

Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva

vili

**Druga domaća zadaća iz predmeta  
"Uvod u teoriju računarstva"**

Zadatak broj 2001

Zagreb, lipanj 2009.

## Druga domaća zadaća iz predmeta "Uvod u teoriju računarstva"

**Student:** vili

**Matični broj studenta:** 1234567890

**Zadatak broj 2001:**

Izgraditi TS koji će simulirati rad računala. Računalo ima 2 registra (r1 i r2) i programsko brojilo (PC). Radna se memorija sastoji od 16 lokacija sa 8 bitnim brojevima. Računalo razumije sljedeće instrukcije:

- LOAD ri, A : učitava podatak sa adrese A u registar ri
- STORE ri, A : pohranjuje podatak iz registra ri na memorijsku lokaciju A
- ADD ri, rj :  $r_i + r_j = r_i$
- SUB ri, rj :  $r_i - r_j = r_i$

Sve instrukcije odnose se na cjelobrojne brojeve bez predznaka. Pokaži izvođenje programa:

```
LOAD r1, 11
LOAD r2, 12
ADD r1, r2
STORE r2, 12
```

# 1. Uvod

Turingov stroj (u nastavku TS) je najopćenitiji poznati matematički model računanja pomoću kojeg je moguće opisati i riješiti vrlo komplicirane problemske zadatke. Osnovni model TS-a se sastoji od upravljačke jedinice, ulazne trake i glave za čitanje i pisanje. Upravljačka jedinica može poprimiti konačan broj stanja, ovisno o funkcijama prijelaza. Ulazna traka je omeđena s lijeve, a beskonačna s desne strane i podijeljena na ćelije u koje se zapisuje po jedan od konačnog broja znakova, opet ovisno o funkcijama prijelaza. Glava za čitanje i pisanje se miče od ćelije do ćelije, čita znakove u njima i, ovisno o odgovarajućoj funkciji prijelaza, zapisuje znak u pročitano ćeliju i nakon toga se pomiče ulijevo ili udesno.

Opisani osnovni model može se, ovisno o potrebama konkretnog zadatka, proširiti na više načina. Moguće ga je proširiti tako da umjesto trake omeđene s jedne strane, a beskonačne s druge, imamo traku koja je beskonačna s obje strane. Moguće je koristiti i više ulaznih traka, naravno, svaka traka sa vlastitom glavom za čitanje i pisanje. Također je moguće imati više glava za čitanje i pisanje na jednoj traci, tako da glave rade praktički neovisno jedna o drugoj, oviseći samo o funkcijama prijelaza. Također je moguće ne micati glavu za čitanje i pisanje. Spomenuta proširenja TS-a je, naravno, moguće kombinirati.

Funkcije prijelaza su temelj svakog TS-a. Pomoću njih TS "zna" u koje stanje upravljačka jedinica treba prijeći, koji znak treba zapisati na mjesto pročitano znaka i u kojem se smjeru glava za čitanje i pisanje treba pomaknuti.

**Osnovni** model TS-a ima funkcije prijelaza oblika:

$$\delta(q_0, \alpha) = (q_1, \beta, \lambda),$$

gdje je:

- $q_0$  - stanje u kojem se TS trenutno nalazi
- $\alpha$  - znak u ćeliji iznad koje se glava za čitanje i pisanje trenutno nalazi
- $q_1$  - stanje u koje TS prelazi
- $\beta$  - znak koji glava za čitanje i pisanje zapisuje na mjesto pročitano znaka
- $\lambda$  - smjer u kojem se glava za čitanje i pisanje pomiče, L za lijevo, R za desno

**Prošireni** model TS-a sa  $k$  glava za čitanje i pisanje ima funkcije prijelaza oblika:

$$\delta(q_0, \{\alpha_1, \dots, \alpha_k\}) = (q_1, \{\beta_1, \dots, \beta_k\}, \{\lambda_1, \dots, \lambda_k\}),$$

gdje je:

- $q_0$  - stanje u kojem se TS trenutno nalazi
- $\alpha_k$  - znak u ćeliji iznad koje se glava  $k$  trenutno nalazi
- $q_1$  - stanje u koje TS prelazi
- $\beta_k$  - znak koji glava  $k$  zapisuje na mjesto pročitano znaka
- $\lambda_k$  - smjer u kojem se glava  $k$  pomiče, L za lijevo, R za desno, S za ostanak na istoj ćeliji

Kod definiranja TS-a, prije funkcija prijelaza potrebno je odrediti skup stanja, skup ulaznih znakova, skup znakova trake, početno stanje, skup prihvatljivih stanja i konačno, funkcije prijelaza. Ovaj način definiranja TS-a je u praksi teško ostvariv, jer prilikom same izgradnje nerijetko je potrebno dodati neko novo stanje, neki novi znak trake i sl., tako da se gore spomenuti skupovi često proširuju.

## 2. Ostvarenje

Na prvi pogled zadani zadatak izgleda u najmanju ruku kompliciran, no ta kompliciranost se može lako riješiti tako da se zadatak razloži na više jednostavnijih podzadataka i na kraju se sve spoji u jednu cjelinu koja rješava početni zadatak.

Kao prvo, moramo se odlučiti koji ćemo model TS-a koristiti. Sasvim je očito da nam osnovni model neće biti dovoljan za sve mogućnosti koje moramo implementirati, tako da se odlučujemo za prošireni model. Daljnjim razmatranjem problema dolazimo do odabira broja traka i pripadajućih glava. Odlučio sam se za brojku od četiri trake, s time da prva traka koja predstavlja radnu memoriju ima dvije glave za čitanje i pisanje. Pošto se na toj traci nalaze i naredbe i podaci, jednom glavom čitamo i dekodiramo naredbe, a drugom glavom zapisujemo ili čitamo podatke s te trake. Detaljnije ću objasniti taj princip u dijelu u kojem ću opisati funkcioniranje LOAD / STORE naredbi. Druga traka će predstavljati registar r1, a treća registar r2. Odvojene trake za registre su nužne za jednostavnije implementiranje ADD i SUB naredbi kod kojih će biti potrebno istovremeno čitanje oba registra. Četvrta traka će predstavljati PC (program counter). Ovdje dolazim do prve **dileme**: pošto je u zadatku navedeno da je potrebno implementirati samo četiri osnovne naredbe koje će se izvršavati onim redoslijedom kojim su zapisane u memoriju, dakle slijedno, bez ikakvih skokova, PC nam služi samo za brojanje izvedenih naredbi, pa smatram da nije potrebno ostvariti povratnu vezu između PC-a i glave za čitanje naredbi. Da je u zadatku zadano da treba implementirati i naredbe skokova, recimo naredbu GOTO ili razne petlje, dodatnom skupinom prijelaza bih ostvario povratnu vezu između PC-a i druge glave, no pošto nije potrebno ostvariti te naredbe, na kraju izvršavanja svake naredbe samo povećamo PC za jedan.

Prvi problem s kojim se suočavamo jest prepoznavanje naredbi koje su zapisane kao 8 bitni brojevi. Naredbe su u memoriju kodirane na sljedeći način:

Prva dva bita označavaju tip naredbe:

- 00 - LOAD
- 01 - STORE
- 10 - ADD
- 11 - SUB

Sljedeći bit je u svim naredbama 0, a bit nakon njega označava broj registra:

- 0 - registar r1
- 1 - registar r2

ostala 4 bita su ovisna o tipu naredbe:

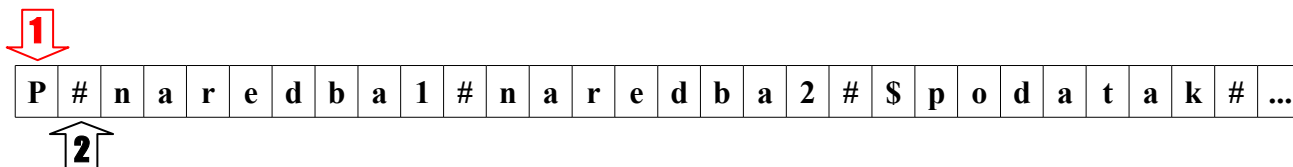
- kod LOAD / STORE naredbi označavaju adresu memorijske lokacije u binarnom obliku
- kod ADD / SUB naredbi sljedeća 3 bita su 0, a zadnji bit označava broj registra:

- 0 - registar r1
- 1 - registar r2

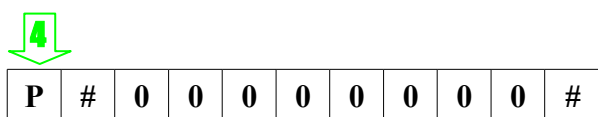
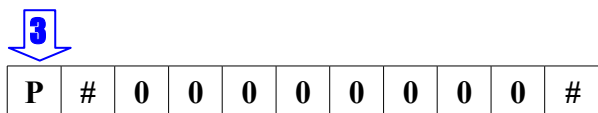
Sada kada smo definirali način na koji su naredbe kodirane u memoriju moramo konstruirati prijelaze za svaku pojedinu naredbu, no prije toga moramo definirati početno stanje svake trake i svake glave.

## 2.1 Početni izgled traka i položaj glava

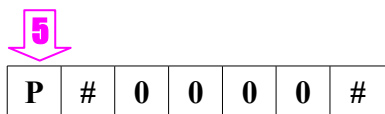
Prva traka predstavlja radnu memoriju. Na njenom početku je znak P. Slijedi znak #, pa zatim blokovi od po 8 bitova odijeljeni znakom #, a graničnik instrukcijskog i podatkovnog dijela memorije je, nakon znaka #, znak \$. Instrukcijski dio može sadržavati proizvoljni broj blokova od 8 bitova. Radna memorija završava također znakom #. Naredbe su kodirane na način opisan u prethodnom dijelu. Prva glava (**crvena**), koja će se kretati po cijeloj memoriji na početku pokazuje na znak P, a druga glava (**crna**), koja samo dohvaća naredbe na početku pokazuje na drugi znak trake, znak #:



Druga i treća traka koje predstavljaju registre r1 i r2 također počinju znakom P, a nakon njega slijedi blok od 8 bitova koji sa svake strane ima znak #. Treća glava (**plava**) se kreće po traci r1, a četvrta glava (**zelená**) po traci r2. Glave za čitanje i pisanje koje odgovaraju navedenim trakama pokazuju na prvo mjesto, tj. na znak P. Na svakoj od ovih traka koristit ćemo samo prvih 11 ćelija:



Četvrta traka predstavlja PC, a počinje znakom P. Slijede 4 znaka 0 koji predstavljaju adresu prvog bloka memorije. Pošto trebamo moći adresirati 16 lokacija, 4 bita su nam sasvim dovoljna. Znak # se nalazi s obje strane, prema tome, ukupno koristimo 7 ćelija ove trake. Glava četvrte trake (**magneta**) pokazuje na mjesto P.



