Middle East Technical University **Ceng336**
Computer Engineering Department **Spring 2017-2018**

(Individual) Take Home Exam-1

Revision 1.0

Due: Mar 13, 2018, 23:55:50hrs

Submission: ***via ODTÜCLASS***

The purpose of this assignment is to familiarize you with **basic input/output operations on MPLAB X IDE** simulation environment.

Any clarifications and revisions for the assignment will be posted to the discussion group on ODTÜCLASS.

This is not a group exam, you must solve it **individually.**

Your mission is to implement the below scenario using the MPLAB X IDE simulation environment. You can also program the MCDEV boards to observe actual behavior.

Scenario

The snake *Loki* loves to travel inside an embedded board. It has two patterns for travelling; *State1* and *State2*. In *State1*, the snake will take the clockwise path, whereas in *State2* counter-clockwise path will be taken by the snake. The way that *Loki* moves in its path will be changed by button actions.

Specifications

Four ports will be used in this assignment: **PORTA**, **PORTB**, **PORTC,** and **PORTD**. PORTC and PORTD will only be used as output; those LED pins are **RC [0-3]** and **RD [0-3].** To implement state changer button action, **RA4 pin** must be configured as a **digital input,** please look at the relevant section in datasheet. Other pins of PORTA, namely **RA [0-3]** must be used as output to show LED lights. Similarly, PORTB will be both used as the LED outputs (**RB [0-3]**) and as an input button that turns direction of the snake (**RB5)**.

The default state of the program is State1. At the very beginning of the program, you must turn on **first four LEDs of all the ports** (16 pins in total) for 2 seconds. Afterwards, place the snake at **RA0** (tail) and **RB0** (head) position by turning on those two LEDs, and start move along the path. The path is composed of pins **RA0, RB0, RC0, RD0, RD1, RD2, RD3, RC3, RB3, RA3, RA2, RA1,** in order. In each step, the snake advances its head and tail to the next LED positions and turns off old tail LED. If the head ever reaches to corners (**RD0, RD3, RA3, RA0**), the snake stops and waits for an **RB5** button action. RB5 push and release action allows the snake to continue moving on next side of the square. All RB5 button actions will be ignored if the snake's head is not in a corner LED position.

Whenever **RA4** is pushed and released, the state of the game changes to State2. This time the snake will follow the path in reverse order. At the state changes, the snake stops and makes its head as new tail and vice versa. Then it continues from where it was left of before the state change. Note that RA4 button action is different than RB5 button action in a way that it can be triggered regardless of the position. In other words, even if the snake is not on corners, RA4 button action may occur asynchronously and change the state of the game. Notice that RA4 button action also makes the snake move (in reverse direction) while it waits for RB5 button action on one of the corners.

LED switching during movement of the snake must be done in **750 ms** (+/- 50 ms) in the *State1*, whereas **400 ms** (+/- 50 ms) in the *State2*.

The naming of pins is <R>+"PORT NAME"+"BIT ORDER". R represents PORT. Here, ports names are A, B, C, and D. And the range of the BIT ORDER is between 0 and 7.

Implementation

You are expected to simulate the above scenario in the **MPLAB X IDE simulator**. You can track the status of the LEDs and buttons, by using **RA0, RA1, RA2, RA3, RB0, RB3, RC0, RC3, RD0, RD1, RD2, RD3, RA4,** and **RB5** pins of the simulator. The output of the pins can be seen by using the **I/O pins menu** which is available under the **Window --> Simulator** menu. A screen shot from the I/O pins menu of the simulator is given in *Figure 1*. There you can find states of some pins that are representing the LEDs and buttons. You can also simulate push buttons by using the **Stimulus** menu. This feature is available under the **Window --> Simulator** menu. You can control voltage levels of the buttons as High or Low by clicking the **RB5** and **RA4** pin button which is can be seen in *Figure 2*. You can add the pin corresponded with push button by using **the I/O pins menu** to follow the changes on this pin. **Note that** in *Figure 1* and *Figure 2*, some pins are not relevant to this assignment and only given as an example.

Coding Rules:

- You will code your program using PIC assembly language.
- Your program should be written for **PIC18F8722** working at **40 MHz.**
- When the push button is pressed, then RB5/RA4 pin goes high, and when we release the push button, the **RB5/RA4** pin goes low.
- When you are writing your codes, please look at the relevant lecture notes and recitation documents. In particular, you are expected to apply the **round robin approach** to your code, so your program will execute in an infinite loop, and there will be **your own tasks for your states** in this loop at the main scope.
- Be precise for constructing time delays (intervals). You may need to count the number of instructions for those sections.

Hand in Instructions

- You should submit your code as a single file named as `the_1_##.asm` through ODTÜCLASS where ## represents your seven-digit student number.

Hints

- You can use “Stopwatch” tool of the simulator in MPLAB X IDE to measure the time spent by a code segment. You can reach this tool from `"Windows -> Debugging -> Stopwatch"` menu. But, before starting simulator you must configure your clock speed to 10 MHz from `"Project properties -> Simulator -> Instruction Frequency"` (Since 40 MHz main clock frequency is divided by 4 inside microcontroller and one instruction cycle is equal to 4 cycle of 40 MHz clock signal, you are setting “Instruction Frequency” to 10 MHz). You can reach the project properties by right clicking to the project name in the left side “Projects” panel and selecting properties. By **putting breakpoints** on the lines between which you want to measure the amount of time spent by that code segment and running the code between these two breakpoints, you can see the time spent in stopwatch window.

- You will use **RA4 pin** as a **digital input** by configuring a **special register**, please check the data sheet for this configuration.
- You can also see the states of your ports, variables and pins by using the logical analyzer, SFR and variables menu under the window menu.

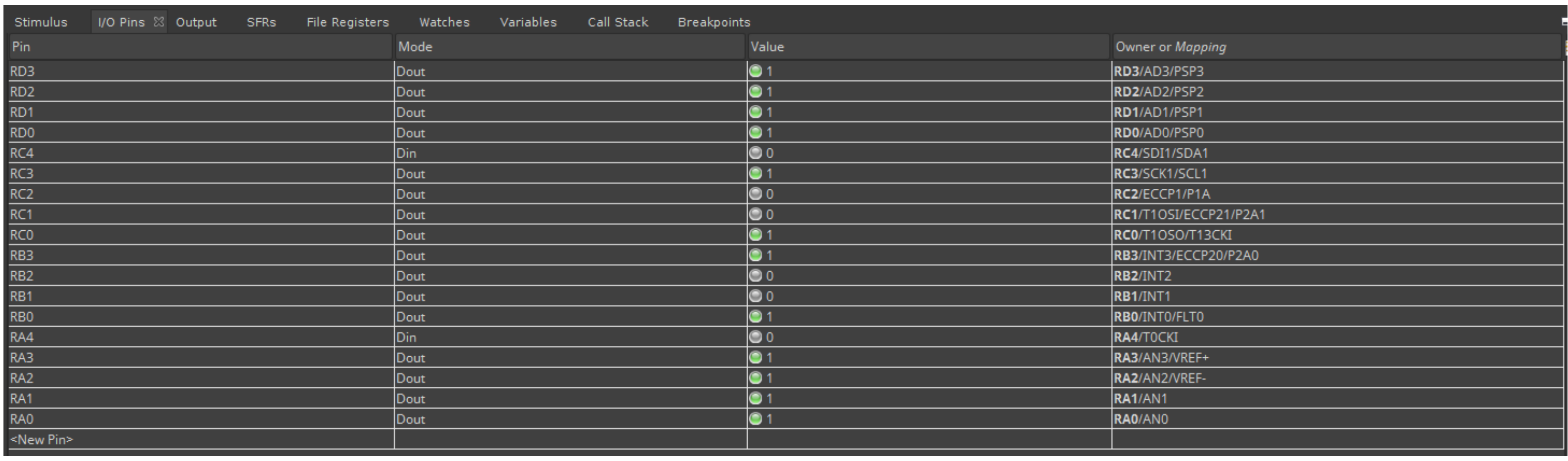
Grading

Your codes will be evaluated on the MPLAB X simulation environment. However, if you obey the specified rules, the program can directly be executed on the boards without any problem.

