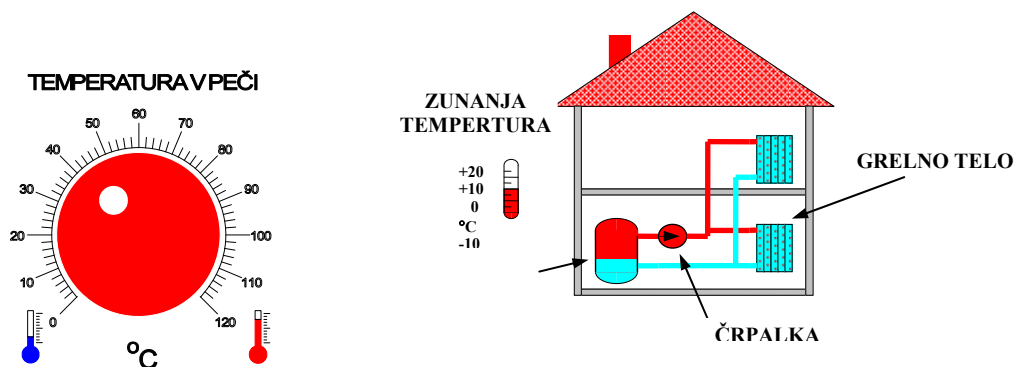


## KRMILJENJE

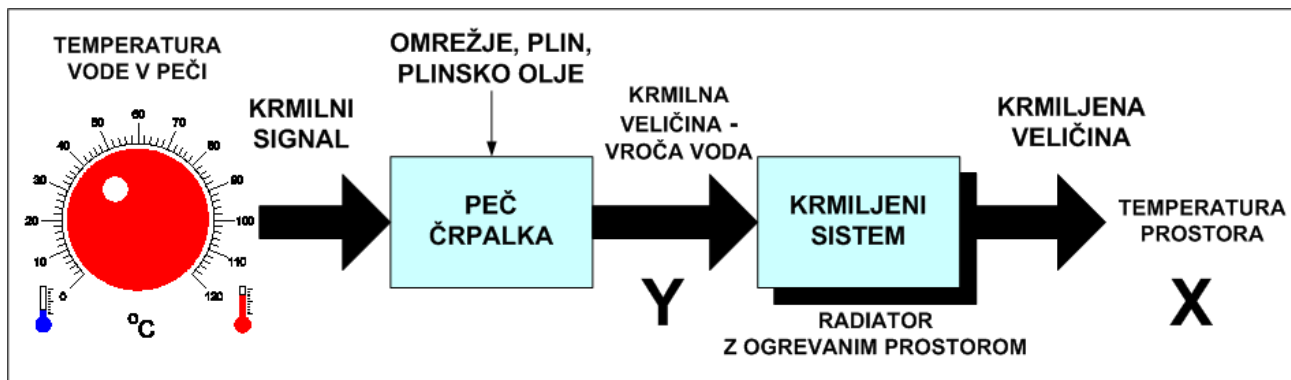
Temperatura prostora v katerem živimo je za nas pomembna veličina. Zato želimo da bo ta temperatura čim manj odvisna od zunanjih vplivov in da bo ostala na vnaprej postavljeni želeni vrednosti.

Želeno vrednost temperature prostora lahko dosežemo npr. z nastavitvijo primerne temperature tople vode, ki kroži skozi grelna tela.



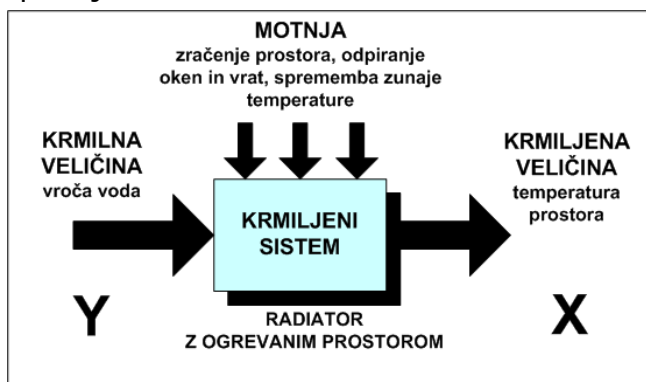
Da bomo naš prostor ogreli na želeno temperaturo, bomo spreminjali temperaturo vode v peči. Ta bo odvisna od zunanje temperature.