Treba napomenuti da su na početku sve ćelije ispunjene nulama, osim onih za koje je gore drugačije napisano. Nakon definiranja početnog izgleda traka i položaja glava možemo krenuti na objašnjavanje postupka i konstruiranje funkcija prijelaza za svaku pojedinu operaciju, no prije toga jedno čisto tehničko pitanje izgleda prijelaza.

## 2.2 Izgled prijelaza

Funkcije prijelaza su oblika:

$$\delta(q_0, \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}) = (q_1, \{\beta_1, \beta_2, \beta_3, \beta_4, \beta_5\}, \{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5\}),$$

Boja svakog znaka u funkciji prijelaza označava na koju se glavu odnosi. Oznčavanje bojama služi tome da se lakše vidi koji se znakovi odnose na koju glavu.

## 2.3 LOAD

### 2.3.1 Prepoznavanje naredbe LOAD

Na početku se nalazimo u stanju  $q_P$ . Drugom glavom prve trake pročitamo znak # i prelazimo u stanje  $q_N$ . Čitamo sljedeći znak i ako je on 0 prelazimo u stanje  $q_{MEM}$ , međutim ako je on \$, to znači da smo stigli do kraja instrukcijskog dijela memorije i time prelazimo u prihvatljivo stanje  $q_{END}$ .

Nakon pročitane nule opet čitamo znak 0 i prelazimo u stanje  $q_{LD}$ . U tom stanju preskačemo jednu nulu.

$$\begin{aligned}\delta(q_P, \{P, \#, P, P, P\}) &= (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_N, \{P, \$, P, P, P\}) &= (q_{END}, \{P, \$, P, P, P\}, \{S, L, S, S, S\}) \\ \delta(q_N, \{P, 0, P, P, P\}) &= (q_{MEM}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{MEM}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{LD}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})\end{aligned}$$

Zatim čitamo bit koji označava broj registra i ovisno o njemu pomičemo glavu na traci tog registra za jedno mjesto u desno. Nakon toga znak # zamjenjujemo znakom L čime označavamo naredbu LOAD, prelazimo u stanje  $q_{POZI}$  koje služi za čitanje adrese i mičemo glavu u lijevo, na znak P (kako bi minimizirali krajnji broj prijelaza):

$$\begin{aligned}\delta(q_{LD}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, S, R, S, S\}) \\ \delta(q_{LD}, \{P, 0, \#, P, P\}) &= (q_{POZI}, \{P, 0, L, P, P\}, \{R, R, L, S, R\}) \\ \delta(q_{LD}, \{P, 1, P, P, P\}) &= (q_{LD}, \{P, 1, P, P, P\}, \{S, S, S, R, S\}) \\ \delta(q_{LD}, \{P, 1, P, \#, P\}) &= (q_{POZI}, \{P, 1, P, L, P\}, \{R, R, S, L, R\})\end{aligned}$$

### 2.3.2 Čitanje adrese i pozicioniranje glave

Slijedi čitanje adrese i pozicioniranje prve trake prve glave na odgovarajuće mjesto. Na ovaj algoritam i konstruiranu skupinu prijelaza ću se referencirati prilikom definicije STORE skupine prijelaza. Postepeno ću opisivati algoritam i istovremeno navoditi prijelaze.

Privremeno ćemo koristiti drugi znak četvrte trake kao broj pročitanih bitova adrese.

Sljedeća dva prijelaza čitaju bit adrese najveće težine, i ako je on 0, prelazi se na sljedeći bit adrese, no ako je taj bit 1, to znači da prva glava mora preskočiti 8 memorijskih lokacija i prebacujemo se u pripadajuće stanje:

$$\delta(q_{POZ1}, \{\#, 0, P, P, \#\}) = (q_{POZ2}, \{\#, 0, P, P, 1\}, \{S, R, S, S, S\})$$

$$\delta(q_{POZ1}, \{\#, 1, P, P, \#\}) = (q_{ADR\_8}, \{\#, 1, P, P, 1\}, \{R, S, S, S, L\})$$

Analogne prijelaze napravimo za ostale 3 pozicije u adresi. Dvije razlike su u tome što na četvrtu traku zapisujemo odgovarajuću poziciju netom pročitano bita i što kod pročitane jedinice prelazimo u stanje za preskakanje onoliko bitova kolika je težina te jedinice:

$$\delta(q_{POZ2}, \{\#, 0, P, P, 1\}) = (q_{POZ3}, \{\#, 0, P, P, 2\}, \{S, R, S, S, S\})$$

$$\delta(q_{POZ2}, \{\#, 1, P, P, 1\}) = (q_{ADR\_4}, \{\#, 1, P, P, 2\}, \{R, S, S, S, L\})$$

$$\delta(q_{POZ3}, \{\#, 0, P, P, 2\}) = (q_{POZ4}, \{\#, 0, P, P, 3\}, \{S, R, S, S, S\})$$

$$\delta(q_{POZ3}, \{\#, 1, P, P, 2\}) = (q_{ADR\_2}, \{\#, 1, P, P, 3\}, \{R, S, S, S, L\})$$

$$\delta(q_{POZ4}, \{\#, 0, P, P, 3\}) = (q_{ADR\_END}, \{\#, 0, P, P, \#\}, \{S, R, S, S, S\})$$

$$\delta(q_{POZ4}, \{\#, 1, P, P, 3\}) = (q_{ADR\_1}, \{\#, 1, P, P, 4\}, \{R, S, S, S, S\})$$

Sljedi skupina prijelaza koja se sastoji od preskakanja nula i jedinica (i eventualno znaka \$) i prelaska u niže stanje pročitanim znakom #:

$$\delta(q_{ADR8}, \{0, 1, P, P, P\}) = (q_{ADR8}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR8}, \{1, 1, P, P, P\}) = (q_{ADR8}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR8}, \{\$, 1, P, P, P\}) = (q_{ADR8}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR8}, \{\#, 1, P, P, P\}) = (q_{ADR7}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{0, 1, P, P, P\}) = (q_{ADR7}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{1, 1, P, P, P\}) = (q_{ADR7}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{\$, 1, P, P, P\}) = (q_{ADR7}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{\#, 1, P, P, P\}) = (q_{ADR6}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{0, 1, P, P, P\}) = (q_{ADR6}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{1, 1, P, P, P\}) = (q_{ADR6}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{\$, 1, P, P, P\}) = (q_{ADR6}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{\#, 1, P, P, P\}) = (q_{ADR5}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{0, 1, P, P, P\}) = (q_{ADR5}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{1, 1, P, P, P\}) = (q_{ADR5}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{\$, 1, P, P, P\}) = (q_{ADR5}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{\#, 1, P, P, P\}) = (q_{ADR4}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{0, 1, P, P, P\}) = (q_{ADR4}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{1, 1, P, P, P\}) = (q_{ADR4}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{\$, 1, P, P, P\}) = (q_{ADR4}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{\#, 1, P, P, P\}) = (q_{ADR3}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{0, 1, P, P, P\}) = (q_{ADR3}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{1, 1, P, P, P\}) = (q_{ADR3}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{\$, 1, P, P, P\}) = (q_{ADR3}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{\#, 1, P, P, P\}) = (q_{ADR2}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

U sljedećem skupu stanja radimo istu stvar kao i u pretnodnim. Jedina razlika je da petu glavu mićemo udesno kako bi u narednom skupu stanja mogli pročitati bit koji označava poziciju bita adrese. Navedenu petu glavu smo u prvoj skupini prijelaza pomaknuli ulijevo, na znak P kako bi minimizirali konačan broj prijelaza.

$$\delta(q_{ADR2}, \{0, 1, P, P, P\}) = (q_{ADR2}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{1, 1, P, P, P\}) = (q_{ADR2}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{\$, 1, P, P, P\}) = (q_{ADR2}, \{\$, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{\#, 1, P, P, P\}) = (q_{ADR1}, \{\#, 1, P, P, P\}, \{R, S, S, S, R\})$$

Kod zadnje skupine prijelaza pročitanim znakom # prelazimo u ono stanje na koje indicira bit na četvrtoj traci:

$$\delta(q_{ADR1}, \{0, 1, P, P, 1\}) = (q_{ADR1}, \{0, 1, P, P, 1\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{1, 1, P, P, 1\}) = (q_{ADR1}, \{1, 1, P, P, 1\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\$, 1, P, P, 1\}) = (q_{ADR1}, \{\$, 1, P, P, 1\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 1\}) = (q_{POZ2}, \{\#, 1, P, P, 1\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADR1}, \{0, 1, P, P, 2\}) = (q_{ADR1}, \{0, 1, P, P, 2\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{1, 1, P, P, 2\}) = (q_{ADR1}, \{1, 1, P, P, 2\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\$, 1, P, P, 2\}) = (q_{ADR1}, \{\$, 1, P, P, 2\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 2\}) = (q_{POZ3}, \{\#, 1, P, P, 2\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADR1}, \{0, 1, P, P, 3\}) = (q_{ADR1}, \{0, 1, P, P, 3\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{1, 1, P, P, 3\}) = (q_{ADR1}, \{1, 1, P, P, 3\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\$, 1, P, P, 3\}) = (q_{ADR1}, \{\$, 1, P, P, 3\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 3\}) = (q_{POZ4}, \{\#, 1, P, P, 3\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADR1}, \{0, 1, P, P, 4\}) = (q_{ADR1}, \{0, 1, P, P, 4\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{1, 1, P, P, 4\}) = (q_{ADR1}, \{1, 1, P, P, 4\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\$, 1, P, P, 4\}) = (q_{ADR1}, \{\$, 1, P, P, 4\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 4\}) = (q_{ADR\_END}, \{\#, 1, P, P, \#\}, \{S, R, S, S, S\})$$

Zadnjim prijelazom vraćamo drugi bit četvrte trake u početno stanje (#) i spremni smo za glavni dio naredbe LOAD: kopiranje podatka iz memorije u odgovarajući registar.

