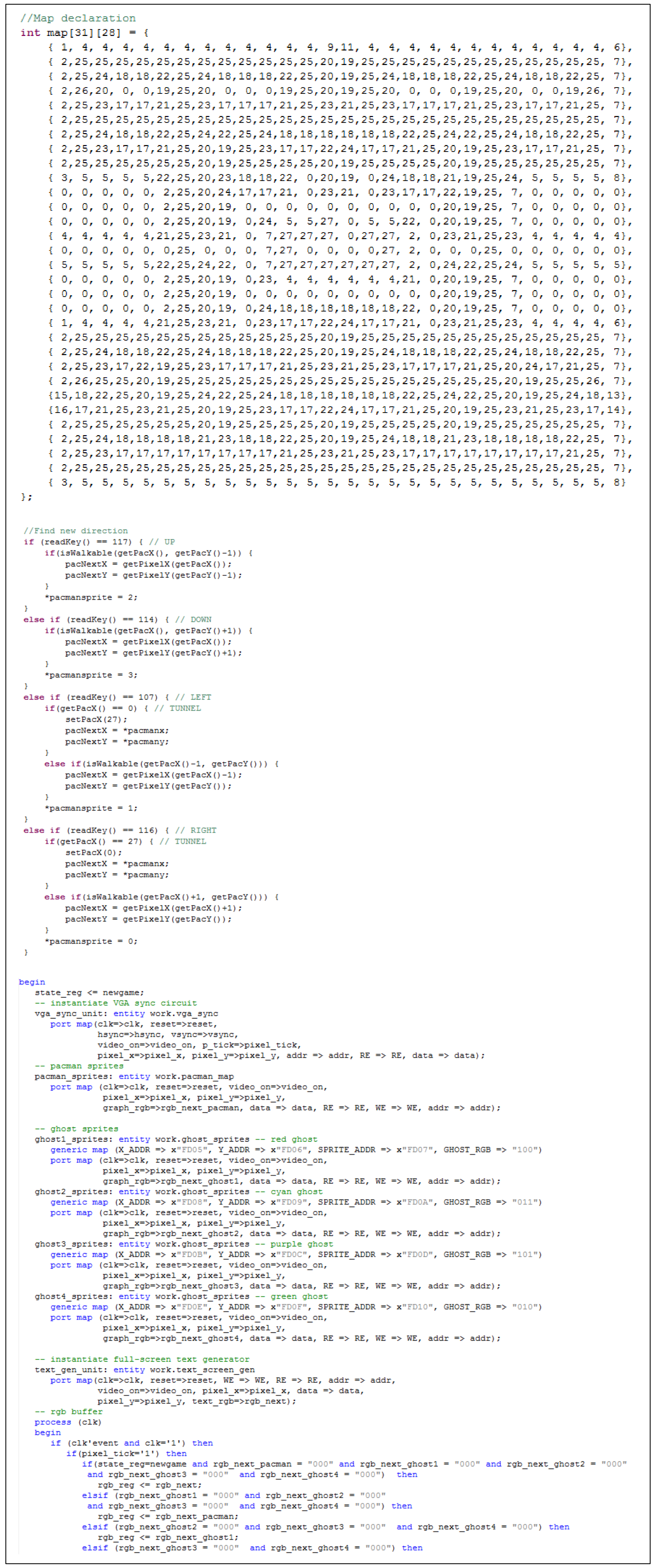