Resources

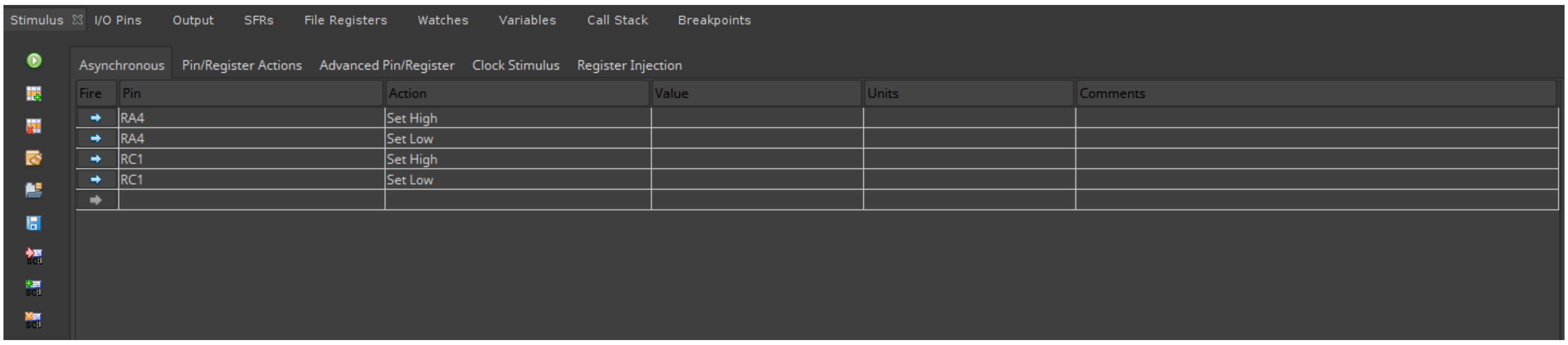
- PIC 18F8722 Datasheet
- ODTÜCLASS course web page
- The **MPLAB X IDE** will be used. You can download by using the following link:

http://www.microchip.com/pagehandler/en-us/family/mplabx/home.html



Pin	Mode	Value	Owner or *Mapping*
RD3	Dout	🟢 1	**RD3**/AD3/PSP3
RD2	Dout	🟢 1	**RD2**/AD2/PSP2
RD1	Dout	🟢 1	**RD1**/AD1/PSP1
RD0	Dout	🟢 1	**RD0**/AD0/PSP0
RC4	Din	⚪ 0	**RC4**/SDI1/SDA1
RC3	Dout	🟢 1	**RC3**/SCK1/SCL1
RC2	Dout	⚪ 0	**RC2**/ECCP1/P1A
RC1	Dout	⚪ 0	**RC1**/T1OSI/ECCP21/P2A1
RC0	Dout	🟢 1	**RC0**/T1OSO/T13CKI
RB3	Dout	🟢 1	**RB3**/INT3/ECCP20/P2A0
RB2	Dout	⚪ 0	**RB2**/INT2
RB1	Dout	⚪ 0	**RB1**/INT1
RB0	Dout	🟢 1	**RB0**/INT0/FLT0
RA4	Din	⚪ 0	**RA4**/T0CKI
RA3	Dout	🟢 1	**RA3**/AN3/VREF+
RA2	Dout	🟢 1	**RA2**/AN2/VREF-
RA1	Dout	🟢 1	**RA1**/AN1
RA0	Dout	🟢 1	**RA0**/AN0
<New Pin>			

Figure 1: **S**tate of various LED and button pins shown in I/O pins window.

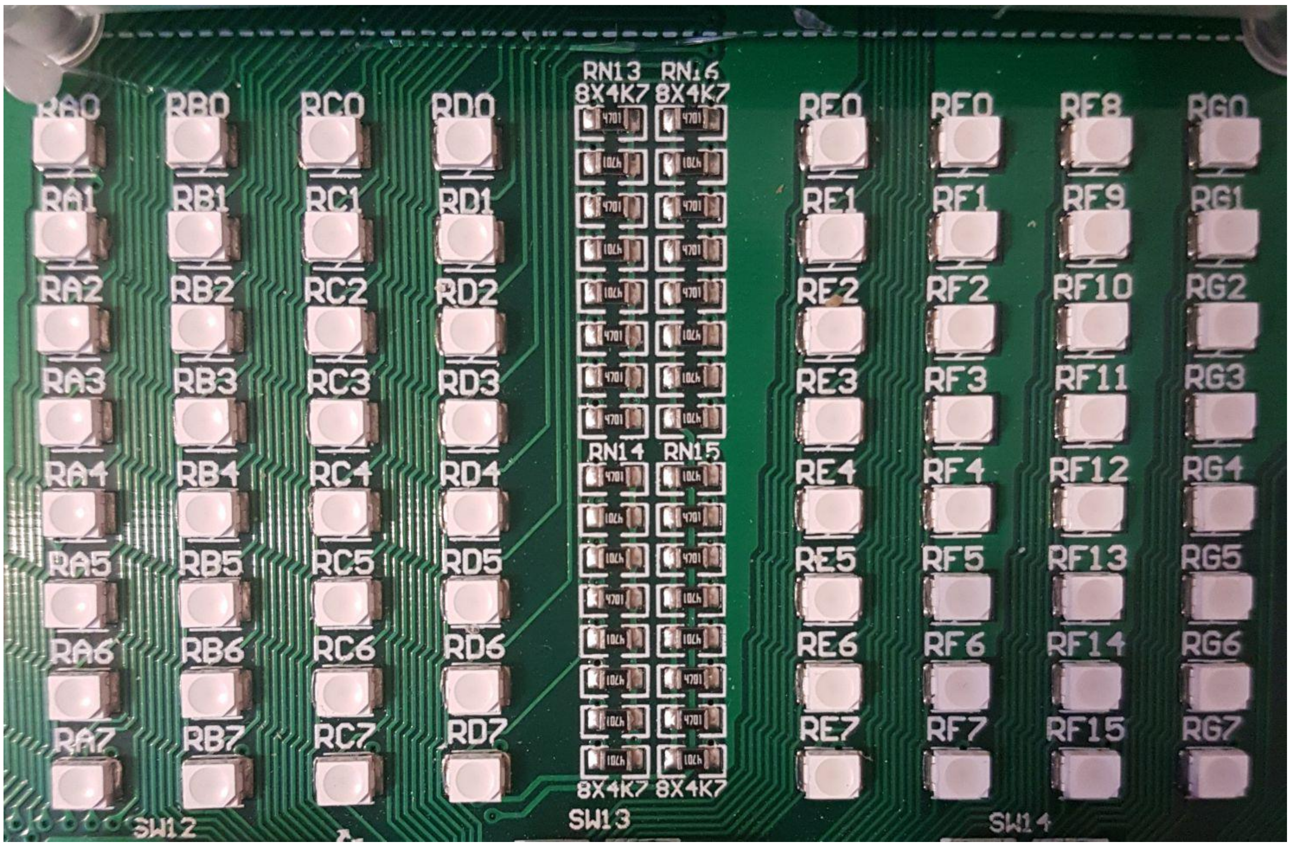

Stimulus | I/O Pins | Output | SFRs | File Registers | Watches | Variables | Call Stack | Breakpoints

Asynchronous | Pin/Register Actions | Advanced Pin/Register | Clock Stimulus | Register Injection