### 2.3.3 Kopiranje podatka iz memorije u registar

Na kraju prethodne operacije nalazimo se u stanju  $q_{ADR\_END}$  koje je zajedničko LOAD i STORE operaciji, pa moramo prijeći u stanje koje izvršava operaciju koju želimo. U dijelu 2.3.1 smo konstruirali prijelaze za upisivanje znakova L i S u registre r1 i r2. Upravo ti znakovi će nam pomoći da prijeđemo u pravo stanje. Glave oba registra pomaknemo udesno. Čitamo znak koji je zapisan na drugom mjestu svakog registra i ovisno o njemu ulazimo u jedno od četiri stanja. Pošto



sada opisujemo LOAD naredbu koncentrirat ćemo se na prva dva stanja:  $q_{LDR1}$  i  $q_{LDR2}$ . Ovisno u kojem smo stanju završili, glavu registra koji je sadržan u imenu trenutnog stanja pomičemo udesno na prvi bit sadržaja registra, a glavu drugog registra pomičemo ulijevo, na znak P. Treba napomenuti da glave oba registra na drugo mjesto zapisuju znak #. Slijedi najlakši dio posla, ujedno i glavni dio LOAD naredbe: kopiranje podatka iz memorije u odgovarajući registar. Prije kopiranja moramo paziti na znak \$ koji preskačemo:

Početni prijelaz:

$$\delta(q_{ADR\_END}, \{\#, \#, P, P, P\}) = (q_{ADR\_END}, \{\#, \#, P, P, P\}, \{S, S, R, R, S\})$$

Punjenje registra R1:

$$\delta(q_{ADR\_END}, \{\#, \#, L, \#, P\}) = (q_{LDR1}, \{\#, \#, \#, \#, P\}, \{R, S, R, L, S\})$$

$$\delta(q_{LDR1}, \{\$, \#, 0, P, P\}) = (q_{LDR1}, \{\$, \#, 0, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{LDR1}, \{\$, \#, 1, P, P\}) = (q_{LDR1}, \{\$, \#, 1, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{LDR1}, \{0, \#, 0, P, P\}) = (q_{LDR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{LDR1}, \{0, \#, 1, P, P\}) = (q_{LDR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{LDR1}, \{1, \#, 0, P, P\}) = (q_{LDR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{LDR1}, \{1, \#, 1, P, P\}) = (q_{LDR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

Vraćanje glave registra R1 na početak:

$$\delta(q_{LDR1}, \{\#, \#, \#, P, P\}) = (q_{R1\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, 0, P, P\}) = (q_{R1\_BACK}, \{\#, \#, 0, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, 1, P, P\}) = (q_{R1\_BACK}, \{\#, \#, 1, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, \#, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, \#, P, P\}, \{L, S, L, S, S\})$$

Punjenje registra R2:

$$\delta(q_{ADR\_END}, \{\#, \#, \#, L, P\}) = (q_{LDR2}, \{\#, \#, \#, \#, P\}, \{R, S, L, R, S\})$$

$$\delta(q_{LDR2}, \{\$, \#, P, 0, P\}) = (q_{LDR2}, \{\$, \#, P, 0, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{LDR2}, \{\$, \#, P, 1, P\}) = (q_{LDR2}, \{\$, \#, P, 1, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{LDR2}, \{0, \#, P, 0, P\}) = (q_{LDR2}, \{0, \#, P, 0, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{LDR2}, \{0, \#, P, 1, P\}) = (q_{LDR2}, \{0, \#, P, 0, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{LDR2}, \{1, \#, P, 0, P\}) = (q_{LDR2}, \{1, \#, P, 1, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{LDR2}, \{1, \#, P, 1, P\}) = (q_{LDR2}, \{1, \#, P, 1, P\}, \{R, S, S, R, S\})$$

Vraćanje glave registra R2 na početak:

$$\delta(q_{LDR2}, \{\#, \#, \#, P, P\}) = (q_{R2\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, S, L, S\})$$

$$\delta(q_{R2\_BACK}, \{\#, \#, P, 0, P\}) = (q_{R2\_BACK}, \{\#, \#, P, 0, P\}, \{S, S, S, L, S\})$$

$$\delta(q_{R2\_BACK}, \{\#, \#, P, 1, P\}) = (q_{R2\_BACK}, \{\#, \#, P, 1, P\}, \{S, S, S, L, S\})$$

$$\delta(q_{R2\_BACK}, \{\#, \#, P, \#, P\}) = (q_{ADR\_BACK}, \{\#, \#, P, \#, P\}, \{L, S, S, L, S\})$$

Vraćanje prve glave na početak:

$$\delta(q_{ADR\_BACK}, \{0, \#, P, P, P\}) = (q_{ADR\_BACK}, \{0, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{1, \#, P, P, P\}) = (q_{ADR\_BACK}, \{1, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{\#, \#, P, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{\$, \#, P, P, P\}) = (q_{ADR\_BACK}, \{\$, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{P, \#, P, P, P\}) = (q_{PC\_POZI}, \{P, \#, P, P, P\}, \{S, S, S, S, R\})$$

### 2.3.4 Završni radovi

Na kraju rada LOAD, pa tako i svake naredbe, moramo na traku koja predstavlja PC upisati broj za jedan veći od prethodno upisanog.

Pozicioniranje na zadnji bit PC-a:

$$\begin{aligned}\delta(q_{PC\_POZI}, \{P, \#, P, P, \#\}) &= (q_{PC\_POZ}, \{P, \#, P, P, \#\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, 0\}) &= (q_{PC\_POZ}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, 1\}) &= (q_{PC\_POZ}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, \#\}) &= (q_{PC\_1}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

Slijedi opis i prijelazi algoritma za povećanje binarnog broja za jedan. Kada smo se pozicionirali na krajnje desni bit broja, provjeravamo njegovu vrijednost. Ako je taj bit nula, umjesto njega zapisujemo 1 i vraćamo se na početak trake:

$$\begin{aligned}\delta(q_{PC\_1}, \{P, \#, P, P, 0\}) &= (q_{PC\_BACK}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_BACK}, \{P, \#, P, P, 0\}) &= (q_{PC\_BACK}, \{P, \#, P, P, 0\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_BACK}, \{P, \#, P, P, 1\}) &= (q_{PC\_BACK}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_BACK}, \{P, \#, P, P, \#\}) &= (q_P, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

No, ako je taj bit jedan, pomičemo se ulijevo sve dok ne nađemo na bit 0 i promijenimo ga u 1.

$$\begin{aligned}\delta(q_{PC\_1}, \{P, \#, P, P, 1\}) &= (q_{PC\_2}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_2}, \{P, \#, P, P, 1\}) &= (q_{PC\_2}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_2}, \{P, \#, P, P, 0\}) &= (q_{PC\_2\_END}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\})\end{aligned}$$

Nakon toga mijenjamo sve bitove desno od njega iz jedinice u nulu. Kada smo došli do krajnje desnog bita broja, postupak je gotov i pomoću gore opisanog stanja  $q_{PC\_BACK}$  se vraćamo na početak četvrte trake i prebacujemo u stanje  $q_P$  koje služi za čitanje nove naredbe.

$$\begin{aligned}\delta(q_{PC\_2\_END}, \{P, \#, P, P, 1\}) &= (q_{PC\_2\_END}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_2\_END}, \{P, \#, P, P, \#\}) &= (q_{PC\_BACK}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

## 2.4 STORE

### 2.4.1 Prepoznavanje naredbe STORE

Kao i kod naredbe LOAD, tako se i ovdje nalazimo u stanju  $q_P$ . Drugom glavom prve trake pročitamo znak # i prelazimo u stanje  $q_N$ . Ovisno je li sljedeći znak \$ prelazimo u prihvatljivo stanje, inače čitamo 0 i prelazimo u stanje  $q_{MEM}$ . Ta tri prijelaza su identična onima opisanim kod naredbe LOAD, pa ih nećemo ponovno pisati. Nakon toga čitamo znak 1 i prelazimo u stanje  $q_{ST}$ . U tom stanju preskačemo jednu nulu:

$$\begin{aligned}\delta(q_{MEM}, \{P, 1, P, P, P\}) &= (q_{ST}, \{P, 1, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{ST}, \{P, 0, P, P, P\}) &= (q_{ST}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})\end{aligned}$$

Nakon toga čitamo oznaku registra i ovisno o tom bitu pomičemo glavu udesno na odgovarajućoj traci. Nakon toga znak # zamjenjujemo znakom S čime označavamo naredbu STORE, i mičemo glavu u lijevo, na znak P (kako bi minimizirali krajnji broj prijelaza):

$$\delta(q_{ST}, \{P, 0, P, P, P\}) = (q_{ST}, \{P, 0, P, P, P\}, \{S, S, R, S, S\})$$

$$\delta(q_{ST}, \{P, 0, \#, P, P\}) = (q_{POZI}, \{P, 0, S, P, P\}, \{R, R, L, S, R\})$$

$$\delta(q_{ST}, \{P, 1, P, P, P\}) = (q_{ST}, \{P, 1, P, P, P\}, \{S, S, S, R, S\})$$

$$\delta(q_{ST}, \{P, 1, P, \#, P\}) = (q_{POZI}, \{P, 1, P, S, P\}, \{R, R, S, L, R\})$$

## 2.4.2 Čitanje adrese i pozicioniranje glave

Algoritam za traženje adrese, kao i prijelazi, potpuno su identični onima u odjeljku 2.3.2.

## 2.4.3 Kopiranje podataka iz registra u memoriju

Način na koji iz stanja  $q_{ADR}$  prelazimo u stanje  $q_{ST}$ , tj. stanja  $q_{STR1}$  i  $q_{STR2}$  potpuno je analogan načinu opisanom u odjeljku 2.3.3, pa prijelaze zapravo samo kopiramo i modificiramo da odgovaraju naredbi STORE. Još moramo napisati prijelaze koji obavljaju funkciju kopiranja podatka odgovarajućeg registra u memoriju.

Kopiranje u memoriju iz registra R1:

$$\delta(q_{ADR\_END}, \{\#, \#, S, \#, P\}) = (q_{STR1}, \{\#, \#, \#, \#, P\}, \{R, S, R, L, S\})$$

$$\delta(q_{STR1}, \{0, \#, 0, P, P\}) = (q_{STR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{0, \#, 1, P, P\}) = (q_{STR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{1, \#, 0, P, P\}) = (q_{STR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{1, \#, 1, P, P\}) = (q_{STR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

Vraćanje glave registra R1 na početak (ostala 3 prijelaza su identična onima prije opisanim):

$$\delta(q_{STR1}, \{\#, \#, \#, P, P\}) = (q_{R1\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, L, S, S\})$$

Kopiranje u memoriju iz registra R2:

$$\delta(q_{ADR\_END}, \{\#, \#, \#, S, P\}) = (q_{STR2}, \{\#, \#, \#, \#, P\}, \{R, S, L, R, S\})$$

$$\delta(q_{STR2}, \{0, \#, P, 0, P\}) = (q_{STR2}, \{0, \#, P, 0, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{STR2}, \{0, \#, P, 1, P\}) = (q_{STR2}, \{0, \#, P, 0, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{STR2}, \{1, \#, P, 0, P\}) = (q_{STR2}, \{1, \#, P, 1, P\}, \{R, S, S, R, S\})$$

$$\delta(q_{STR2}, \{1, \#, P, 1, P\}) = (q_{STR2}, \{1, \#, P, 1, P\}, \{R, S, S, R, S\})$$

Vraćanje glave registra R2 na početak (ostala 3 prijelaza su identična onima prije opisanim):

$$\delta(q_{STR2}, \{\#, \#, \#, P, P\}) = (q_{R2\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, S, L, S\})$$

Također, prijelazi za vraćanje prve glave na početak su isti prije opisanim:

## 2.4.4 Završni radovi

Kao što je opisano u odjeljku 2.3.4, na kraju rada STORE naredbe, moramo za jedan povećati broj upisan na četvrtoj traci. Algoritam za to, kao i funkcije prijelaza, potpuno su identični onima iz navedenog odjeljka.