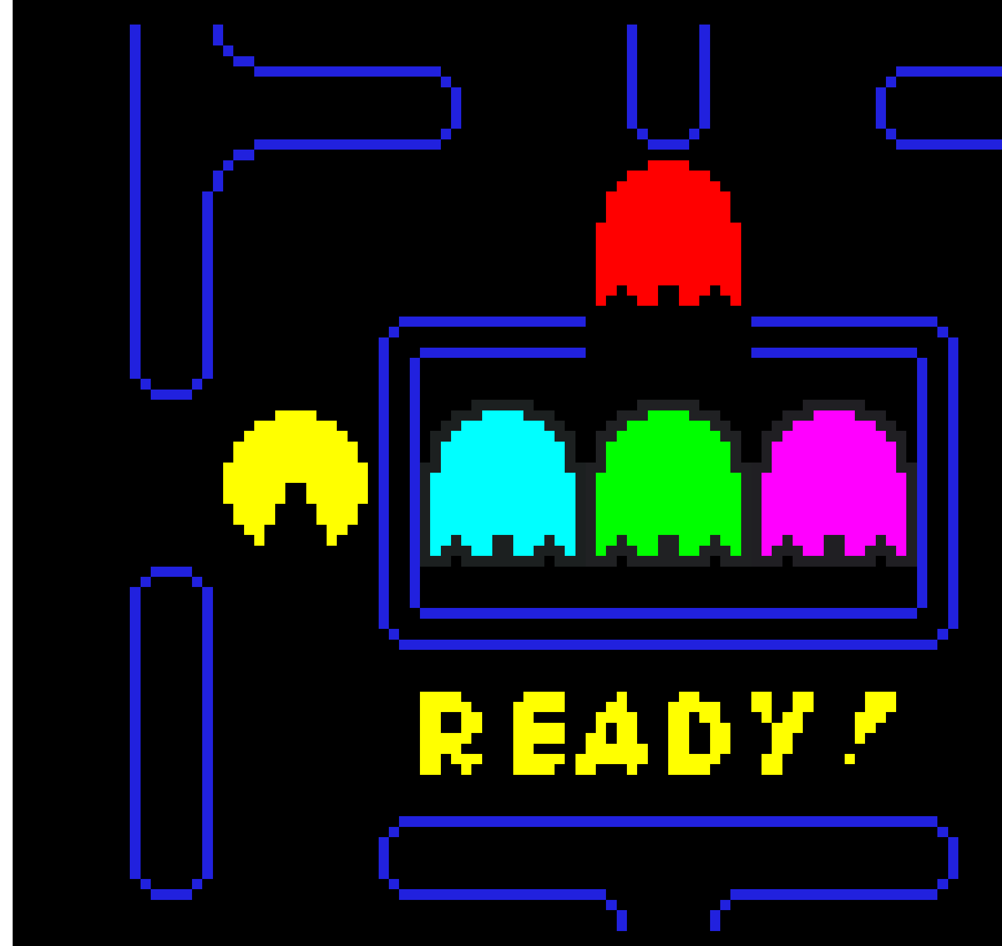