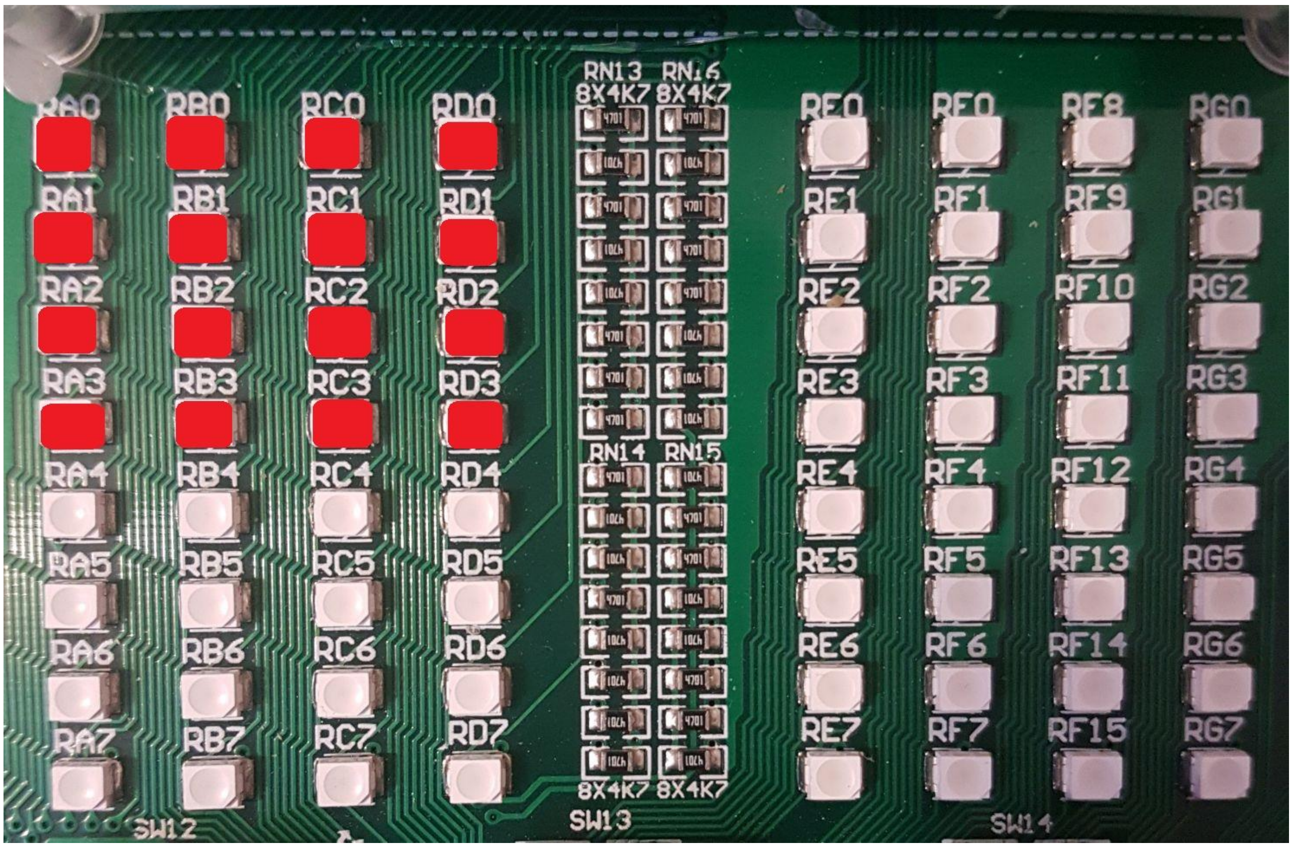
Fire	Pin	Action	Value	Units	Comments
➡	RA4	Set High			
➡	RA4	Set Low			
➡	RC1	Set High			
➡	RC1	Set Low			
➡					

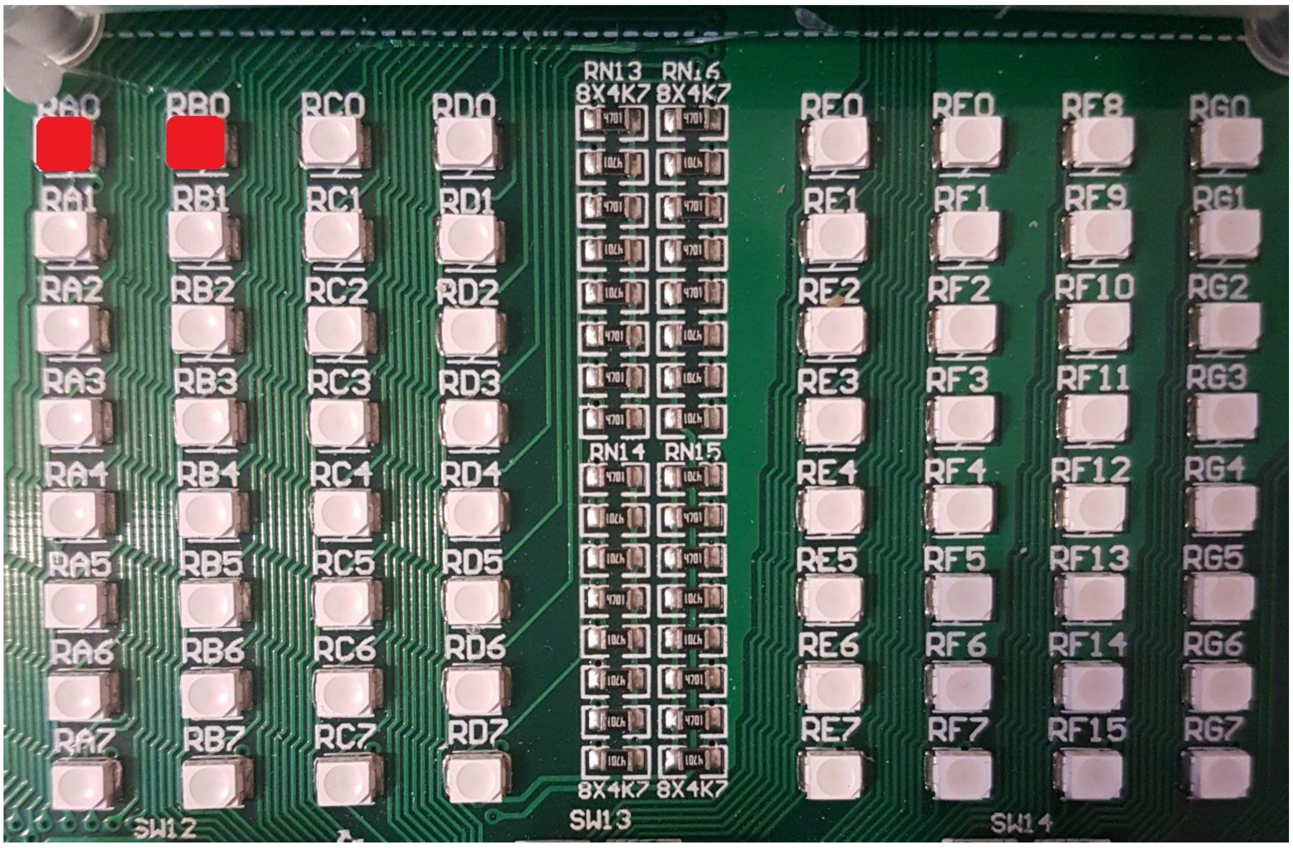
Figure 2: Example button (RA4 & RC1) control using the stimulus window.

