
YHTEENVETO TTK-91 KÄSKYKANNASTA

This document is also available in [english](#)

Symbolisessa konekielessä käskyt esitetään muodossa:

VIITE OPER Rj,M ADDR(Ri)

| | | |
|-------|------|---|
| missä | OPER | käskyn symbolinen nimi |
| | Rj | ensimmäinen operandi (rekisteri R0..R7) |
| | M | osoitusmoodi: |
| | = | välitön operandi |
| | | suora osoitus (tyhjä, ei siis merkitä) |
| | @ | epäsuora osoitus |
| | ADDR | osoiteosa (muistiosoite tai välitön operandi) |
| | Ri | mahdollinen indeksirekisteri (rekisteri R0..R7) |

Jos jollakin käskyn osalla ei ole merkitystä, sen voi jättää merkitsemättä. Käskyn eteen voi laittaa viitteeksi symbolisen osoitteen, jonka tulee sisältää vähintään yksi muu kuin numeromerkki. Kelvollisia merkkejä ovat A-Ö, 0-9 ja `_`. Vain 8 ensimmäistä merkkiä huomioidaan.

Lähes kaikille käskyille on käytettävissä seuraavat muodot:

| | |
|-------------------|-----------------------------------|
| OPER Rj,ADDR | suora muistiosoitus |
| OPER Rj,Ri | suora rekisteriosoitus |
| OPER Rj,=ADDR | välitön operandi |
| OPER Rj,@ADDR | epäsuora muistiosoitus |
| OPER Rj,@Ri | epäsuora muistiosoitus |
| OPER Rj,ADDR(Ri) | indeksoitu osoitus |
| OPER Rj,=ADDR(Ri) | indeksoitu välitön operandi |
| OPER Rj,@ADDR(Ri) | indeksoitu epäsuora muistiosoitus |

Poikkeuksia:

| | |
|-----------------|---|
| STORE | jälkimmäinen operandi on aina kohdeosoite, ei voi olla rekisteri, tai vakio |
| POP | jälkimmäisen operandin tulee aina olla rekisteri |
| PUSHR | jälkimmäisellä operandilla ei merkitystä |
| POPR | jälkimmäisellä operandilla ei merkitystä |
| NOT | jälkimmäisellä operandilla ei merkitystä |
| NOP | operandeilla ei merkitystä |
| Haarauma | jälkimmäinen operandi on aina kohdeosoite, ei voi olla vakio. Tilarekisteriä tutkivissa käskyissä on ensimmäinen operandi merkityksetön |

TTK-91 symbolisen konekielen käskyt

Tiedonsiirtokäskyt

| | |
|--------------|--|
| LOAD | vie jälkimmäisen operandin arvon rekisterin Rj arvoksi. |
| STORE | talleta rekisterissä Rj oleva kokonaisluku jälkimmäisen operandin arvoksi. |
| IN | lukee jälkimmäisenä operandina kerrotulta laitteelta kokonaisluvun rekisteriin Rj (esim näppäimistöltä lukeminen: IN R1, =KBD) |
| OUT | tulostaa rekisterissä Rj olevan kokonaisluvun jälkimmäisenä operandina kerrotulle laitteelle (esim näytölle kirjoitus: OUT R1, =CRT) |