## 2.5 ADD

### 2.5.1 Prepoznavanje naredbe ADD

Nalazimo se u stanju  $q_N$ . Drugom glavom čitamo 1 i prelazimo u stanje  $q_{ARI}$ . Nakon toga opet čitamo znak 0 i prelazimo u stanje  $q_{ADD}$ . U tom stanju preskačemo jednu nulu.

$$\delta(q_N, \{P, 1, P, P, P\}) = (q_{ARI}, \{P, 1, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{ARI}, \{P, 0, P, P, P\}) = (q_{ADD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADD}, \{P, 0, P, P, P\}) = (q_{ADD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

Zatim čitamo bit koji označava broj registra i ovisno o njemu pomičemo glavu na traci tog registra za jedno mjesto u desno. Nakon toga znak # zamjenjujemo znakom A čime označavamo naredbu ADD, i mičemo glavu u lijevo, na znak P (kako bi minimizirali krajnji broj prijelaza):

$$\delta(q_{ADD}, \{P, 0, P, P, P\}) = (q_{ADD}, \{P, 0, P, P, P\}, \{S, S, R, S, S\})$$

$$\delta(q_{ADD}, \{P, 0, \#, P, P\}) = (q_{ADD\_SKIP4}, \{P, 0, A, P, P\}, \{S, R, L, S, S\})$$

$$\delta(q_{ADD}, \{P, 1, P, P, P\}) = (q_{ADD}, \{P, 1, P, P, P\}, \{S, S, S, R, S\})$$

$$\delta(q_{ADD}, \{P, 1, P, \#, P\}) = (q_{ADD\_SKIP4}, \{P, 1, P, A, P\}, \{S, R, S, L, S\})$$

Stanje  $q_{ADD\_SKIP4}$  služi za preskakanje ostala četiri bita naredbe koji se sastoje od tri znaka 0 i bita oznake drugog registra. Pošto naše računalo ima samo dva registra, možemo zaključiti da je oznaka drugog registra 1 ako je oznaka prvog 0, odnosno 0 ako je oznaka prvog 1.

$$\delta(q_{ADD\_SKIP4}, \{P, 0, P, P, P\}) = (q_{ADD\_SKIP4}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADD\_SKIP4}, \{P, 1, P, P, P\}) = (q_{ADD\_SKIP4}, \{P, 1, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{ADD\_SKIP4}, \{P, \#, P, P, P\}) = (q_{ADD\_R1R2}, \{P, \#, P, P, P\}, \{S, S, R, R, S\})$$

### 2.5.2 Zbrajanje

U prethodnom odjeljku smo na drugo mjesto registra koji je u ADD naredbi naveden na prvom mjestu zapisali A, prema tome znamo da u njega trebamo spremiti rezultat. Ovisno u kojem registru je A, prelazimo u pripadajuće stanje:  $q_{ADD\_R1}$  ili  $q_{ADD\_R2}$ . U stanju  $q_{ADD\_R1}$  rezultat spremamo u registar r1, a u stanju  $q_{ADD\_R2}$  rezultat spremamo u r2:

$$\delta(q_{ADD\_R1R2}, \{P, \#, A, \#, P\}) = (q_{ADD\_R1\_P}, \{P, \#, \#, \#, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{ADD\_R1R2}, \{P, \#, \#, A, P\}) = (q_{ADD\_R2\_P}, \{P, \#, \#, \#, P\}, \{S, S, R, R, S\})$$

Prema tome, moramo imati dva odvojena skupa stanja koja u biti rade identičan posao. To bi se moglo izbjeći korištenjem dodatne, radne trake, no tada bismo kod konstruiranja svakog prijelaza TS-a morali navesti još 3 nove brojke, pa smatram da to nije potrebno radi bolje preglednosti samih funkcija prijelaza. Postupak zbrajanja je sljedeći: u oba stanja se prvo moramo pozicionirati na krajnje desni bit oba registra. Nakon toga imamo 4 slučaja:

- ako su oba bita 0, mičemo obje glave ulijevo
- ako je jedan od bitova 1, upisujemo 1 u odgovarajući registar, i mičemo obje glave u lijevo
- ako su oba bita 1, upisujemo 0 u odgovarajući registar, mičemo obje glave u lijevo i prelazimo u stanje u kojem na rezultat zbrajanja dodajemo 1

Pozicioniranje na krajnje desni bit, za registar r1:

$$\begin{aligned}\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 0, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 1, P\}) &= (q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 1, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 0, P\}) &= (q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 0, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 1, P\}) &= (q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 1, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R1\_P}}, \{P, \#, \#, \#, P\}) &= (q_{\text{ADD\_R1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, S\})\end{aligned}$$

Glavni dio operacije zbrajanja (za r1):

$$\begin{aligned}\delta(q_{\text{ADD\_R1}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R1}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R1}}, \{P, \#, 0, 1, P\}) &= (q_{\text{ADD\_R1}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}) &= (q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R1}}, \{P, \#, 1, 1, P\}) &= (q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R1}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\}) \\ \delta(q_{\text{CARRY\_R1}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}) &= (q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R1}}, \{P, \#, 1, 0, P\}) &= (q_{\text{CARRY\_R1}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R1}}, \{P, \#, 1, 1, P\}) &= (q_{\text{CARRY\_R1}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R1}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\})\end{aligned}$$

Pozicioniranje na krajnje desni bit, za registar r2:

$$\begin{aligned}\delta(q_{\text{ADD\_R2\_P}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R2\_P}}, \{P, \#, 0, 0, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R2\_P}}, \{P, \#, 0, 1, P\}) &= (q_{\text{ADD\_R2\_P}}, \{P, \#, 0, 1, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R2\_P}}, \{P, \#, 1, 0, P\}) &= (q_{\text{ADD\_R2\_P}}, \{P, \#, 1, 0, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R2\_P}}, \{P, \#, 1, 1, P\}) &= (q_{\text{ADD\_R2\_P}}, \{P, \#, 1, 1, P\}, \{S, S, R, R, S\}) \\ \delta(q_{\text{ADD\_R2\_P}}, \{P, \#, \#, \#, P\}) &= (q_{\text{ADD\_R2}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, S\})\end{aligned}$$

Glavni dio operacije zbrajanja (za r2):

$$\begin{aligned}\delta(q_{\text{ADD\_R2}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R2}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R2}}, \{P, \#, 0, 1, P\}) &= (q_{\text{ADD\_R2}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R2}}, \{P, \#, 1, 0, P\}) &= (q_{\text{ADD\_R2}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R2}}, \{P, \#, 1, 1, P\}) &= (q_{\text{CARRY\_R2}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{ADD\_R2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\}) \\ \delta(q_{\text{CARRY\_R2}}, \{P, \#, 0, 0, P\}) &= (q_{\text{ADD\_R2}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R2}}, \{P, \#, 0, 1, P\}) &= (q_{\text{CARRY\_R2}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R2}}, \{P, \#, 1, 0, P\}) &= (q_{\text{CARRY\_R2}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R2}}, \{P, \#, 1, 1, P\}) &= (q_{\text{CARRY\_R2}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{CARRY\_R2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\})\end{aligned}$$

## 2.5.3 Završni radovi

U posljednjem prijelazu prethodnog dijela prebacujemo se u stanje  $q_{\text{PC\_POZ1}}$  koje je opisano u dijelu 2.3.4.

## 2.6 SUB

### 2.6.1 Prepoznavanje naredbe SUB

Nalazimo se u stanju  $q_{ARI}$ . Drugom glavom prve trake pročitamo znak 0 i prelazimo u stanje  $q_{SUB}$ . U tom stanju preskačemo jednu nulu.

$$\delta(q_{ARI}, \{P, 0, P, P, P\}) = (q_{SUB}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{SUB}, \{P, 0, P, P, P\}) = (q_{SUB}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

Zatim čitamo bit koji označava broj registra i ovisno o njemu pomičemo glavu na traci tog registra za jedno mjesto u desno. Nakon toga znak # zamjenjujemo znakom - (već imamo oznaku S za STORE) čime označavamo naredbu SUB, i mičemo glavu u lijevo, na znak P (kako bi minimizirali krajnji broj prijelaza):

$$\delta(q_{SUB}, \{P, 0, P, P, P\}) = (q_{SUB}, \{P, 0, P, P, P\}, \{S, S, R, S, S\})$$

$$\delta(q_{SUB}, \{P, 0, \#, P, P\}) = (q_{SUB\_SKIP4}, \{P, 0, -, P, P\}, \{S, R, L, S, S\})$$

$$\delta(q_{SUB}, \{P, 1, P, P, P\}) = (q_{SUB}, \{P, 1, P, P, P\}, \{S, S, S, R, S\})$$

$$\delta(q_{SUB}, \{P, 1, P, \#, P\}) = (q_{SUB\_SKIP4}, \{P, 1, P, -, P\}, \{S, R, S, L, S\})$$

Stanje  $q_{SUB\_SKIP4}$  služi za preskakanje ostala četiri bita naredbe koji se sastoje od tri znaka 0 i bita oznake drugog registra. Pošto naše računalo ima samo dva registra, možemo zaključiti da je oznaka drugog registra 1 ako je oznaka prvog 0, odnosno 0 ako je oznaka prvog 1.

$$\delta(q_{SUB\_SKIP4}, \{P, 0, P, P, P\}) = (q_{SUB\_SKIP4}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{SUB\_SKIP4}, \{P, 1, P, P, P\}) = (q_{SUB\_SKIP4}, \{P, 1, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{SUB\_SKIP4}, \{P, \#, P, P, P\}) = (q_{SUB\_R1R2}, \{P, \#, P, P, P\}, \{S, S, R, R, S\})$$

### 2.6.2 Oduzimanje

Oduzimanje radimo na način da zbrojimo prvi broj i drugi komplement drugog broja. Za to je potrebno drugi broj prebaciti u format dvojnog komplementa i zatim prijeći u stanje  $q_{ADD}$  koje već ima konstruirane prijelaze za zbrajanje dva broja. Na kraju je potrebno drugi broj vratiti u prvotno stanje.