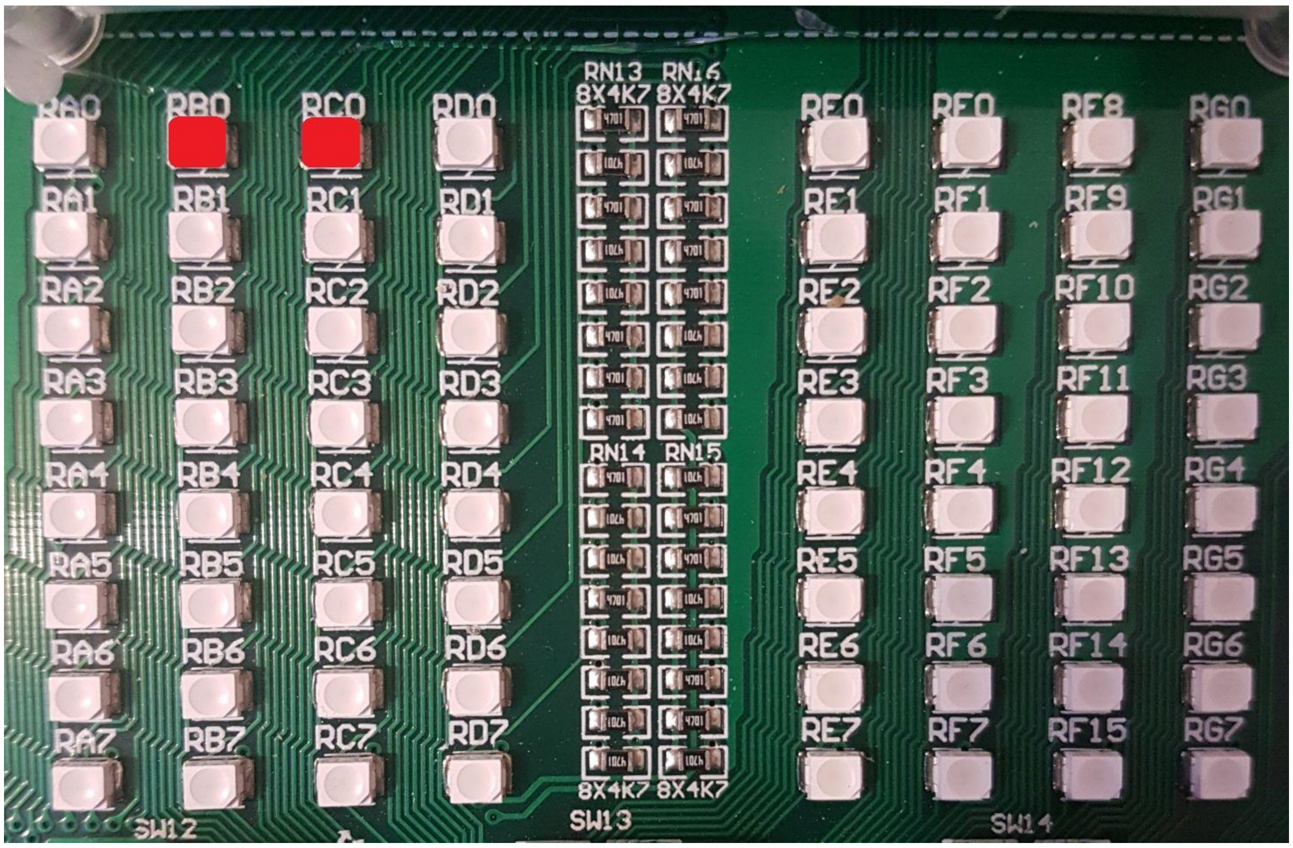
Game Snapshots

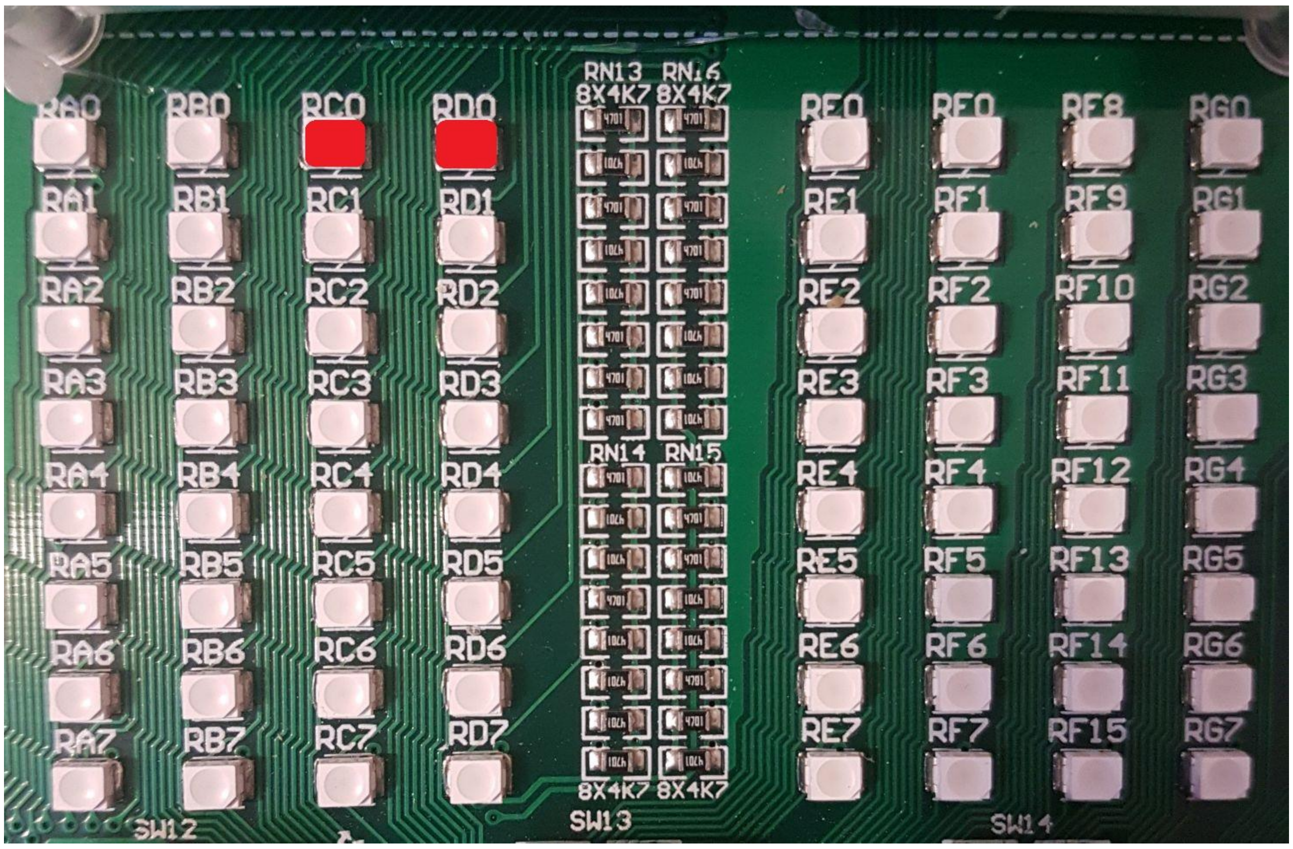


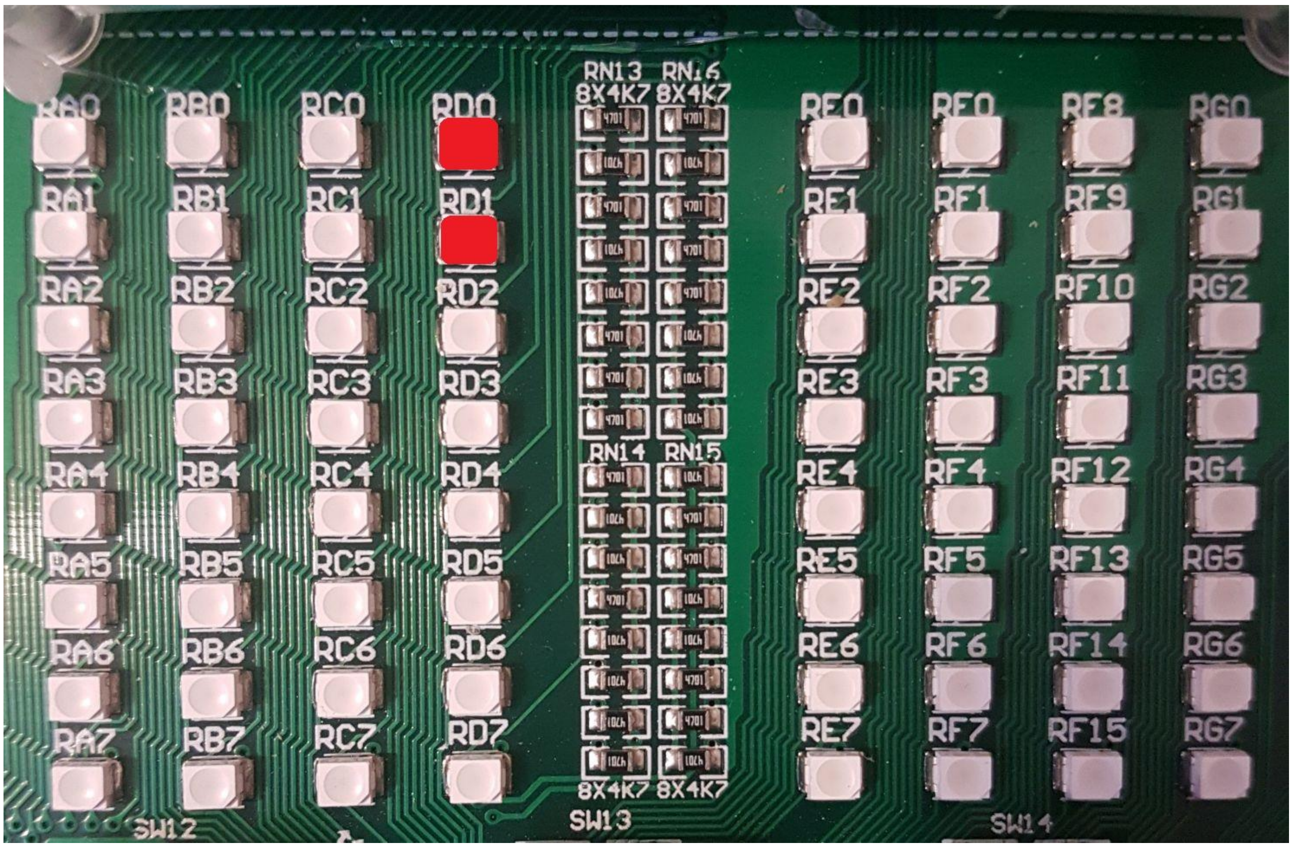
Board is off


Initial Configuration for 2 seconds

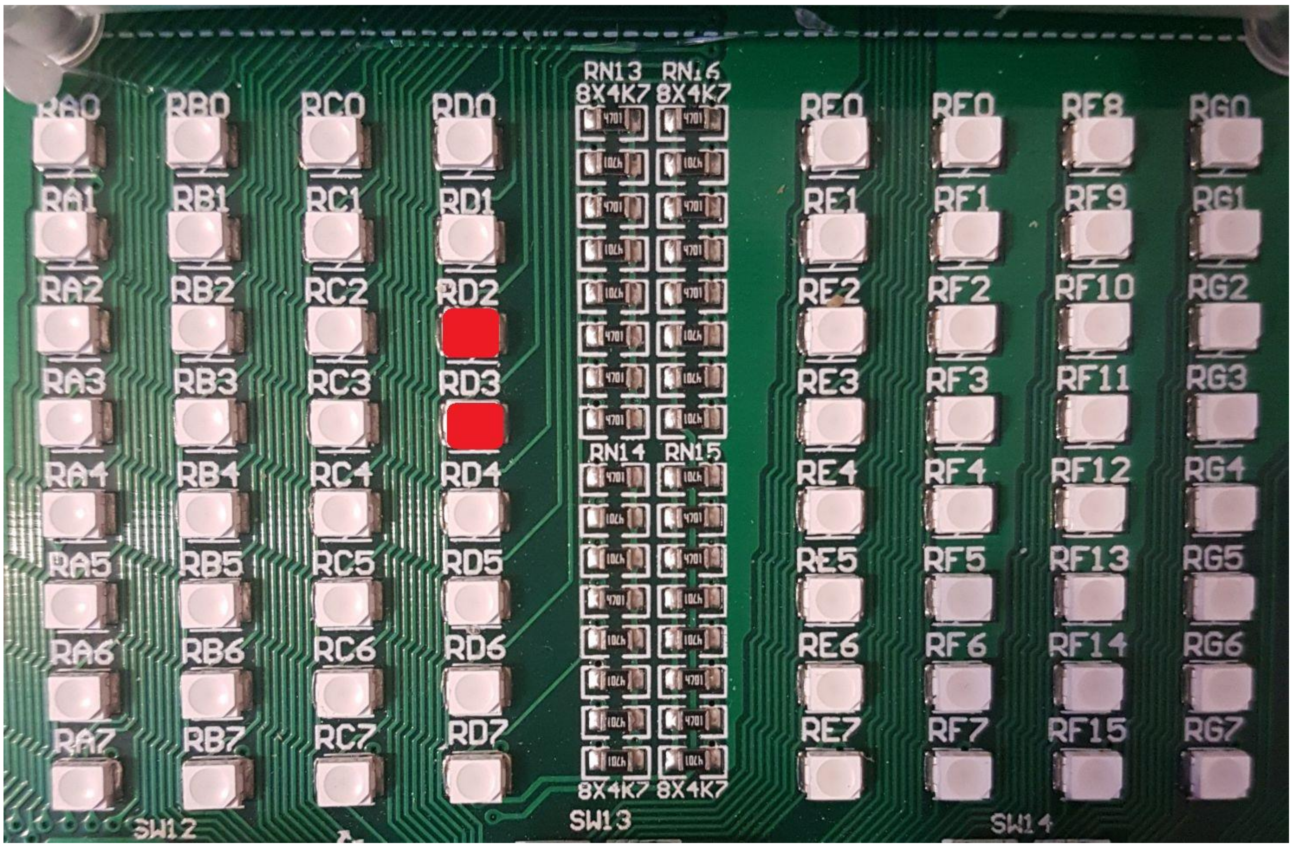

The snake is at the first position (RB0 is the head)


RA0
RB0
RC0
RD0
RN13 8X4K7
RN16 8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RA4
RB4
RC4
RD4
RN14
RN15
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
8X4K7
8X4K7
SW12
SW13
SW14

