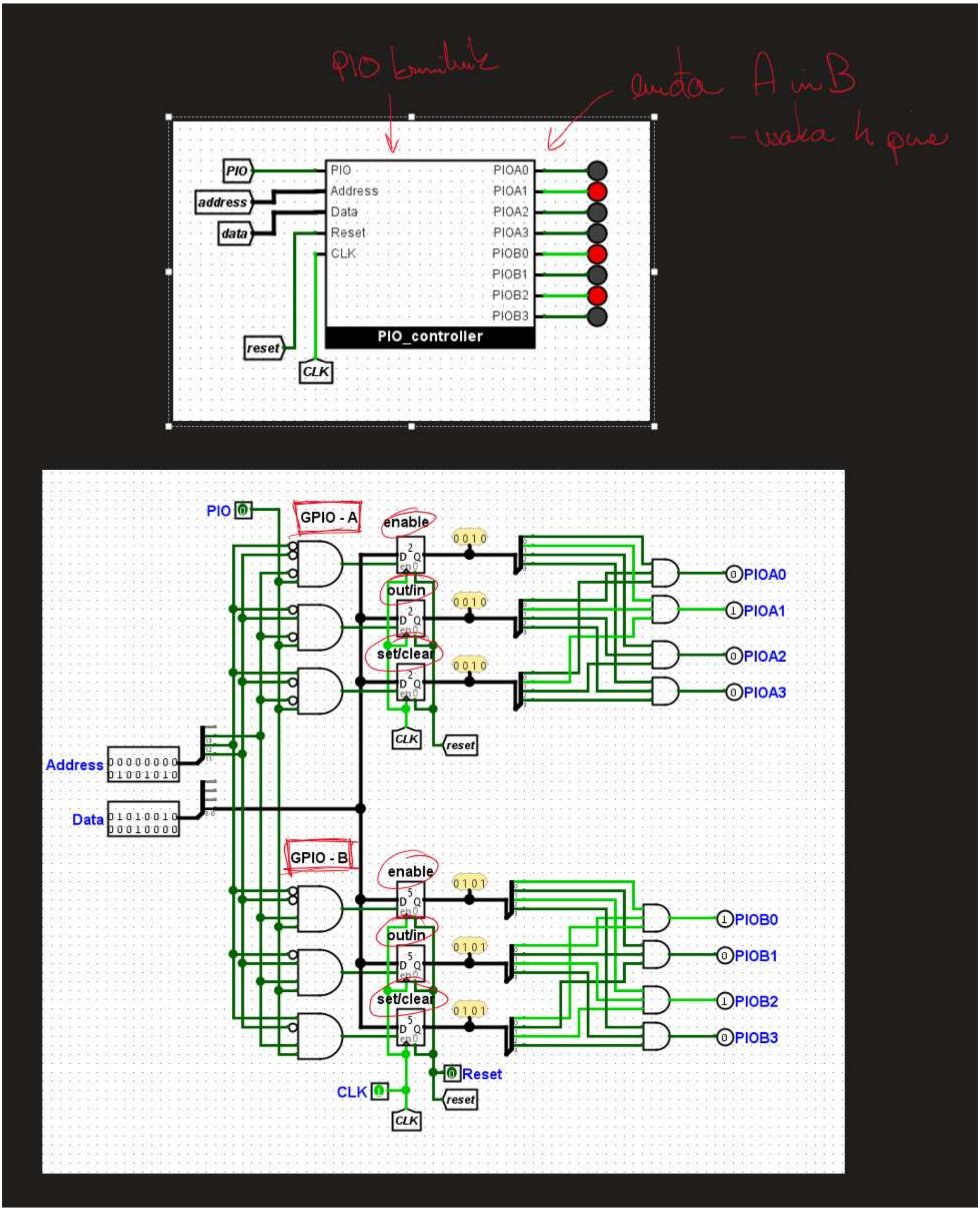


Realizacija PIO kontrolerja, podobno kot FRI-SMS.
Sestavljajo ga 6 registrov (3 registre PIOA, 3 registri PIOB) in 8 naprav (pinov 0,1,2,3)

- **enable register** (omogoci pin 0,1,2,3)
- **output/input register** (doloci vhod/izhod pina 0,1,2,3)
- **set/clear register** (aktivirja/porbisi pin 0,1,2,3)

Address

- bit 15,14 (11) - PIO controller

- bit 13 = 0 (PIOA)
- bit 13 = 1 (PIOB)
- bit 12,11 (00) - enable interface
- bit 12,11 (01) - output(1)/input(0)
- bit 12,11 (10) - set(1)/clear(0)