U prethodnom odjeljku smo na drugo mjesto registra koji je u SUB naredbi naveden na prvom mjestu zapisali -, prema tome znamo da u njega trebamo spremati rezultat. Ovisno u kojem registru je -, prelazimo u pripadajuće stanje:  $q_{SUB\_R1}$  ili  $q_{SUB\_R2}$ . U stanju  $q_{SUB\_R1}$  rezultat spremamo u registar  $r1$ , a u stanju  $q_{SUB\_R2}$  rezultat spremamo u  $r2$ :

$$\delta(q_{SUB\_R1R2}, \{P, \#, -, \#, P\}) = (q_{SUB\_R1\_P}, \{P, \#, \#, \#, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{SUB\_R1R2}, \{P, \#, \#, -, P\}) = (q_{SUB\_R2\_P}, \{P, \#, \#, \#, P\}, \{S, S, R, R, S\})$$

Prema tome, moramo imati dva odvojena skupa stanja koja u biti rade identičan posao. To bi se moglo izbjeći korištenjem dodatne, radne trake, no tada bismo kod konstruiranja svakog prijelaza TS-a morali navesti još 3 nove brojke, pa smatram da to nije potrebno radi bolje preglednosti samih funkcija prijelaza.

Prvo pozicioniramo obje glave na zadnji znak traka:

$$\begin{aligned}
\delta(q_{\text{SUB\_R1\_P}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R1\_P}}, \{P, \#, 0, 0, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R1\_P}}, \{P, \#, 0, 1, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R1\_P}}, \{P, \#, 1, 0, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R1\_P}}, \{P, \#, 1, 1, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_K1}}, \{P, \#, \#, \#, P\}, \{S, S, S, L, S\})
\end{aligned}$$

$$\begin{aligned}
\delta(q_{\text{SUB\_R2\_P}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R2\_P}}, \{P, \#, 0, 0, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R2\_P}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R2\_P}}, \{P, \#, 0, 1, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R2\_P}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R2\_P}}, \{P, \#, 1, 0, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R2\_P}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R2\_P}}, \{P, \#, 1, 1, P\}, \{S, S, R, R, S\}) \\
\delta(q_{\text{SUB\_R2\_P}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_K1}}, \{P, \#, \#, \#, P\}, \{S, S, L, S, S\})
\end{aligned}$$

Sljedeća stvar koju moramo napraviti je dvojno komplementiranje broja koji se oduzima. Postupak za to je sljedeći: krećemo se ulijevo i prepisujemo bitove. Kada naiđemo na znak 1, njega također prepisemo i prebacujemo se u stanje  $q_{\text{SUB\_R1\_K2}}$ :

$$\begin{aligned}
\delta(q_{\text{SUB\_R1\_K1}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_K1}}, \{P, \#, \#, 0, P\}, \{S, S, S, L, S\}) \\
\delta(q_{\text{SUB\_R1\_K1}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, 1, P\}, \{S, S, S, L, S\}) \\
\delta(q_{\text{SUB\_R2\_K1}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_K1}}, \{P, \#, 0, \#, P\}, \{S, S, L, S, S\}) \\
\delta(q_{\text{SUB\_R2\_K1}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_K2}}, \{P, \#, 1, \#, P\}, \{S, S, L, S, S\})
\end{aligned}$$

Ono komplementira sve sljedeće bitove na koje naiđe. Kada dođemo do početka registra, prebacujemo se u stanje  $q_{\text{SUB\_R1\_P2}}$ .

$$\begin{aligned}
\delta(q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, 1, P\}, \{S, S, S, L, S\}) \\
\delta(q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, 0, P\}, \{S, S, S, L, S\}) \\
\delta(q_{\text{SUB\_R1\_K2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, \#, P\}, \{S, S, S, R, S\}) \\
\delta(q_{\text{SUB\_R2\_K2}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_K2}}, \{P, \#, 1, \#, P\}, \{S, S, L, S, S\}) \\
\delta(q_{\text{SUB\_R2\_K2}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_K2}}, \{P, \#, 0, \#, P\}, \{S, S, L, S, S\}) \\
\delta(q_{\text{SUB\_R2\_K2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_P2}}, \{P, \#, \#, \#, P\}, \{S, S, R, S, S\})
\end{aligned}$$

Ono vraća glavu na kraj registra:

$$\begin{aligned}
\delta(q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, 0, P\}, \{S, S, S, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, 1, P\}, \{S, S, S, R, S\}) \\
\delta(q_{\text{SUB\_R1\_P2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_ADD}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, S\}) \\
\delta(q_{\text{SUB\_R2\_P2}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_P2}}, \{P, \#, 0, \#, P\}, \{S, S, R, S, S\}) \\
\delta(q_{\text{SUB\_R2\_P2}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_P2}}, \{P, \#, 1, \#, P\}, \{S, S, R, S, S\}) \\
\delta(q_{\text{SUB\_R2\_P2}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_ADD}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, S\})
\end{aligned}$$

Zatim krećemo sa zbrajanjem ta dva broja koje je potpuno analogno već opisanom postupku:

$$\begin{aligned}
\delta(q_{\text{SUB\_R1\_ADD}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R1\_ADD}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\
\delta(q_{\text{SUB\_R1\_ADD}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R1\_ADD}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\
\delta(q_{\text{SUB\_R1\_ADD}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R1\_ADD}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\
\delta(q_{\text{SUB\_R1\_ADD}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\
\delta(q_{\text{SUB\_R1\_ADD}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, \#, P\}, \{S, S, L, R, R\}) \\
\delta(q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R1\_ADD}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\})
\end{aligned}$$



$$\begin{aligned}\delta(q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R1\_CARRY}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R1\_CARRY}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, \#, P\}, \{S, S, S, R, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{SUB\_R2\_ADD}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R2\_ADD}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_ADD}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R2\_ADD}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_ADD}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R2\_ADD}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_ADD}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_ADD}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_P3}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\}) \\ \delta(q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 0, 0, P\}) &= (q_{\text{SUB\_R2\_ADD}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 0, 1, P\}) &= (q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 1, 0, P\}) &= (q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 1, 1, P\}) &= (q_{\text{SUB\_R2\_CARRY}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\}) \\ \delta(q_{\text{SUB\_R2\_CARRY}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_P3}}, \{P, \#, \#, \#, P\}, \{S, S, S, L, S\})\end{aligned}$$

Na kraju ne smijemo zaboraviti ponovno dvojno komplementirati broj kako bi dobili broj koji je bio originalno zapisan u registru, jer ni jedna operacija TS-a ne smije biti destruktivna. Prvo se odgovarajućom glavom pozicioniramo na zadnji bit registra:

$$\begin{aligned}\delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, 0, P\}, \{S, S, S, R, S\}) \\ \delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, 1, P\}, \{S, S, S, R, S\}) \\ \delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R1\_K3}}, \{P, \#, \#, \#, P\}, \{S, S, S, L, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_P3}}, \{P, \#, 0, \#, P\}, \{S, S, R, S, S\}) \\ \delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_P3}}, \{P, \#, 1, \#, P\}, \{S, S, R, S, S\}) \\ \delta(q_{\text{SUB\_R1\_P3}}, \{P, \#, \#, \#, P\}) &= (q_{\text{SUB\_R2\_K3}}, \{P, \#, \#, \#, P\}, \{S, S, L, S, S\})\end{aligned}$$

Na kraju dvojno komplementiramo broj kako bi dobili početni broj. Postupak za to je opisan nešto ranije:

$$\begin{aligned}\delta(q_{\text{SUB\_R1\_K3}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_K3}}, \{P, \#, \#, 0, P\}, \{S, S, S, L, S\}) \\ \delta(q_{\text{SUB\_R1\_K3}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, 1, P\}, \{S, S, S, L, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{SUB\_R2\_K3}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_K3}}, \{P, \#, 0, \#, P\}, \{S, S, L, S, S\}) \\ \delta(q_{\text{SUB\_R2\_K3}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_K4}}, \{P, \#, 1, \#, P\}, \{S, S, L, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, 0, P\}) &= (q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, 1, P\}, \{S, S, S, L, S\}) \\ \delta(q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, 1, P\}) &= (q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, 0, P\}, \{S, S, S, L, S\}) \\ \delta(q_{\text{SUB\_R1\_K4}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{SUB\_R2\_K4}}, \{P, \#, 0, \#, P\}) &= (q_{\text{SUB\_R2\_K4}}, \{P, \#, 1, \#, P\}, \{S, S, L, S, S\}) \\ \delta(q_{\text{SUB\_R2\_K4}}, \{P, \#, 1, \#, P\}) &= (q_{\text{SUB\_R2\_K4}}, \{P, \#, 0, \#, P\}, \{S, S, L, S, S\}) \\ \delta(q_{\text{SUB\_R2\_K4}}, \{P, \#, \#, \#, P\}) &= (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\})\end{aligned}$$

### 2.6.3 Završni radovi

U posljednjem prijelazu prethodnog dijela prebacujemo se u stanje  $q_{\text{PC\_POZ1}}$  koje je opisano u dijelu 2.3.4.



### 3. Izvođenje zadanog programa

Odmah par napomena:

Pošto nije definirano u zadatku, a bilo bi beskorisno zbrajati dvije nule, na adresi 11 zapisano je 34 (binarno 00100010), a na adresi 12 zapisano je 124 (binarno 01111100). Rezultat zbrajanja bi na kraju trebao ispasti 158 (binarno 10011110).

Također, zadnja naredba glasi: STORE r2, 12. Pretpostavljam da je autor zadatka mislio na memorijsku lokaciju 12 spremi rezultat zbrajanja, a on je u r1. Tako da ću promijeniti tu naredbu u STORE r1, 12.

Izgled traka na početku je opisan u odjeljku 2.1. Potrebno je samo definirati točan izgled instrukcijskog dijela prva trake. Način kodiranja naredbi opisan je u odjeljku 2:



P	#	0	0	0	0	1	0	1	1	#	0	0	0	1	1	1	0	0	#	1	0	0	0	0	0	1	#	0	1	0	0	1	1	0	0	#	\$	..



Ostatak trake za radnu memoriju je ispunjen nulama, osim memorijskih lokacija 11 i 12 na koje su upisani brojevi 34 i 124.

Početni izgled ostale 3 trake je identičan kao u odjeljku 2.1.