Aritmeettiset ja loogiset käskyt

| | |
|-------------|--|
| ADD | (add) lisää rekisterissä Rj olevaan lukuun jälkimmäisen operandin arvon. |
| SUB | (subtract) vähennä rekisterissä Rj olevasta luvusta jälkimmäisen operandin arvon. |
| MUL | (multiply) kerro rekisterissä Rj oleva luku jälkimmäisen operandin arvolla. |
| DIV | (divide) jaa rekisterissä Rj oleva luku jälkimmäisen operandin arvolla ja vie tulos rekisteriin Rj. |
| MOD | (modulo) jaa rekisterissä Rj oleva luku jälkimmäisen operandin arvolla ja vie jakojäännös rekisteriin Rj. |
| AND | (boolean AND) looginen JA-operaatio. |
| OR | (boolean OR) looginen TAI-operaatio. |
| XOR | (boolean XOR) looginen poissulkeva TAI-operaatio. |
| NOT | (boolean NOT) kääntää kaikki rekisterin Rj bitit. |
| SHL | (shift left) siirrä rekisterin Rj bittejä vasemmalle toisen operandin ilmoittama määrä. Täytä oikeaa päätä 0-biteillä. |
| SHR | (shift right) kuten SHL, mutta siirrä oikealle. |
| SHRA | (arithmetic shift right) aritmeettinen siirto oikealle, säilyttää etumerkin. |
| COMP | (compare) vertaa ensimmäisen operandin arvoa toisen operandin arvoon ja aseta vertailun tulos tilarekisterin bitteihin SR L=pienempi, E=yhtäsuuri, G=suurempi. |

Haarautumiskäskyt

| | |
|-------------|--|
| JUMP | (unconditional jump) ehdoton hyppy toisen operandin ilmaisemaan kohdeosoitteeseen. |
| JNEG | (jump if negative) jos Rj < 0, niin hyppää jälkimmäisenä operandina olevaan osoitteeseen, muuten jatka seuraavasta käskystä. |

| | |
|--------------|---|
| JZER | (jump if zero) jos $R_j = 0$ |
| JPOS | (jump if positive) jos $R_j > 0$ |
| JNNEG | (jump if not negative) jos $R_j \geq 0$ |
| JNZER | (jump if not zero) jos $R_j \neq 0$ |
| JNPOS | (jump if not positive) jos $R_j \leq 0$ |
| JLES | (jump if less) jos tilarekisterin SR bitti L asetettu, niin hyppää jälkimmäisenä operandina olevaan osoitteeseen, muuten jatkaa seuraavasta käskystä (käyttö COMP-käskyn yhteydessä). |
| JEQU | (jump if equal) jos bitti E asetettu |
| JGRE | (jump if greater) jos bitti G asetettu |
| JNLES | (jump if not less) jos bitti E tai G asetettu |
| JNEQU | (jump if not equal) jos bitti L tai G asetettu |
| JNGRE | (jump if not greater) jos bitti L tai E asetettu |

Pinokäskyt

Käskyn ensimmäinen operandi rekisteri R_j osoittaa pinon huipulle pinon päällimmäiseen alkioon. Pino-osoittimena käytetään tavallisesti rekisteriä SP (eli R6).

| | |
|--------------|---|
| PUSH | kasvattaa pino-osoittimen R_j arvoa yhdellä ja tallentaa jälkimmäisen operandin pinon päällimmäiseksi alkioksi. |
| POP | poistaa pinosta päällimmäinen alkion ja vie sen jälkimmäisenä operandina kerrottuun rekisteriin (HUOM: aina rekisteri). Vähentää pino-osoittimen R_j arvoa yhdellä. |
| PUSHR | tallentaa rekisterit R0, R1, R2, R3, R4, R5 ja R6 (SP) pinoon, tässä järjestyksessä. Pino-osoittimen R_j arvo kasvaa seitsemällä. |
| POPR | asettaa rekisterit R6 (SP), R5, R4, R3, R2, R1 ja R0 pinosta löytyvillä arvoilla, tässä järjestyksessä. Pino-osoittimen R_j arvo vähenee seitsemällä. |

Aliohjelmakäskyt

| | |
|-------------|---|
| CALL | (call procedure) aliohjelmakutsu, eli kontrollin siirto toisen operandin ilmoittamaan osoitteeseen. Tallettaa paluusoitteen ja frame-pointterin (FP eli R7) pinoon, jonka huipulle osoittaa R_j . |
| EXIT | palaa aliohjelmasta kutsua seuraavaan käskyyn. Hakee pinosta (R_j) frame-pointterin ja paluusoitteen. Jälkimmäisenä operandina pinossa välitettyjen parametrien lukumäärä (myös nämä poistetaan pinosta). |