Register

- bit 3,2,1,0 - pin 3,2,1,0

BASE_PIOA = 0b1100000000000000 = 49152
BASE_PIOB = 0b1110000000000000 = 57344

ENABLE_INTERFACE = BASE + 0
OUT_IN = BASE + 2048
SET_CLR = BASE + 4096

example PIOA1

```
enable interface 0b0000000000000010 = 2
enable output    0b0010000000000010 = 8194
set output 1     0b0100000000000010 = 16386
```

example PIOB0,2

```
enable interface 0b1000000000000101 = 32773
enable output    0b1010000000000101 = 40965
set output 1     0b1100000000000101 = 49157
```

INIT_PIO

```
INIT_PIO:   push    r0
            push    r1
            push    r2

            # BASE_PIOA = 49152
            # BASE_PIOB = 57344

            li      r0, 49152
            li      r1, 57344

            # ENABLE_INTERFACE = BASE + 0
            # OUT_IN           = BASE + 2048
            # SET_CLR          = BASE + 4096

            li      r2, 2           # 0b1001 (pin1)

            swi r2, r0, 0      # enable PIOA pin1
            swi r2, r0, 2048   # output PIOA pin1
            swi r2, r0, 4096   # set PIOA pin1

            li      r2, 5           # 0b1001 (pin2,0)

            swi r2, r1, 0      # enable PIOB pin2,0
            swi r2, r1, 2048   # output PIOB pin2,0
            swi r2, r1, 4096   # set PIOB pin2,0

            pop    r2
            pop    r1
            pop    r0

            rts
```

PIN_ON

```
PIN_ON:     push r2 # vrednost registra set/clear
            push r3 # rezultat
            # r0 -> inpit PIO address
            # r1 -> input pin

            lwi     r2, r0, 4096
            or      r3, r2, r1
            swi     r3, r0, 4096

            pop r3
            pop r2
            rts
```

PIN_OFF

```
PIN_OFF:    push r2 # vrednost registra set/clear
            push r3 # rezultat
            # r0 -> inpit PIO address
            # r1 -> input pin
```

```
              lwi     r2, r0, 4096
              nor      r3, r2, r1
              swi     r3, r0, 4096

              pop r3
              pop r2
              rts
```

N5

Program

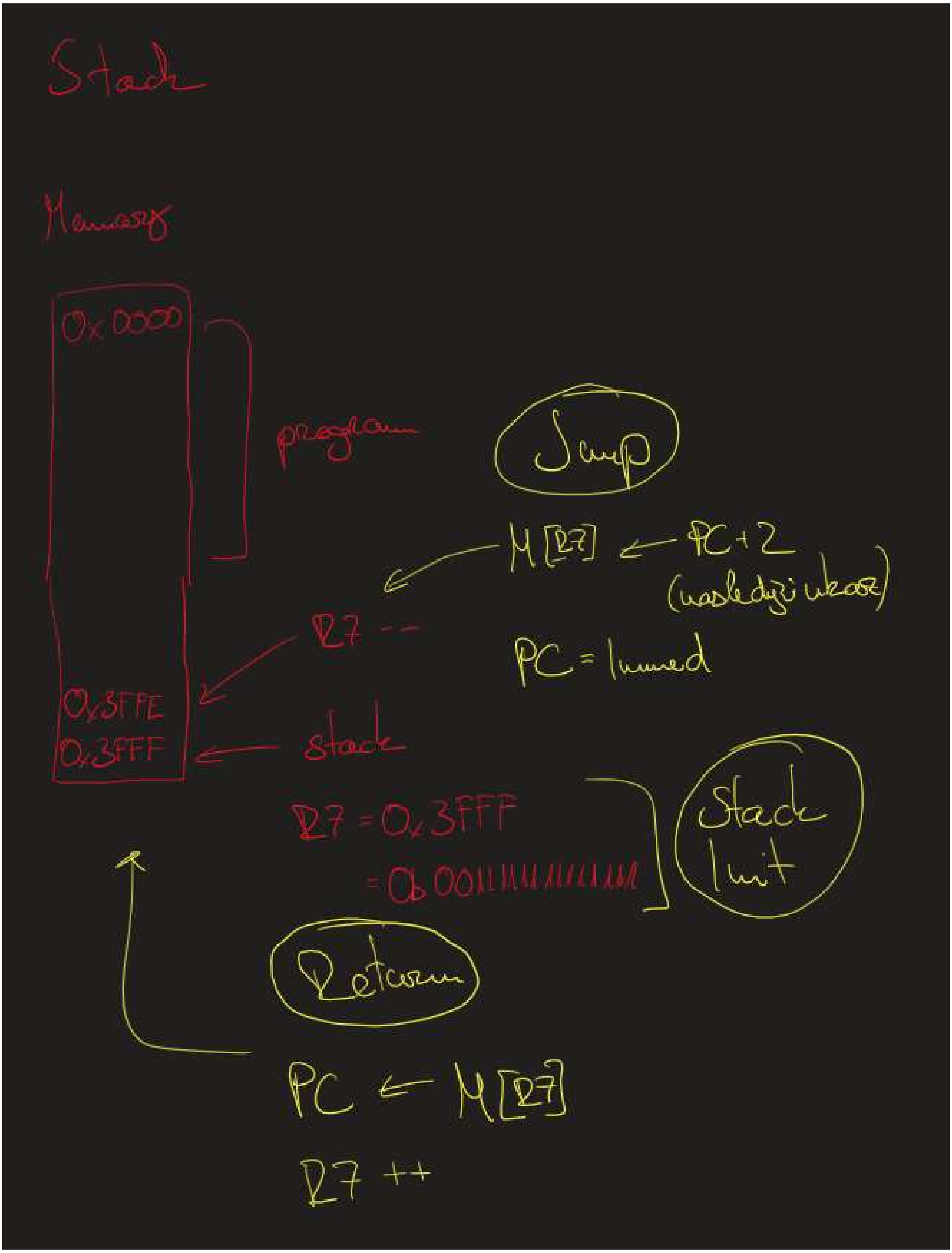
Najprej inicializira PIO naprave.
Nato v zanki priziga (ON) in ugasa (OFF) pin1 (PIOA) z delayom (DELAY)