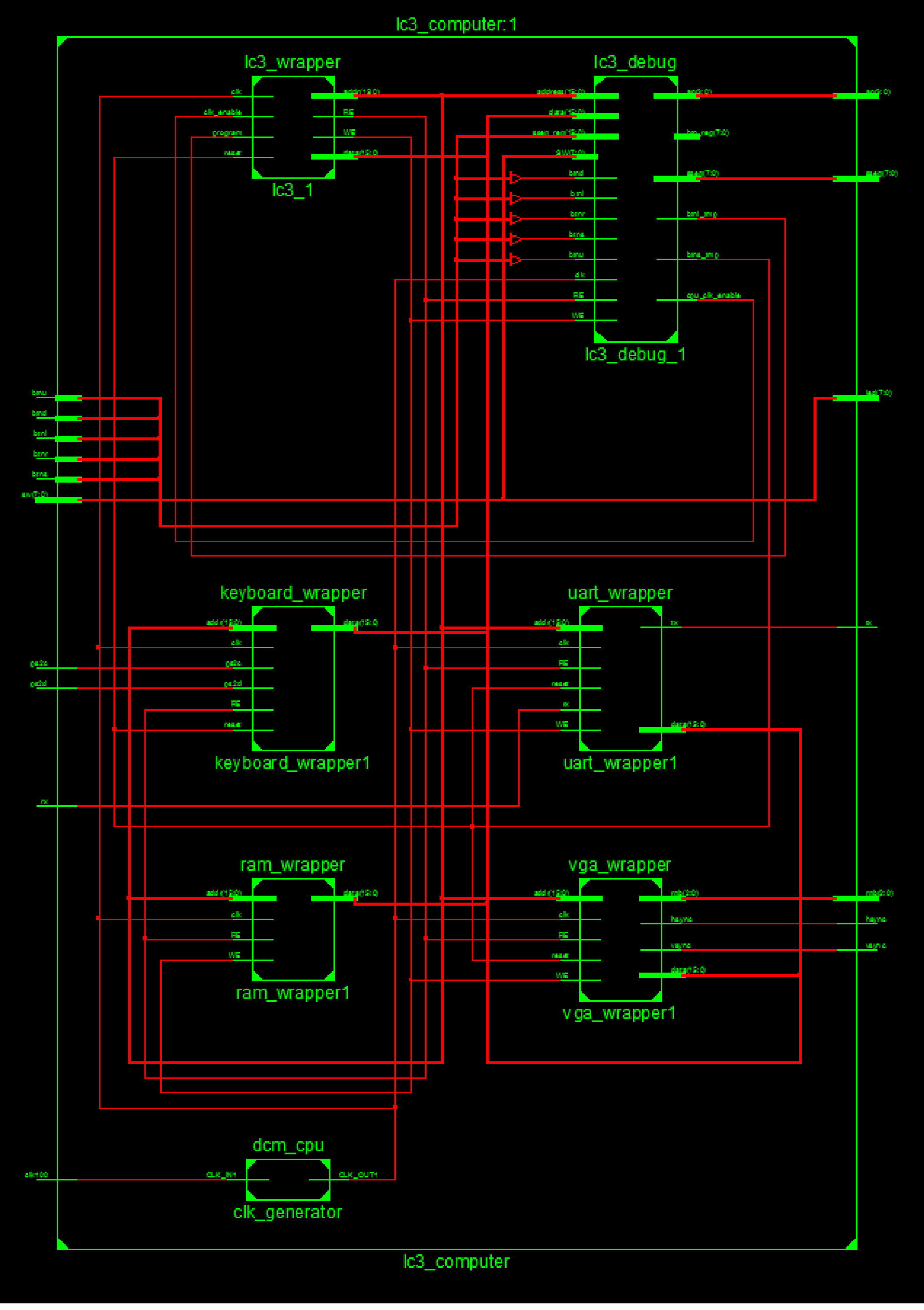
DTU Compute