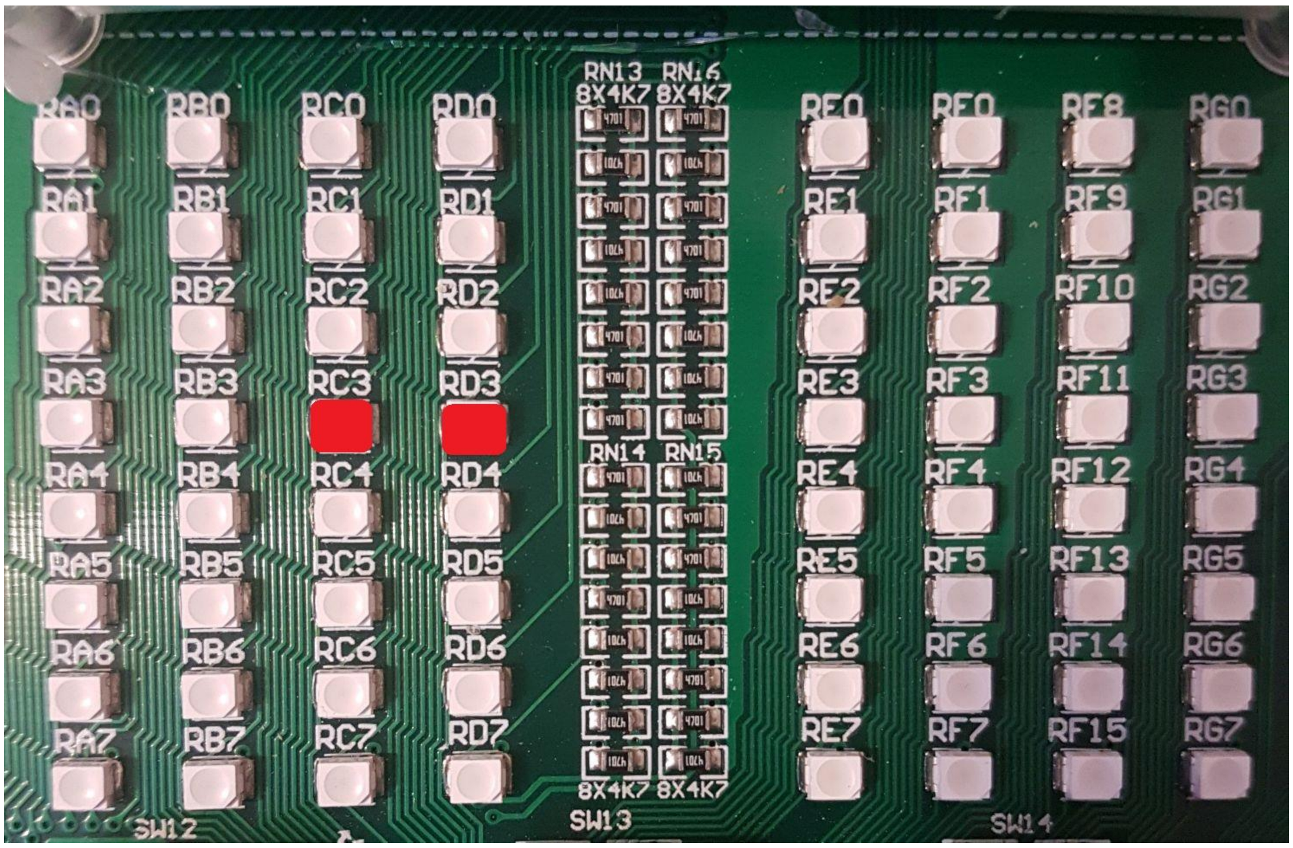

RA0
RB0
RC0
RD0
RN13 8X4K7
RN16 8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RA4
RB4
RC4
RD4
RN14
RN15
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
8X4K7
8X4K7
SW12
SW13
SW14

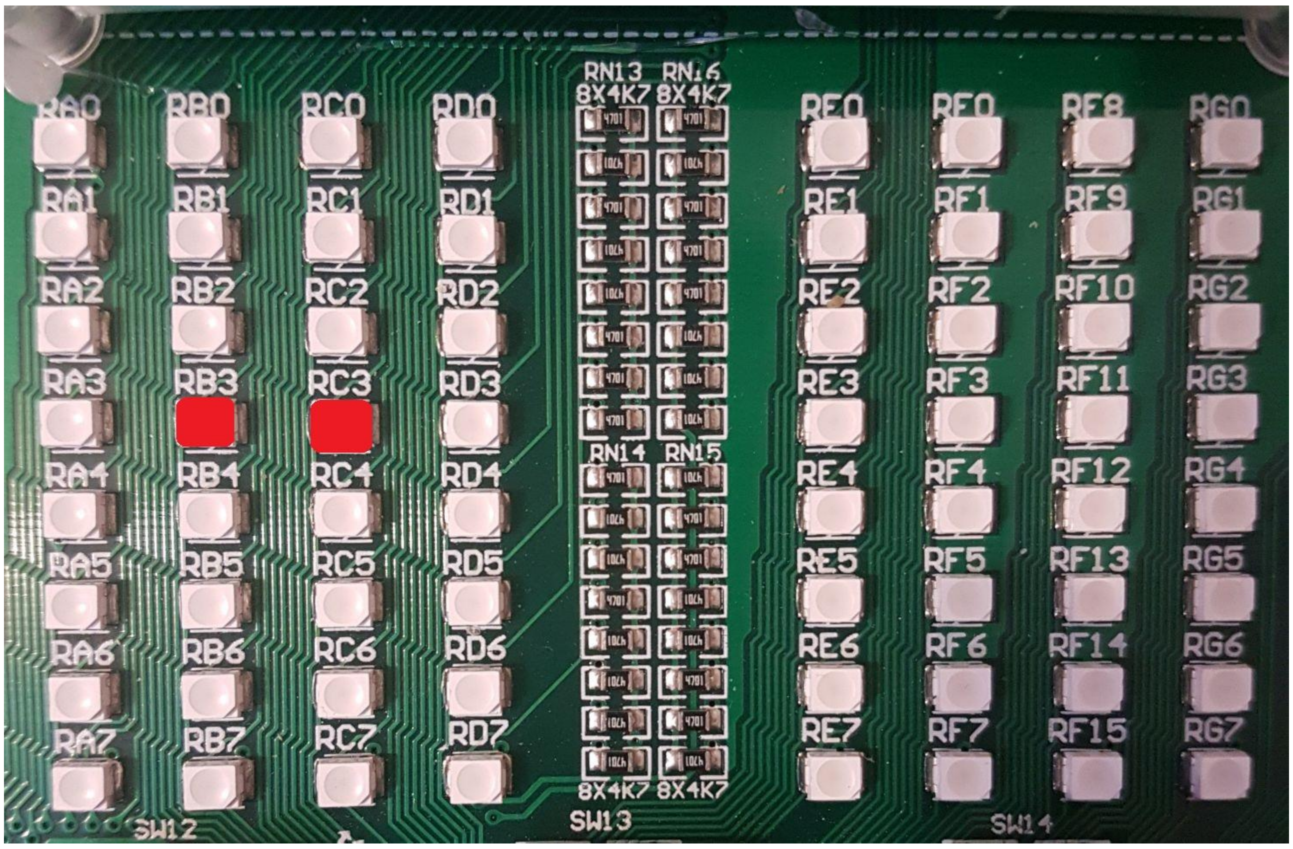


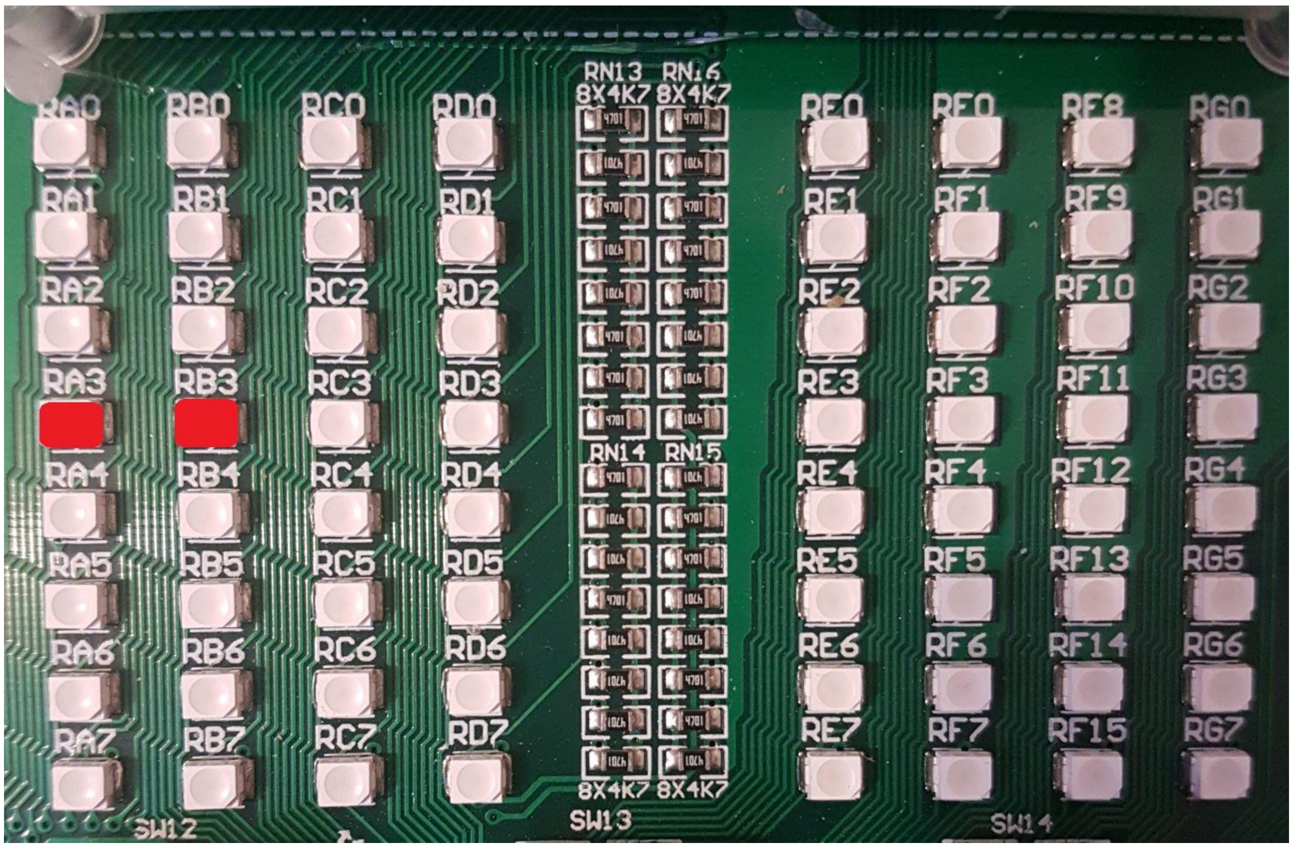
RA0 RB0 RC0 RD0 RE0 RF0 RF8 RG0
RA1 RB1 RC1 RD1 RE1 RF1 RF9 RG1
RA2 RB2 RC2 RD2 RE2 RF2 RF10 RG2
RA3 RB3 RC3 RD3 RE3 RF3 RF11 RG3
RA4 RB4 RC4 RD4 RE4 RF4 RF12 RG4
RA5 RB5 RC5 RD5 RE5 RF5 RF13 RG5
RA6 RB6 RC6 RD6 RE6 RF6 RF14 RG6
RA7 RB7 RC7 RD7 RE7 RF7 RF15 RG7
RN13 8X4K7 RN16 8X4K7
RN14 RN15
8X4K7 8X4K7
SW12 SW13 SW14