Systeemikutsut

| | |
|--------------|---|
| SVC | (supervisor call) käyttöjärjestelmän palvelurutiinin kutsu. Ensimmäisenä operandina pinon huippu Rj ja toisena operandina palvelun numero. Alla palvelunumerot: |
| HALT | Lopettaa ohjelman suorituksen. |
| TIME | Antaa kellonajan. Pinossa välitettävä osoitteet, jonne halutaan tunnit, minuutit, sekunnit (HUOM: järjestys!). |
| DATE | Antaa päiväyksen. Pinossa välitettävä osoitteet, jonne halutaan vuosi, kuukausi ja päivä. (HUOM: järjestys!). |
| READ | Lukee kokonaisluvun. Pinossa välitettävä osoite, jonne luku halutaan luettavan. |
| WRITE | Kirjoittaa kokonaisluvun. Pinossa välitettävä tulostettava arvo. |

Muut käskyt

NOP (no operation) ei toimintoa, operandeilla ei merkitystä

Kääntäjän ohjauskäskyt (valekäskyt)

Kääntäjän ohjauskäskyt antavat ohjeita symbolisen konekielen kääntäjälle. Ne EIVÄT ole varsinaisia symbolisen konekielen käskyjä.

tunnus **EQU** arvo

Samaistuskäsky EQU määrittelee symboliselle tunnukselle kokonaislukuarvon. Tunnusta voi käyttää käskyn ADDR-kentässä, jolloin se käsitellään kuten vastaavaan paikkaan kirjoitettu numeroarvo.

tunnus **DC** arvo

Muistinvarauskäsky DC (data constant) varaa yhden muistisanan vakiota varten, samaistaa varatun muistipaikan osoitteen ja symbolisen osoitteen "tunnus" sekä asettaa varatun muistipaikan sisällöksi luvun "arvo". Tunnusta voi käyttää käskyn ADDR-kentässä kuten muistiosoitetta.

tunnus **DS** koko

Muistinvarauskäsky DS (data segment) varaa muistialueen, jonka koko on "koko" (sanoina) ja samaistaa varatun muistialueen alkuosoitteen ja symbolisen osoitteen "tunnus". Käytetään globaalien muuttujien tilanvaraukseen. Tunnusta voi käyttää käskyn ADDR-kentässä kuten muistiosoitetta.

Osoitusmoodit

Välitön operandi

Operandin arvo voidaan määritellä itse käskyssä, sen sijaan että määriteltäisiin muistiosoite tai rekisteri josta arvo noudetaan. Välitön operandi on 16-bittinen etumerkillinen kokonaisluku, joten tuettu arvoalue on -32768...32767.

Esim:

1) `LOAD R1, =100` Vie rekisteriin R1 arvon 100.

Suora osoitus

Suorassa osoituksessa määritellään operandin sisältävä muistipaikka tai rekisteri.

Esim:

2) `LOAD R1, 100` Vie rekisteriin R1 muistipaikan 100 sisällön.
3) `LOAD R1, R2` Vie rekisteriin R1 rekisterin R2 sisällön.

Epäsuora osoitus

Epäsuorassa osoituksessa määritellään operandin sisältävän muistipaikan osoite.

Esim:

4) `LOAD R1, @R2` Vie rekisteriin R1 muistipaikan sisällön, jonka osoite saadaan rekisteristä R2.
5) `LOAD R1, @100` Vie rekisteriin R1 muistipaikan sisällön, jonka osoite saadaan muistipaikasta 100.

Indeksoitu osoitus

Indeksoidussa osoituksessa lasketaan ensin yhteen vakio ja määritellyn rekisterin arvo. Tämän jälkeen suoritetaan välitön, suora tai epäsuora operandihaku.