Podporogrami

- INIT_PIO - inicializira PIO naprave
- PIN_ON - set pin (INPUT: r0 -> BASE_PIO address, r1 -> pin)
- PIN_OFF - clear pin (INPUT: r0 -> BASE_PIO address, r1 -> pin)
- DELAY - loop 30 times

Ukazi

- inicializacija sklada (li r7, 16383 # 0x3FFF)
- jsr - skok v podporgram (naslov shranjen v Link register - R7)
- rts - vrnitev iz podporgrama (vrednsot na naslovu R7 vnesena v PC)
- push - dodajanje registreov na sklad (ohranjenje regstrov)
- pop - jemanje registrov s sklada (obnavljanje registrov)
- lwi - load from M[r + immed]
- swi - strore to M [r + immed]



```
        li      r7, 16383 # 0x3FFF - stack init


main:   jsr INIT_PIO
        li      r5,  30


loop:   dec     r5
```

```
        li      r0, 49152      # BASE_PIOA
        li      r1, 2           # pin 1

        jsr     PIN_OFF
        jsr     DELAY

        jsr     PIN_ON
        jsr     DELAY

        jgtz    r5, loop


inf:    jmp inf                     # infinity loop

INIT_PIO:   push    r0
            push    r1
            push    r2

            # BASE_PIOA = 49152
            # BASE_PIOB = 57344

            li      r0, 49152
            li      r1, 57344

            # ENABLE_INTERFACE = BASE + 0
            # OUT_IN           = BASE + 2048
            # SET_CLR          = BASE + 4096

            li      r2, 2           # 0b1001 (pin1)

            swi r2, r0, 0      # enable PIOA pin1
            swi r2, r0, 2048   # output PIOA pin1
            swi r2, r0, 4096   # set PIOA pin1

            li      r2, 5           # 0b1001 (pin2,0)

            swi r2, r1, 0      # enable PIOB pin2,0
            swi r2, r1, 2048   # output PIOB pin2,0
            swi r2, r1, 4096   # set PIOB pin2,0

            pop    r2
            pop    r1
            pop    r0
            rts

PIN_ON:     push r2 # vrednost registra set/clear
            push r3 # rezultat
            # r0 -> input PIO address
            # r1 -> input pin

            lwi     r2, r0, 4096
            or      r3, r2, r1
            swi     r3, r0, 4096

            pop r3
            pop r2
            rts

PIN_OFF:    push r2 # vrednost registra set/clear
            push r3 # rezultat
            # r0 -> input PIO address
            # r1 -> input pin

            lwi     r2, r0, 4096
            nor      r3, r2, r1
            swi     r3, r0, 4096

            pop r3
            pop r2
            rts


DELAY:      push    r2
            li      r2, 30

loop1:      dec     r2
            jgtz    r2, loop1

            pop     r2
            rts
```