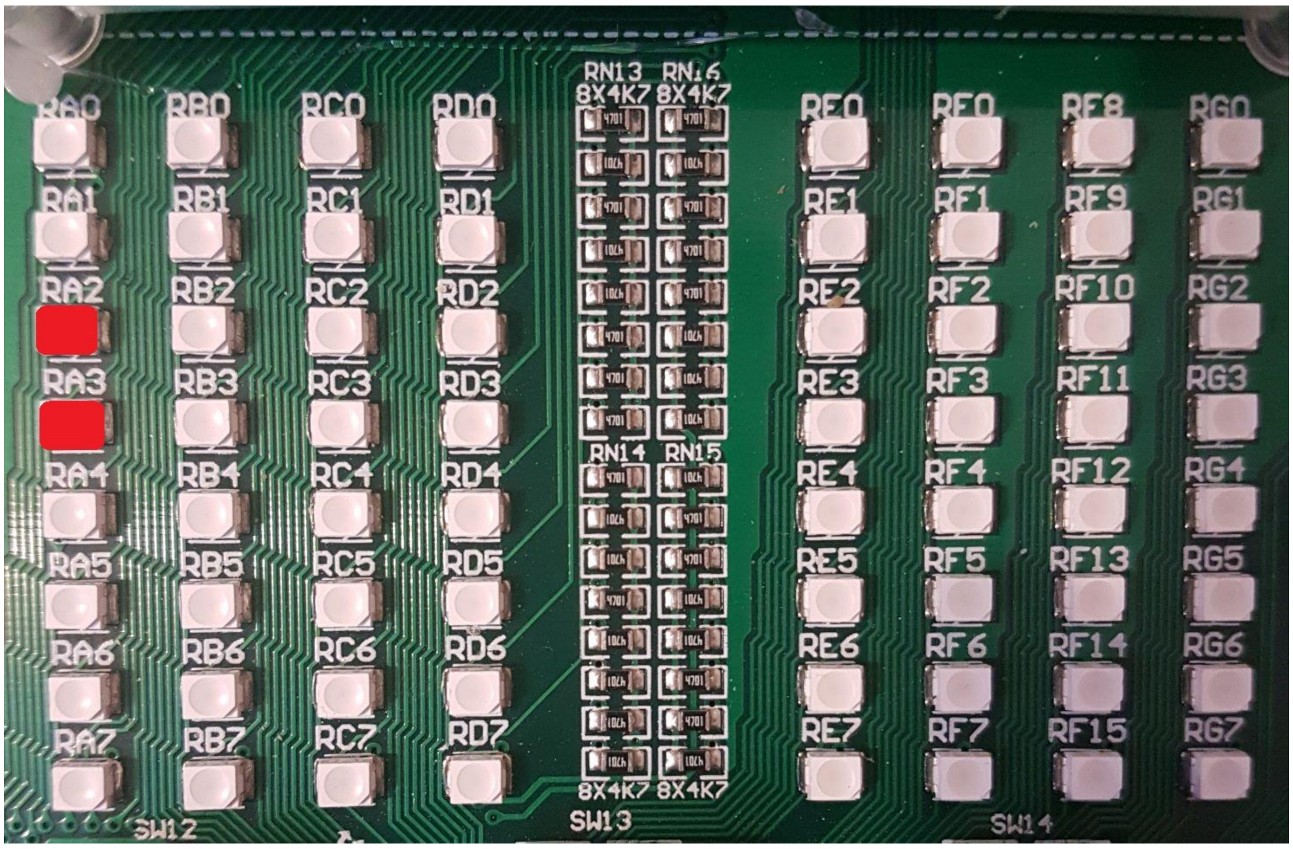


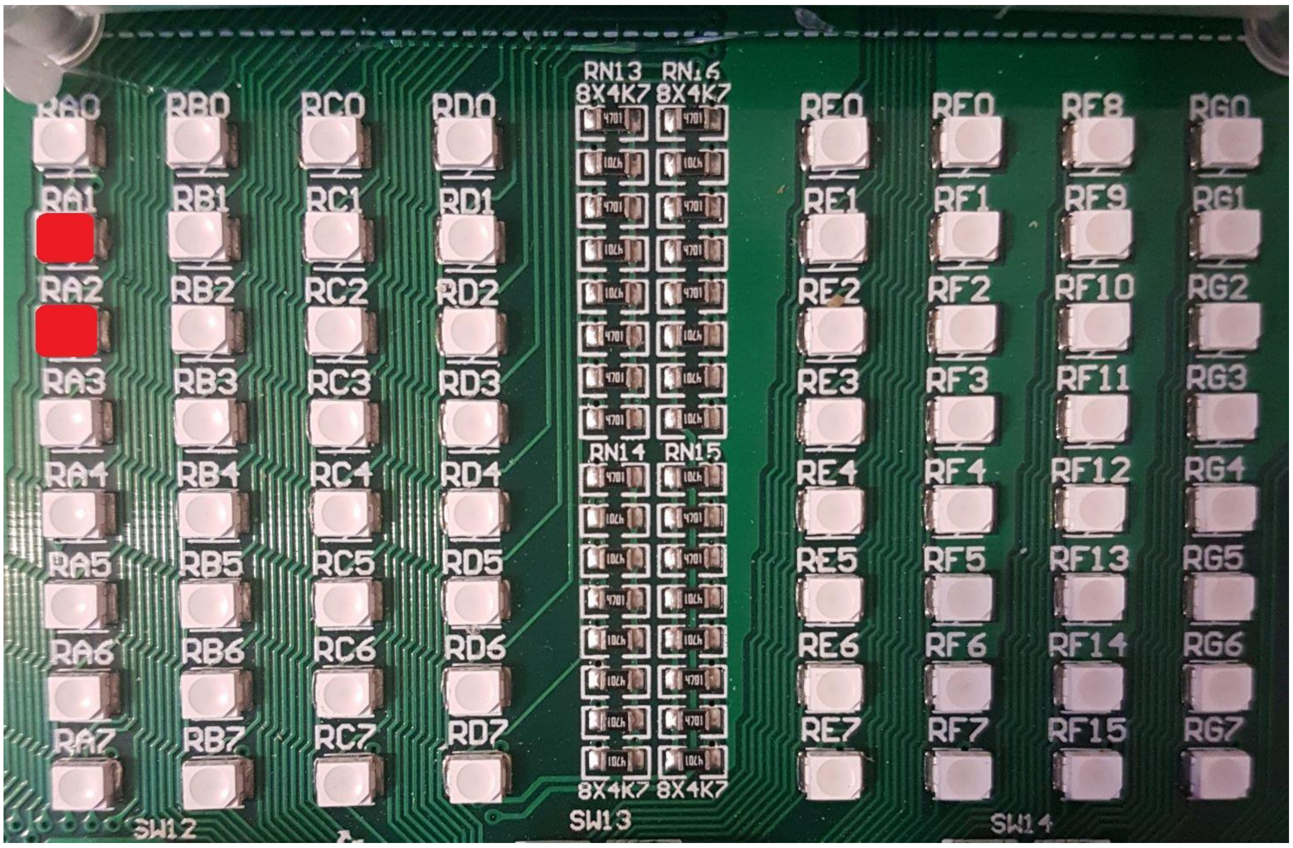
RA0 RB0 RC0 RD0 RE0 RF0 RF8 RG0
RA1 RB1 RC1 RD1 RE1 RF1 RF9 RG1
RA2 RB2 RC2 RD2 RE2 RF2 RF10 RG2
RA3 RB3 RC3 RD3 RE3 RF3 RF11 RG3
RA4 RB4 RC4 RD4 RE4 RF4 RF12 RG4
RA5 RB5 RC5 RD5 RE5 RF5 RF13 RG5
RA6 RB6 RC6 RD6 RE6 RF6 RF14 RG6
RA7 RB7 RC7 RD7 RE7 RF7 RF15 RG7
RN13 8X4K7 RN16 8X4K7
RN14 RN15
8X4K7 8X4K7
SW12 SW13 SW14