#### **LOAD r1, 11:**

Dekodiranje vrste naredbe i registra:

$$\begin{aligned}\delta(q_P, \{P, \#, P, P, P\}) &= (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_N, \{P, 0, P, P, P\}) &= (q_{MEM}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{MEM}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{LD}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{LD}, \{P, 0, P, P, P\}) &= (q_{LD}, \{P, 0, P, P, P\}, \{S, S, R, S, S\}) \\ \delta(q_{LD}, \{P, 0, \#, P, P\}) &= (q_{POZI}, \{P, 0, L, P, P\}, \{R, R, L, S, R\})\end{aligned}$$

Traženje adrese 1011:

Čitanje prvog bita adrese, i pošto je on 1, to znači da moramo preskočiti 8 blokova:

$$\begin{aligned}\delta(q_{POZI}, \{\#, 1, P, P, \#\}) &= (q_{ADR_8}, \{\#, 1, P, P, 1\}, \{R, S, S, S, L\}) \\ \delta(q_{ADR_8}, \{0, 1, P, P, P\}) &= (q_{ADR_8}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_8}, \{1, 1, P, P, P\}) &= (q_{ADR_8}, \{1, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_8}, \{\#, 1, P, P, P\}) &= (q_{ADR_7}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{ADR_7}, \{0, 1, P, P, P\}) &= (q_{ADR_7}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_7}, \{1, 1, P, P, P\}) &= (q_{ADR_7}, \{1, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_7}, \{\#, 1, P, P, P\}) &= (q_{ADR_6}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{ADR_6}, \{0, 1, P, P, P\}) &= (q_{ADR_6}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_6}, \{1, 1, P, P, P\}) &= (q_{ADR_6}, \{1, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{ADR_6}, \{\#, 1, P, P, P\}) &= (q_{ADR_5}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR5}}, \{0, 1, P, P, P\}) &= (q_{\text{ADR5}}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR5}}, \{1, 1, P, P, P\}) &= (q_{\text{ADR5}}, \{1, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR5}}, \{\#, 1, P, P, P\}) &= (q_{\text{ADR4}}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR4}}, \{0, 1, P, P, P\}) &= (q_{\text{ADR4}}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR4}}, \{S, 1, P, P, P\}) &= (q_{\text{ADR4}}, \{S, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR4}}, \{\#, 1, P, P, P\}) &= (q_{\text{ADR3}}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR3}}, \{0, 1, P, P, P\}) &= (q_{\text{ADR3}}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR3}}, \{\#, 1, P, P, P\}) &= (q_{\text{ADR2}}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR2}}, \{0, 1, P, P, P\}) &= (q_{\text{ADR2}}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR2}}, \{\#, 1, P, P, P\}) &= (q_{\text{ADR1}}, \{\#, 1, P, P, P\}, \{R, S, S, S, R\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR1}}, \{0, 1, P, P, 1\}) &= (q_{\text{ADR1}}, \{0, 1, P, P, 1\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR1}}, \{\#, 1, P, P, 1\}) &= (q_{\text{POZ2}}, \{\#, 1, P, P, 1\}, \{S, R, S, S, S\})\end{aligned}$$

Pošto je drugi bit adrese 0, prelazimo na treći bit koji je 1 i prema tome moramo preskočiti 2 bloka:

$$\delta(q_{\text{POZ2}}, \{\#, 0, P, P, 1\}) = (q_{\text{POZ3}}, \{\#, 0, P, P, 2\}, \{S, R, S, S, S\})$$

$$\begin{aligned}\delta(q_{\text{POZ3}}, \{\#, 1, P, P, 2\}) &= (q_{\text{ADR}_2}, \{\#, 1, P, P, 3\}, \{R, S, S, S, L\}) \\ \delta(q_{\text{ADR2}}, \{0, 1, P, P, P\}) &= (q_{\text{ADR2}}, \{0, 1, P, P, P\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR2}}, \{\#, 1, P, P, P\}) &= (q_{\text{ADR1}}, \{\#, 1, P, P, P\}, \{R, S, S, S, R\})\end{aligned}$$

$$\begin{aligned}\delta(q_{\text{ADR1}}, \{0, 1, P, P, 3\}) &= (q_{\text{ADR1}}, \{0, 1, P, P, 3\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR1}}, \{\#, 1, P, P, 3\}) &= (q_{\text{POZ4}}, \{\#, 1, P, P, 3\}, \{S, R, S, S, S\})\end{aligned}$$

Prelazimo na obradu četvrtog bita adrese koji je 1, i prema tome preskačemo samo jedan blok:

$$\begin{aligned}\delta(q_{\text{POZ4}}, \{\#, 1, P, P, 3\}) &= (q_{\text{ADR}_1}, \{\#, 1, P, P, 4\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR1}}, \{0, 1, P, P, 4\}) &= (q_{\text{ADR1}}, \{0, 1, P, P, 4\}, \{R, S, S, S, S\}) \\ \delta(q_{\text{ADR1}}, \{\#, 1, P, P, 4\}) &= (q_{\text{ADR\_END}}, \{\#, 1, P, P, \#\}, \{S, R, S, S, S\})\end{aligned}$$

Pomičemo glave oba registra kako bi pročitali oznaku naredbe:

$$\delta(q_{\text{ADR\_END}}, \{\#, \#, P, P, P\}) = (q_{\text{ADR\_END}}, \{\#, \#, P, P, P\}, \{S, S, R, R, S\})$$

Prelazak u stanje LDR1 pomoću oznake L na traci r1:

$$\delta(q_{\text{ADR\_END}}, \{\#, \#, L, \#, P\}) = (q_{\text{LDR1}}, \{\#, \#, \#, \#, P\}, \{R, S, R, L, S\})$$

Kopiranje u r1:

$$\begin{aligned}\delta(q_{\text{LDR1}}, \{0, \#, 0, P, P\}) &= (q_{\text{LDR1}}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\}) \\ \delta(q_{\text{LDR1}}, \{0, \#, 1, P, P\}) &= (q_{\text{LDR1}}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\}) \\ \delta(q_{\text{LDR1}}, \{1, \#, 0, P, P\}) &= (q_{\text{LDR1}}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\}) \\ \delta(q_{\text{LDR1}}, \{1, \#, 1, P, P\}) &= (q_{\text{LDR1}}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})\end{aligned}$$

Vraćanje glave r1:

$$\begin{aligned}\delta(q_{\text{LDR1}}, \{\#, \#, \#, P, P\}) &= (q_{\text{R1\_BACK}}, \{\#, \#, \#, P, P\}, \{S, S, L, S, S\}) \\ \delta(q_{\text{R1\_BACK}}, \{\#, \#, 0, P, P\}) &= (q_{\text{R1\_BACK}}, \{\#, \#, 0, P, P\}, \{S, S, L, S, S\})\end{aligned}$$

$$\delta(q_{R1\_BACK}, \{\#, \#, 1, P, P\}) = (q_{R1\_BACK}, \{\#, \#, 1, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, \#, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, \#, P, P\}, \{L, S, L, S, S\})$$

Vraćanje prve glave:

$$\delta(q_{ADR\_BACK}, \{0, \#, P, P, P\}) = (q_{ADR\_BACK}, \{0, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{1, \#, P, P, P\}) = (q_{ADR\_BACK}, \{1, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{\#, \#, P, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{S, \#, P, P, P\}) = (q_{ADR\_BACK}, \{S, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{P, \#, P, P, P\}) = (q_{PC\_POZ1}, \{P, \#, P, P, P\}, \{S, S, S, S, R\})$$

Pozicioniranje na zadnji bit PC-a:

$$\delta(q_{PC\_POZ1}, \{P, \#, P, P, \#\}) = (q_{PC\_POZ}, \{P, \#, P, P, \#\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_POZ}, \{P, \#, P, P, 0\}) = (q_{PC\_POZ}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_POZ}, \{P, \#, P, P, \#\}) = (q_{PC\_1}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

Povećanje za 1:

$$\delta(q_{PC\_1}, \{P, \#, P, P, 0\}) = (q_{PC\_BACK}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

Vraćanje na početak PC-a i prelazak na obradu sljedeće naredbe:

$$\delta(q_{PC\_BACK}, \{P, \#, P, P, 0\}) = (q_{PC\_BACK}, \{P, \#, P, P, 0\}, \{S, S, S, S, L\})$$

$$\delta(q_{PC\_BACK}, \{P, \#, P, P, \#\}) = (q_P, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

## LOAD r2, 12

Dekodiranje vrste naredbe i registra:

$$\delta(q_P, \{P, \#, P, P, P\}) = (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_N, \{P, 0, P, P, P\}) = (q_{MEM}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{MEM}, \{P, 0, P, P, P\}) = (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{LD}, \{P, 0, P, P, P\}) = (q_{LD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{LD}, \{P, 1, P, P, P\}) = (q_{LD}, \{P, 1, P, P, P\}, \{S, S, S, R, S\})$$

$$\delta(q_{LD}, \{P, 1, P, \#, P\}) = (q_{POZ1}, \{P, 1, P, L, P\}, \{R, R, S, L, R\})$$

Traženje adrese 1100:

Čitanje prvog bita adrese, i pošto je on 1, to znači da moramo preskočiti 8 blokova:

$$\delta(q_{POZ1}, \{\#, 1, P, P, \#\}) = (q_{ADR\_8}, \{\#, 1, P, P, 1\}, \{R, S, S, S, L\})$$

$$\delta(q_{ADR\_8}, \{0, 1, P, P, P\}) = (q_{ADR\_8}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_8}, \{1, 1, P, P, P\}) = (q_{ADR\_8}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_8}, \{\#, 1, P, P, P\}) = (q_{ADR\_7}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_7}, \{0, 1, P, P, P\}) = (q_{ADR\_7}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_7}, \{1, 1, P, P, P\}) = (q_{ADR\_7}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_7}, \{\#, 1, P, P, P\}) = (q_{ADR\_6}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_6}, \{0, 1, P, P, P\}) = (q_{ADR\_6}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR\_6}, \{1, 1, P, P, P\}) = (q_{ADR\_6}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{\text{ADR6}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR5}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR5}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR5}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR5}}, \{1, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR5}}, \{1, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR5}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR4}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR4}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR4}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR4}}, \{\$, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR4}}, \{\$, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR4}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR3}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR3}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR3}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR3}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR2}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR2}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR2}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR2}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR1}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{R}\})$$

$$\delta(q_{\text{ADR1}}, \{0, 1, \text{P}, \text{P}, 1\}) = (q_{\text{ADR1}}, \{0, 1, \text{P}, \text{P}, 1\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR1}}, \{\#, 1, \text{P}, \text{P}, 1\}) = (q_{\text{POZ2}}, \{\#, 1, \text{P}, \text{P}, 1\}, \{\text{S}, \text{R}, \text{S}, \text{S}, \text{S}\})$$