Department of Applied Mathematics and Computer Science



```
//Map declaration
int map[31][28] = {
    { 1, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 9,11, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 6},
    { 2,25,25,25,25,25,25,25,25,25,25,25,25,20,19,25,25,25,25,25,25,25,25,25,25,25,25, 7},
    { 2,25,24,18,18,22,25,24,18,18,18,22,25,20,19,25,24,18,18,18,22,25,24,18,18,22,25, 7},
    { 2,26,20, 0, 0,19,25,20, 0, 0, 0,19,25,20,19,25,20, 0, 0, 0,19,25,20, 0, 0,19,26, 7},
    { 2,25,23,17,17,21,25,23,17,17,17,21,25,23,21,25,23,17,17,17,21,25,23,17,17,21,25, 7},
    { 2,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25, 7},
    { 2,25,24,18,18,22,25,24,22,25,24,18,18,18,18,18,18,22,25,24,22,25,24,18,18,22,25, 7},
    { 2,25,23,17,17,21,25,20,19,25,23,17,17,22,24,17,17,21,25,20,19,25,23,17,17,21,25, 7},
    { 2,25,25,25,25,25,25,20,19,25,25,25,25,20,19,25,25,25,25,20,19,25,25,25,25,25,25, 7},
    { 3, 5, 5, 5, 5,22,25,20,23,18,18,22, 0,20,19, 0,24,18,18,21,19,25,24, 5, 5, 5, 5, 8},
    { 0, 0, 0, 0, 0, 2,25,20,24,17,17,21, 0,23,21, 0,23,17,17,22,19,25, 7, 0, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 2,25,20,19, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,20,19,25, 7, 0, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 2,25,20,19, 0,24, 5, 5,27, 0, 5, 5,22, 0,20,19,25, 7, 0, 0, 0, 0, 0},
    { 4, 4, 4, 4, 4,21,25,23,21, 0, 7,27,27,27, 0,27,27, 2, 0,23,21,25,23, 4, 4, 4, 4, 4},
    { 0, 0, 0, 0, 0, 0,25, 0, 0, 0, 7,27, 0, 0, 0, 0,27, 2, 0, 0, 0,25, 0, 0, 0, 0, 0, 0},
    { 5, 5, 5, 5, 5,22,25,24,22, 0, 7,27,27,27,27,27,27, 2, 0,24,22,25,24, 5, 5, 5, 5, 5},
    { 0, 0, 0, 0, 0, 2,25,20,19, 0,23, 4, 4, 4, 4, 4, 4,21, 0,20,19,25, 7, 0, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 2,25,20,19, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,20,19,25, 7, 0, 0, 0, 0, 0},
    { 0, 0, 0, 0, 0, 2,25,20,19, 0,24,18,18,18,18,18,18,22, 0,20,19,25, 7, 0, 0, 0, 0, 0},
    { 1, 4, 4, 4, 4,21,25,23,21, 0,23,17,17,22,24,17,17,21, 0,23,21,25,23, 4, 4, 4, 4, 6},
    { 2,25,25,25,25,25,25,25,25,25,25,25,25,20,19,25,25,25,25,25,25,25,25,25,25,25,25, 7},
    { 2,25,24,18,18,22,25,24,18,18,18,22,25,20,19,25,24,18,18,18,22,25,24,18,18,22,25, 7},
    { 2,25,23,17,22,19,25,23,17,17,17,21,25,23,21,25,23,17,17,17,21,25,20,24,17,21,25, 7},
    { 2,26,25,25,20,19,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,20,19,25,25,26, 7},
    {15,18,22,25,20,19,25,24,22,25,24,18,18,18,18,18,18,22,25,24,22,25,20,19,25,24,18,13},
    {16,17,21,25,23,21,25,20,19,25,23,17,17,22,24,17,17,21,25,20,19,25,23,21,25,23,17,14},
    { 2,25,25,25,25,25,25,20,19,25,25,25,25,20,19,25,25,25,25,20,19,25,25,25,25,25,25, 7},
    { 2,25,24,18,18,18,18,21,23,18,18,22,25,20,19,25,24,18,18,21,23,18,18,18,18,22,25, 7},
    { 2,25,23,17,17,17,17,17,17,17,17,21,25,23,21,25,23,17,17,17,17,17,17,17,17,21,25, 7},
    { 2,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25,25, 7},
    { 3, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 8}
};


 //Find new direction
 if (readKey() == 117) { // UP
     if(isWalkable(getPacX(), getPacY()-1)) {
         pacNextX = getPixelX(getPacX());
         pacNextY = getPixelY(getPacY()-1);
     }
     *pacmansprite = 2;
 }
 else if (readKey() == 114) { // DOWN
     if(isWalkable(getPacX(), getPacY()+1)) {
         pacNextX = getPixelX(getPacX());
         pacNextY = getPixelY(getPacY()+1);
     }
     *pacmansprite = 3;
 }
 else if (readKey() == 107) { // LEFT
     if(getPacX() == 0) { // TUNNEL
         setPacX(27);
         pacNextX = *pacmanx;
         pacNextY = *pacmany;
     }
     else if(isWalkable(getPacX()-1, getPacY())) {
         pacNextX = getPixelX(getPacX()-1);
         pacNextY = getPixelY(getPacY());
     }
     *pacmansprite = 1;
 }
 else if (readKey() == 116) { // RIGHT
     if(getPacX() == 27) { // TUNNEL
         setPacX(0);
         pacNextX = *pacmanx;
         pacNextY = *pacmany;
     }
     else if(isWalkable(getPacX()+1, getPacY())) {