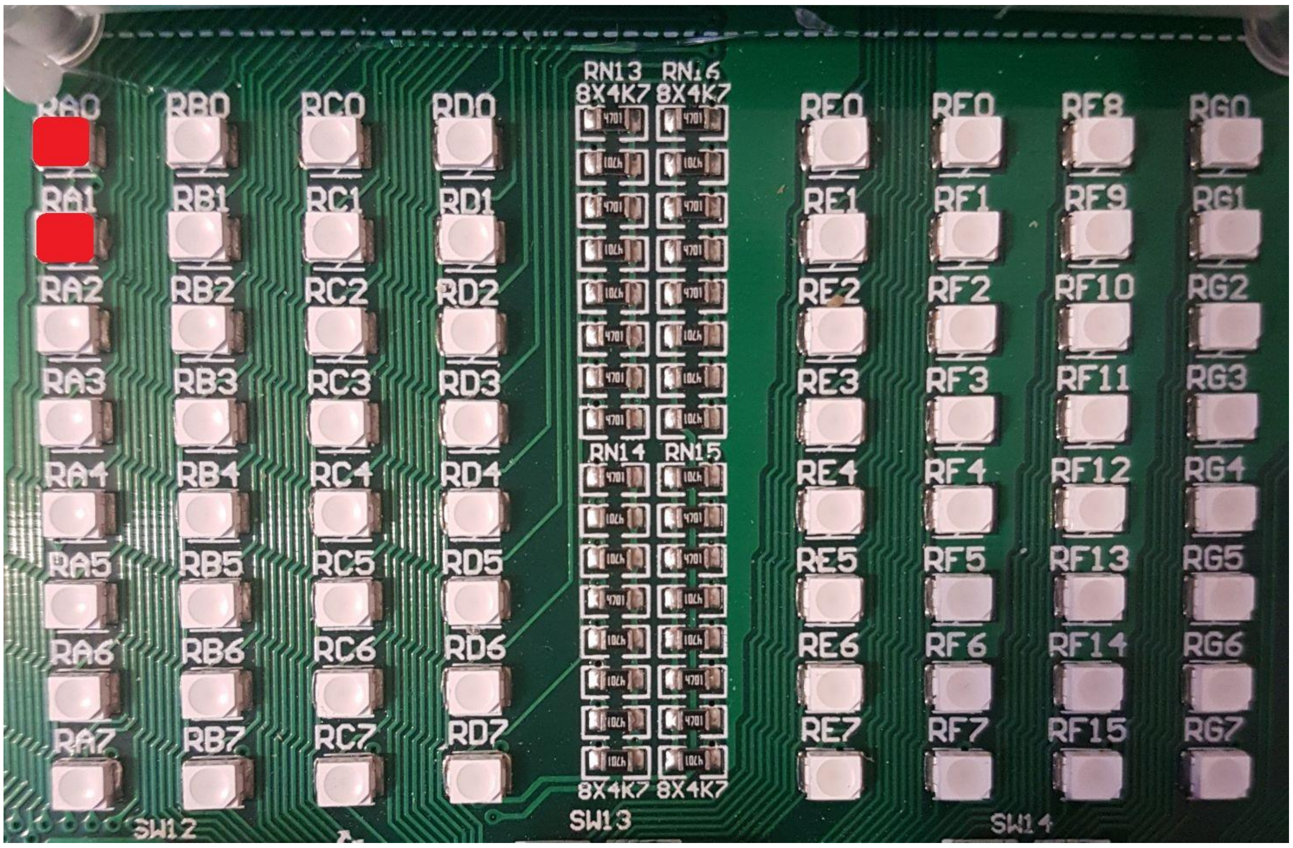

RA0 RA1 RA2 RA3 RA4 RA5 RA6 RA7
RB0 RB1 RB2 RB3 RB4 RB5 RB6 RB7
RC0 RC1 RC2 RC3 RC4 RC5 RC6 RC7
RD0 RD1 RD2 RD3 RD4 RD5 RD6 RD7
RN13 RN16
8X4K7 8X4K7
RN14 RN15
8X4K7 8X4K7
RE0 RE1 RE2 RE3 RE4 RE5 RE6 RE7
RF0 RF1 RF2 RF3 RF4 RF5 RF6 RF7
RF8 RF9 RF10 RF11 RF12 RF13 RF14 RF15
RG0 RG1 RG2 RG3 RG4 RG5 RG6 RG7
SW12
SW13
SW14


RA0 RA1 RA2 RA3 RA4 RA5 RA6 RA7
RB0 RB1 RB2 RB3 RB4 RB5 RB6 RB7
RC0 RC1 RC2 RC3 RC4 RC5 RC6 RC7
RD0 RD1 RD2 RD3 RD4 RD5 RD6 RD7
RN13 RN16
8X4K7 8X4K7
RN14 RN15
8X4K7 8X4K7
RE0 RE1 RE2 RE3 RE4 RE5 RE6 RE7
RF0 RF1 RF2 RF3 RF4 RF5 RF6 RF7
RF8 RF9 RF10 RF11 RF12 RF13 RF14 RF15
RG0 RG1 RG2 RG3 RG4 RG5 RG6 RG7
SW12
SW13
SW14


RA0
RB0
RC0
RD0
RN13
RN16
8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RN14
RN15
RA4
RB4
RC4
RD4
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
SW12
SW13
SW14


RA0
RB0
RC0
RD0
RN13
RN16
8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RN14
RN15
RA4
RB4
RC4
RD4
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
SW12
SW13
SW14


RA0
RB0
RC0
RD0
RN13
RN16
8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RA4
RB4
RC4
RD4
RN14
RN15
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
SW12
SW13
SW14


RA0
RB0
RC0
RD0
RN13
RN16
8X4K7
RE0
RF0
RF8
RG0
RA1
RB1
RC1
RD1
RE1
RF1
RF9
RG1
RA2
RB2
RC2
RD2
RE2
RF2
RF10
RG2
RA3
RB3
RC3
RD3
RE3
RF3
RF11
RG3
RA4
RB4
RC4
RD4
RN14
RN15
RE4
RF4
RF12
RG4
RA5
RB5
RC5
RD5
RE5
RF5
RF13
RG5
RA6
RB6
RC6
RD6
RE6
RF6
RF14
RG6
RA7
RB7
RC7
RD7
RE7
RF7
RF15
RG7
SW12
SW13
SW14