Prelazimo na obradu drugog bita adrese, a pošto je on 1, preskačemo 4 bita:

$$\delta(q_{\text{POZ2}}, \{\#, 1, \text{P}, \text{P}, 1\}) = (q_{\text{ADR}_4}, \{\#, 1, \text{P}, \text{P}, 2\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{L}\})$$

$$\delta(q_{\text{ADR4}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR4}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR4}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR3}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR3}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR3}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR3}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR2}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR2}}, \{0, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR2}}, \{0, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR2}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR1}}, \{\#, 1, \text{P}, \text{P}, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{R}\})$$

$$\delta(q_{\text{ADR1}}, \{0, 1, \text{P}, \text{P}, 1\}) = (q_{\text{ADR1}}, \{0, 1, \text{P}, \text{P}, 1\}, \{\text{R}, \text{S}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{ADR1}}, \{\#, 1, \text{P}, \text{P}, 2\}) = (q_{\text{POZ3}}, \{\#, 1, \text{P}, \text{P}, 2\}, \{\text{S}, \text{R}, \text{S}, \text{S}, \text{S}\})$$

Prelazimo na obradu ostalih bitova, no pošto su oba 0, preskačemo ih:

$$\delta(q_{\text{POZ3}}, \{\#, 0, \text{P}, \text{P}, 2\}) = (q_{\text{POZ4}}, \{\#, 0, \text{P}, \text{P}, 3\}, \{\text{S}, \text{R}, \text{S}, \text{S}, \text{S}\})$$

$$\delta(q_{\text{POZ4}}, \{\#, 0, \text{P}, \text{P}, 3\}) = (q_{\text{ADR\_END}}, \{\#, 0, \text{P}, \text{P}, \#\}, \{\text{S}, \text{R}, \text{S}, \text{S}, \text{S}\})$$

Pomičemo glave oba registra kako bi pročitali oznaku naredbe:

$$\delta(q_{\text{ADR\_END}}, \{\#, \#, \text{P}, \text{P}, \text{P}\}) = (q_{\text{ADR\_END}}, \{\#, \#, \text{P}, \text{P}, \text{P}\}, \{\text{S}, \text{S}, \text{R}, \text{R}, \text{S}\})$$

Prelazak u stanje LDR2 pomoću oznake L na traci tog registra:

$$\delta(q_{\text{ADR\_END}}, \{\#, \#, \#, \text{L}, \text{P}\}) = (q_{\text{LDR2}}, \{\#, \#, \#, \#, \text{P}\}, \{\text{R}, \text{S}, \text{L}, \text{R}, \text{S}\})$$

Kopiranje u r2:

$$\delta(q_{\text{LDR2}}, \{0, \#, \text{P}, 0, \text{P}\}) = (q_{\text{LDR2}}, \{0, \#, \text{P}, 0, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{R}, \text{S}\})$$

$$\delta(q_{\text{LDR2}}, \{1, \#, \text{P}, 0, \text{P}\}) = (q_{\text{LDR2}}, \{1, \#, \text{P}, 1, \text{P}\}, \{\text{R}, \text{S}, \text{S}, \text{R}, \text{S}\})$$

Vraćanje glave r2:

$$\begin{aligned}\delta(q_{LDR2}, \{\#, \#, \#, P, P\}) &= (q_{R2\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, S, L, S\}) \\ \delta(q_{R2\_BACK}, \{\#, \#, P, 0, P\}) &= (q_{R2\_BACK}, \{\#, \#, P, 0, P\}, \{S, S, S, L, S\}) \\ \delta(q_{R2\_BACK}, \{\#, \#, P, 1, P\}) &= (q_{R2\_BACK}, \{\#, \#, P, 1, P\}, \{S, S, S, L, S\}) \\ \delta(q_{R2\_BACK}, \{\#, \#, P, \#, P\}) &= (q_{ADR\_BACK}, \{\#, \#, P, \#, P\}, \{L, S, S, L, S\})\end{aligned}$$

Vraćanje prve glave na početak:

$$\begin{aligned}\delta(q_{ADR\_BACK}, \{0, \#, P, P, P\}) &= (q_{ADR\_BACK}, \{0, \#, P, P, P\}, \{L, S, S, S, S\}) \\ \delta(q_{ADR\_BACK}, \{1, \#, P, P, P\}) &= (q_{ADR\_BACK}, \{1, \#, P, P, P\}, \{L, S, S, S, S\}) \\ \delta(q_{ADR\_BACK}, \{\#, \#, P, P, P\}) &= (q_{ADR\_BACK}, \{\#, \#, P, P, P\}, \{L, S, S, S, S\}) \\ \delta(q_{ADR\_BACK}, \{S, \#, P, P, P\}) &= (q_{ADR\_BACK}, \{S, \#, P, P, P\}, \{L, S, S, S, S\}) \\ \delta(q_{ADR\_BACK}, \{P, \#, P, P, P\}) &= (q_{PC\_POZ1}, \{P, \#, P, P, P\}, \{S, S, S, S, R\})\end{aligned}$$

Pozicioniranje na zadnji bit PC-a:

$$\begin{aligned}\delta(q_{PC\_POZ1}, \{P, \#, P, P, \#\}) &= (q_{PC\_POZ}, \{P, \#, P, P, \#\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, 0\}) &= (q_{PC\_POZ}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, 1\}) &= (q_{PC\_POZ}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_POZ}, \{P, \#, P, P, \#\}) &= (q_{PC\_1}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

Povećanje za 1:

$$\begin{aligned}\delta(q_{PC\_1}, \{P, \#, P, P, 1\}) &= (q_{PC\_2}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_2}, \{P, \#, P, P, 0\}) &= (q_{PC\_2\_END}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_2\_END}, \{P, \#, P, P, 1\}) &= (q_{PC\_2\_END}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\}) \\ \delta(q_{PC\_2\_END}, \{P, \#, P, P, \#\}) &= (q_{PC\_BACK}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

Vraćanje glave PC-a na početak i prelazak na sljedeću naredbu:

$$\begin{aligned}\delta(q_{PC\_BACK}, \{P, \#, P, P, 0\}) &= (q_{PC\_BACK}, \{P, \#, P, P, 0\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_BACK}, \{P, \#, P, P, 1\}) &= (q_{PC\_BACK}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\}) \\ \delta(q_{PC\_BACK}, \{P, \#, P, P, \#\}) &= (q_P, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})\end{aligned}$$

## ADD r1, r2

Dekodiranje vrste naredbe i oznake prvog registra:

$$\begin{aligned}\delta(q_P, \{P, \#, P, P, P\}) &= (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_N, \{P, 1, P, P, P\}) &= (q_{ARI}, \{P, 1, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{ARI}, \{P, 0, P, P, P\}) &= (q_{ADD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{ADD}, \{P, 0, P, P, P\}) &= (q_{ADD}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})\end{aligned}$$

$$\begin{aligned}\delta(q_{ADD}, \{P, 0, P, P, P\}) &= (q_{ADD}, \{P, 0, P, P, P\}, \{S, S, R, S, S\}) \\ \delta(q_{ADD}, \{P, 0, \#, P, P\}) &= (q_{ADD\_SKIP4}, \{P, 0, A, P, P\}, \{S, R, L, S, S\})\end{aligned}$$

Preskakanje ostala 4 bita naredbe, pošto znamo koji je drugi registar:

$$\begin{aligned}\delta(q_{ADD\_SKIP4}, \{P, 0, P, P, P\}) &= (q_{ADD\_SKIP4}, \{P, 0, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{ADD\_SKIP4}, \{P, 1, P, P, P\}) &= (q_{ADD\_SKIP4}, \{P, 1, P, P, P\}, \{S, R, S, S, S\}) \\ \delta(q_{ADD\_SKIP4}, \{P, \#, P, P, P\}) &= (q_{ADD\_R1R2}, \{P, \#, P, P, P\}, \{S, S, R, R, S\})\end{aligned}$$

Prelazimo u stanje R1\_P u kojem znamo da rezultat zbrajanja moramo spremiti u r1:

$$\delta(q_{\text{ADD\_R1R2}}, \{P, \#, A, \#, P\}) = (q_{\text{ADD\_R1\_P}}, \{P, \#, \#, \#, P\}, \{S, S, R, R, S\})$$

Pozicioniranje glava oba registra na kraj:

$$\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 0, P\}) = (q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 0, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 1, P\}) = (q_{\text{ADD\_R1\_P}}, \{P, \#, 0, 1, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 0, P\}) = (q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 0, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 1, P\}) = (q_{\text{ADD\_R1\_P}}, \{P, \#, 1, 1, P\}, \{S, S, R, R, S\})$$

$$\delta(q_{\text{ADD\_R1\_P}}, \{P, \#, \#, \#, P\}) = (q_{\text{ADD\_R1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, S\})$$

Zbrajanje brojeva 00100010 i 01111100 (34 i 124):

$$\delta(q_{\text{ADD\_R1}}, \{P, \#, 0, 0, P\}) = (q_{\text{ADD\_R1}}, \{P, \#, 0, 0, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}) = (q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{ADD\_R1}}, \{P, \#, 0, 1, P\}) = (q_{\text{ADD\_R1}}, \{P, \#, 1, 1, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{ADD\_R1}}, \{P, \#, 1, 1, P\}) = (q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}) = (q_{\text{CARRY\_R1}}, \{P, \#, 0, 1, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{CARRY\_R1}}, \{P, \#, 0, 0, P\}) = (q_{\text{ADD\_R1}}, \{P, \#, 1, 0, P\}, \{S, S, L, L, S\})$$

$$\delta(q_{\text{ADD\_R1}}, \{P, \#, \#, \#, P\}) = (q_{\text{PC\_POZ1}}, \{P, \#, \#, \#, P\}, \{S, S, L, L, R\})$$

Pozicioniranje na zadnji bit PC-a:

$$\delta(q_{\text{PC\_POZ1}}, \{P, \#, P, P, \#\}) = (q_{\text{PC\_POZ}}, \{P, \#, P, P, \#\}, \{S, S, S, S, R\})$$

$$\delta(q_{\text{PC\_POZ}}, \{P, \#, P, P, 0\}) = (q_{\text{PC\_POZ}}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\})$$

$$\delta(q_{\text{PC\_POZ}}, \{P, \#, P, P, 1\}) = (q_{\text{PC\_POZ}}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\})$$

$$\delta(q_{\text{PC\_POZ}}, \{P, \#, P, P, \#\}) = (q_{\text{PC\_1}}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

Povećanje za 1:

$$\delta(q_{\text{PC\_1}}, \{P, \#, P, P, 0\}) = (q_{\text{PC\_BACK}}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

Vraćanje glave PC-a na početak i prelazak na sljedeću naredbu:

$$\delta(q_{\text{PC\_BACK}}, \{P, \#, P, P, 0\}) = (q_{\text{PC\_BACK}}, \{P, \#, P, P, 0\}, \{S, S, S, S, L\})$$

$$\delta(q_{\text{PC\_BACK}}, \{P, \#, P, P, 1\}) = (q_{\text{PC\_BACK}}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

$$\delta(q_{\text{PC\_BACK}}, \{P, \#, P, P, \#\}) = (q_P, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