         pacNextX = getPixelX(getPacX()+1);
         pacNextY = getPixelY(getPacY());
     }
     *pacmansprite = 0;
 }


begin
   state_reg <= newgame;
   -- instantiate VGA sync circuit
   vga_sync_unit: entity work.vga_sync
      port map(clk=>clk, reset=>reset,
               hsync=>hsync, vsync=>vsync,
               video_on=>video_on, p_tick=>pixel_tick,
               pixel_x=>pixel_x, pixel_y=>pixel_y, addr => addr, RE => RE, data => data);
   -- pacman sprites
   pacman_sprites: entity work.pacman_map
      port map (clk=>clk, reset=>reset, video_on=>video_on,
                pixel_x=>pixel_x, pixel_y=>pixel_y,
                graph_rgb=>rgb_next_pacman, data => data, RE => RE, WE => WE, addr => addr);

   -- ghost sprites
   ghost1_sprites: entity work.ghost_sprites -- red ghost
      generic map (X_ADDR => x"FD05", Y_ADDR => x"FD06", SPRITE_ADDR => x"FD07", GHOST_RGB => "100")
      port map (clk=>clk, reset=>reset, video_on=>video_on,
                pixel_x=>pixel_x, pixel_y=>pixel_y,
                graph_rgb=>rgb_next_ghost1, data => data, RE => RE, WE => WE, addr => addr);
   ghost2_sprites: entity work.ghost_sprites -- cyan ghost
      generic map (X_ADDR => x"FD08", Y_ADDR => x"FD09", SPRITE_ADDR => x"FD0A", GHOST_RGB => "011")
      port map (clk=>clk, reset=>reset, video_on=>video_on,
                pixel_x=>pixel_x, pixel_y=>pixel_y,
                graph_rgb=>rgb_next_ghost2, data => data, RE => RE, WE => WE, addr => addr);
   ghost3_sprites: entity work.ghost_sprites -- purple ghost
      generic map (X_ADDR => x"FD0B", Y_ADDR => x"FD0C", SPRITE_ADDR => x"FD0D", GHOST_RGB => "101")
      port map (clk=>clk, reset=>reset, video_on=>video_on,
                pixel_x=>pixel_x, pixel_y=>pixel_y,
                graph_rgb=>rgb_next_ghost3, data => data, RE => RE, WE => WE, addr => addr);
   ghost4_sprites: entity work.ghost_sprites -- green ghost
      generic map (X_ADDR => x"FD0E", Y_ADDR => x"FD0F", SPRITE_ADDR => x"FD10", GHOST_RGB => "010")
      port map (clk=>clk, reset=>reset, video_on=>video_on,
                pixel_x=>pixel_x, pixel_y=>pixel_y,
                graph_rgb=>rgb_next_ghost4, data => data, RE => RE, WE => WE, addr => addr);

   -- instantiate full-screen text generator
   text_gen_unit: entity work.text_screen_gen
      port map(clk=>clk, reset=>reset, WE => WE, RE => RE, addr => addr,
               video_on=>video_on, pixel_x=>pixel_x, data => data,
               pixel_y=>pixel_y, text_rgb=>rgb_next);
   -- rgb buffer
   process (clk)
   begin
      if (clk'event and clk='1') then
         if(pixel_tick='1') then
            if(state_reg=newgame and rgb_next_pacman = "000" and rgb_next_ghost1 = "000" and rgb_next_ghost2 = "000"
             and rgb_next_ghost3 = "000"  and rgb_next_ghost4 = "000")  then
              rgb_reg <= rgb_next;
            elsif (rgb_next_ghost1 = "000" and rgb_next_ghost2 = "000"
             and rgb_next_ghost3 = "000"  and rgb_next_ghost4 = "000") then
              rgb_reg <= rgb_next_pacman;
            elsif (rgb_next_ghost2 = "000" and rgb_next_ghost3 = "000"  and rgb_next_ghost4 = "000") then
              rgb_reg <= rgb_next_ghost1;
            elsif (rgb_next_ghost3 = "000"  and rgb_next_ghost4 = "000") then
```





Introduction

- Create our own LC3 computer
- Hardware formed in VHDL
- Gameplay created in C

Gameplay

- Eat the dots to complete a level
- Avoid the ghosts chasing you
- Eat power pills to eat the ghosts
- Beat the highscore
- Cheat the ghosts by going through the tunnel

Control Pac-Man using the arrow keys

Results

- We managed to build the wanted functionallity in our LC3 computer.
- We got a fantastic gameplay.
- Working ghost AI for chasing Pac-Man
- The graphics came out good with room for future improvements.

02321 HW/SW programming - Diplom-IT - January 2014 – 3 weeks project – Jacob Honoré (s123806), Thomas Mortensen (s110795), Chris Johansen (s113577)