Esim:

6) `LOAD R1, =100(R2)` Vie rekisteriin R1 rekisterin R2 ja vakion 100 summan.
7) `LOAD R1, 100(R2)` Vie rekisteriin R1 muistipaikan sisällön, jonka osoite on vakion 100 ja rekisterin R2 summa.
8) `LOAD R1, @100(R2)` Vie rekisteriin R1 muistipaikan sisällön, jonka osoite löytyy muistipaikastasta 100+R2.

Osoitus laitetasolla

Vaikka yllä on esitelty kahdeksan erilaista osoitustapaa,

```
"LOAD R1, =10"      == "LOAD R1, =10(R0)"
"LOAD R1, R2"       == "LOAD R1, =0(R2)"
"LOAD R1, @R2"      == "LOAD R1, 0(R2)"
"LOAD R1, @10"      == "LOAD R1, @10(R0)"
"LOAD R1, 10"       == "LOAD R1, 10(R0)"
```

| | |
|------------------------------|--|
| a) LOAD R3, R0 | Vie rekisteriin R3 arvon 0, riippumatta R0:n arvosta |
| b) STORE R0, X LOAD R3, X | Tallentaa rekisterin R0 arvon muistipaikkaan X Vie muistipaikan X sisällön rekisteriin R3 |

| | | | | | | |
|----------------|-------|----|----|-------|---------------------------|--|
| 8 bittiä | | 3b | 2b | 3b | 16 bittiä | |
| Operaatiokoodi | | Rj | M | Ri | osoite / välitön operandi | |
| 31 | 24 23 | | | 16 15 | 00 | |

| Käsky | Koodi | Desimaali | Heksadesim. |
|-------|-----------|-----------|-------------|
| NOP | 0000 0000 | 0 | 00 |
| STORE | 0000 0001 | 1 | 01 |
| LOAD | 0000 0010 | 2 | 02 |
| IN | 0000 0011 | 3 | 03 |
| OUT | 0000 0100 | 4 | 04 |
| ADD | 0001 0001 | 17 | 11 |
| SUB | 0001 0010 | 18 | 12 |
| MUL | 0001 0011 | 19 | 13 |
| DIV | 0001 0100 | 20 | 14 |
| MOD | 0001 0101 | 21 | 15 |
| AND | 0001 0110 | 22 | 16 |
| OR | 0001 0111 | 23 | 17 |
| XOR | 0001 1000 | 24 | 18 |
| SHL | 0001 1001 | 25 | 19 |
| SHR | 0001 1010 | 26 | 1A |

| | | | |
|-------|-----------|-----|----|
| NOT | 0001 1011 | 27 | 1B |
| SHRA | 0001 1100 | 28 | 1C |
| COMP | 0001 1111 | 31 | 1F |
| JUMP | 0010 0000 | 32 | 20 |
| JNEG | 0010 0001 | 33 | 21 |
| JZER | 0010 0010 | 34 | 22 |
| JPOS | 0010 0011 | 35 | 23 |
| JNNEG | 0010 0100 | 36 | 24 |
| JNZER | 0010 0101 | 37 | 25 |
| JNPOS | 0010 0110 | 38 | 26 |
| JLES | 0010 0111 | 39 | 27 |
| JEQU | 0010 1000 | 40 | 28 |
| JGRE | 0010 1001 | 41 | 29 |
| JNLES | 0010 1010 | 42 | 2A |
| JNEQU | 0010 1011 | 43 | 2B |
| JNGRE | 0010 1100 | 44 | 2C |
| CALL | 0011 0001 | 49 | 31 |
| EXIT | 0011 0010 | 50 | 32 |
| PUSH | 0011 0011 | 51 | 33 |
| POP | 0011 0100 | 52 | 34 |
| PUSHR | 0011 0101 | 53 | 35 |
| POPR | 0011 0110 | 54 | 36 |
| SVC | 0111 0000 | 112 | 70 |

Osoitusmoodit

| Binary | Dec | Osoitusmoodi |
|--------|-----|---------------------|
| 00 | 0 | indeksoitu välitön |
| 01 | 1 | indeksoitu suora |
| 10 | 2 | indeksoitu epäsuora |