## STORE r1, 12

Dekodiranje vrste naredbe i oznake prvog registra:

$$\delta(q_P, \{P, \#, P, P, P\}) = (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_N, \{P, 0, P, P, P\}) = (q_{\text{MEM}}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{\text{MEM}}, \{P, 1, P, P, P\}) = (q_{\text{ST}}, \{P, 1, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{\text{ST}}, \{P, 0, P, P, P\}) = (q_{\text{ST}}, \{P, 0, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_{\text{ST}}, \{P, 0, P, P, P\}) = (q_{\text{ST}}, \{P, 0, P, P, P\}, \{S, S, R, S, S\})$$

$$\delta(q_{\text{ST}}, \{P, 0, \#, P, P\}) = (q_{\text{POZ1}}, \{P, 0, S, P, P\}, \{R, R, L, S, R\})$$

Traženje adrese 1100:

Čitanje prvog bita adrese, i pošto je on 1, to znači da moramo preskočiti 8 blokova:

$$\delta(q_{POZ1}, \{\#, 1, P, P, \#\}) = (q_{ADR8}, \{\#, 1, P, P, 1\}, \{R, S, S, S, L\})$$

$$\delta(q_{ADR8}, \{0, 1, P, P, P\}) = (q_{ADR8}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR8}, \{1, 1, P, P, P\}) = (q_{ADR8}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR8}, \{\#, 1, P, P, P\}) = (q_{ADR7}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{0, 1, P, P, P\}) = (q_{ADR7}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{1, 1, P, P, P\}) = (q_{ADR7}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR7}, \{\#, 1, P, P, P\}) = (q_{ADR6}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{0, 1, P, P, P\}) = (q_{ADR6}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{1, 1, P, P, P\}) = (q_{ADR6}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR6}, \{\#, 1, P, P, P\}) = (q_{ADR5}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{0, 1, P, P, P\}) = (q_{ADR5}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{1, 1, P, P, P\}) = (q_{ADR5}, \{1, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR5}, \{\#, 1, P, P, P\}) = (q_{ADR4}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{0, 1, P, P, P\}) = (q_{ADR4}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{S, 1, P, P, P\}) = (q_{ADR4}, \{S, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{\#, 1, P, P, P\}) = (q_{ADR3}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{0, 1, P, P, P\}) = (q_{ADR3}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{\#, 1, P, P, P\}) = (q_{ADR2}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{0, 1, P, P, P\}) = (q_{ADR2}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{\#, 1, P, P, P\}) = (q_{ADR1}, \{\#, 1, P, P, P\}, \{R, S, S, S, R\})$$

$$\delta(q_{ADR1}, \{0, 1, P, P, 1\}) = (q_{ADR1}, \{0, 1, P, P, 1\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 1\}) = (q_{POZ2}, \{\#, 1, P, P, 1\}, \{S, R, S, S, S\})$$

Prelazimo na obradu drugog bita adrese, a pošto je on 1, preskačemo 4 bita:

$$\delta(q_{POZ2}, \{\#, 1, P, P, 1\}) = (q_{ADR4}, \{\#, 1, P, P, 2\}, \{R, S, S, S, L\})$$

$$\delta(q_{ADR4}, \{0, 1, P, P, P\}) = (q_{ADR4}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR4}, \{\#, 1, P, P, P\}) = (q_{ADR3}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{0, 1, P, P, P\}) = (q_{ADR3}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR3}, \{\#, 1, P, P, P\}) = (q_{ADR2}, \{\#, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{0, 1, P, P, P\}) = (q_{ADR2}, \{0, 1, P, P, P\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR2}, \{\#, 1, P, P, P\}) = (q_{ADR1}, \{\#, 1, P, P, P\}, \{R, S, S, S, R\})$$

$$\delta(q_{ADR1}, \{0, 1, P, P, 1\}) = (q_{ADR1}, \{0, 1, P, P, 1\}, \{R, S, S, S, S\})$$

$$\delta(q_{ADR1}, \{\#, 1, P, P, 2\}) = (q_{POZ3}, \{\#, 1, P, P, 2\}, \{S, R, S, S, S\})$$

Prelazimo na obradu ostalih bitova, no pošto su oba 0, preskačemo ih:

$$\delta(q_{POZ3}, \{\#, 0, P, P, 2\}) = (q_{POZ4}, \{\#, 0, P, P, 3\}, \{S, R, S, S, S\})$$

$$\delta(q_{POZ4}, \{\#, 0, P, P, 3\}) = (q_{ADR\_END}, \{\#, 0, P, P, \#\}, \{S, R, S, S, S\})$$

Pomičemo glave oba registra kako bi pročitali oznaku naredbe:

$$\delta(q_{ADR\_END}, \{\#, \#, P, P, P\}) = (q_{ADR\_END}, \{\#, \#, P, P, P\}, \{S, S, R, R, S\})$$

Prelazak u stanje STR1 pomoću oznake S na traci tog registra:

$$\delta(q_{ADR\_END}, \{\#, \#, S, \#, P\}) = (q_{STR1}, \{\#, \#, \#, \#, P\}, \{R, S, R, L, S\})$$

Kopiranje broja 10011110 u memoriju iz registra R1:

$$\delta(q_{STR1}, \{0, \#, 1, P, P\}) = (q_{STR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{1, \#, 0, P, P\}) = (q_{STR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{1, \#, 1, P, P\}) = (q_{STR1}, \{1, \#, 1, P, P\}, \{R, S, R, S, S\})$$

$$\delta(q_{STR1}, \{0, \#, 0, P, P\}) = (q_{STR1}, \{0, \#, 0, P, P\}, \{R, S, R, S, S\})$$

Vraćanje glave registra R1 na početak:

$$\delta(q_{STR1}, \{\#, \#, \#, P, P\}) = (q_{R1\_BACK}, \{\#, \#, \#, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, 0, P, P\}) = (q_{R1\_BACK}, \{\#, \#, 0, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, 1, P, P\}) = (q_{R1\_BACK}, \{\#, \#, 1, P, P\}, \{S, S, L, S, S\})$$

$$\delta(q_{R1\_BACK}, \{\#, \#, \#, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, \#, P, P\}, \{L, S, L, S, S\})$$

Vraćanje prve glave na početak:

$$\delta(q_{ADR\_BACK}, \{0, \#, P, P, P\}) = (q_{ADR\_BACK}, \{0, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{1, \#, P, P, P\}) = (q_{ADR\_BACK}, \{1, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{\#, \#, P, P, P\}) = (q_{ADR\_BACK}, \{\#, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{S, \#, P, P, P\}) = (q_{ADR\_BACK}, \{S, \#, P, P, P\}, \{L, S, S, S, S\})$$

$$\delta(q_{ADR\_BACK}, \{P, \#, P, P, P\}) = (q_{PC\_POZ1}, \{P, \#, P, P, P\}, \{S, S, S, S, R\})$$

Pozicioniranje na zadnji bit PC-a:

$$\delta(q_{PC\_POZ1}, \{P, \#, P, P, \#\}) = (q_{PC\_POZ}, \{P, \#, P, P, \#\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_POZ}, \{P, \#, P, P, 0\}) = (q_{PC\_POZ}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_POZ}, \{P, \#, P, P, 1\}) = (q_{PC\_POZ}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_POZ}, \{P, \#, P, P, \#\}) = (q_{PC\_1}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

Povećanje za 1:

$$\delta(q_{PC\_1}, \{P, \#, P, P, 1\}) = (q_{PC\_2}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

$$\delta(q_{PC\_2}, \{P, \#, P, P, 1\}) = (q_{PC\_2}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

$$\delta(q_{PC\_2}, \{P, \#, P, P, 0\}) = (q_{PC\_2\_END}, \{P, \#, P, P, 1\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_2\_END}, \{P, \#, P, P, 1\}) = (q_{PC\_2\_END}, \{P, \#, P, P, 0\}, \{S, S, S, S, R\})$$

$$\delta(q_{PC\_2\_END}, \{P, \#, P, P, \#\}) = (q_{PC\_BACK}, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

Vraćanje glave PC-a na početak i prelazak na sljedeću naredbu:

$$\delta(q_{PC\_BACK}, \{P, \#, P, P, 0\}) = (q_{PC\_BACK}, \{P, \#, P, P, 0\}, \{S, S, S, S, L\})$$

$$\delta(q_{PC\_BACK}, \{P, \#, P, P, 1\}) = (q_{PC\_BACK}, \{P, \#, P, P, 1\}, \{S, S, S, S, L\})$$

$$\delta(q_{PC\_BACK}, \{P, \#, P, P, \#\}) = (q_P, \{P, \#, P, P, \#\}, \{S, S, S, S, L\})$$

Prelazak u stanje  $q_N$  i čitanje znaka \$ koji označava kraj instrukcijskog dijela, što znači da smo obradili sve naredbe, i time prelazimo u prihvatljivo stanje:



$$\delta(q_P, \{P, \#, P, P, P\}) = (q_N, \{P, \#, P, P, P\}, \{S, R, S, S, S\})$$

$$\delta(q_N, \{P, \$, P, P, P\}) = (q_{END}, \{P, \$, P, P, P\}, \{S, L, S, S, S\})$$

Pošto više nema definiranih prijelaza za nastavak, TS završava s radom.

## 4. Zaključak

Razlaganjem naizgled vrlo složenog zadatka na nekoliko manje kompliciranih problema, smatram da sam uspio i manje upućenom čitatelju objasniti svoje rješenje svakog od njih i bar malo približiti tematiku rješavanja problema Turingovim strojem.

Najveći problem mi je predstavljalo pozicioniranje glave na pravi blok u memoriji, no mislim da sam to rješio na relativno jednostavan način.

Pri rješavanju mogao sam na puno drugačijih načina pristupiti svakom od njih, no smatram da sam svojim odabirom uspio bar donekle minimizirati konačan broj prijelaza i u konačnici što više pojednostaviti zadatak.

Npr. mogao sam odmah u početku odlučiti da ću koristiti jednu dodatnu traku i koristiti je kao radnu traku koja bi mi donekle pomogla u traženju adrese kod LOAD / STORE naredbi i kod zbrajanja i oduzimanja, no isto tako to bi mi odmoglo u nekim drugim stvarima.

Također, kod opisane dileme sa PC-om mogao sam postupiti tako da bez obzira što se naredbe izvršavaju slijedno, nizom prijelaza spojim PC i glavu za čitanje naredbi, no mislim da je i ovo rješenje u redu.

Na kraju, smatram da je konačna brojka od 233 prijelaza sasvim prihvatljiva za ovaj tip zadatka.