To **krmiljenje** notranje temperature v odvisnosti od zunanje bo uspešno le toliko časa, dokler se ne spremenijo druge vplivne veličine kot je zračenje prostora, sprememba zunanje temperature in podobno.



## Blokovna shema krmiljenja temperature v prostoru

Simbolično lahko predstavimo sistem **krmiljenja** z blokovno shemo, kot jo prikazuje spodnja slika.



Vplivne veličine so označene z y, x in z.

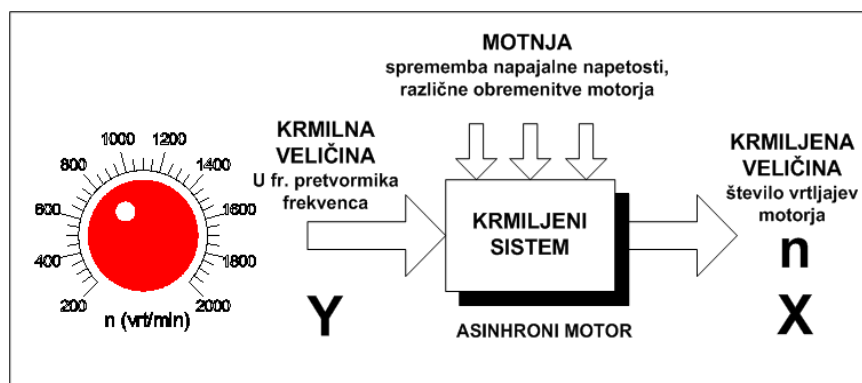
y - krmilna veličina  
x - krmiljena veličina  
z - motilna veličina

S spreminjanjem vrednosti **y** namerno vplivamo na izhodno veličino **x** zato bomo **y** imenovali **krmilna veličina**. Z njo krmilimo izhodno veličino **x**, ki jo imenujemo **krmiljena veličina**.

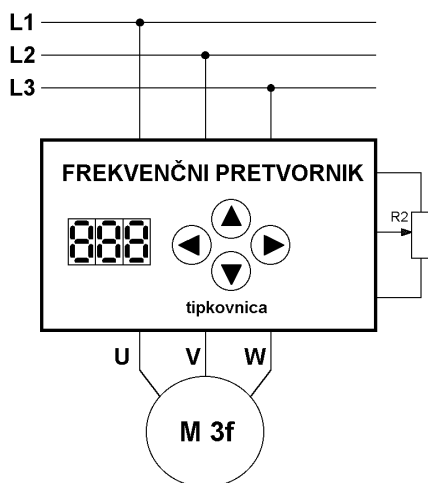
Na krmiljeno veličino vplivajo še druge vhodne veličine, ki se spreminjajo brez našega nadzora, torej neželjeno. Te vplivne veličine imenujemo **motnje** in jih označujemo s črko **z**.

Iz tega lahko povzamemo, da **sistem S** generira izhodno veličino **x**, na njeno trenutno vrednost pa vplivajo motnje **z** in krmilna veličina **y**.

### Primer: Krmiljenje vrtljajev asinhronnega motorja



S potenciometrom ali tipkovnico na frekvenčnem pretvorniku nastavimo želeno število vrtljajev **n**. Nastavljeno število vrtljajev bo toliko časa nespremenjeno, dokler bodo nespremenjene druge vplivne veličine, kot so napajalna napetost ali obremenitev motorja.



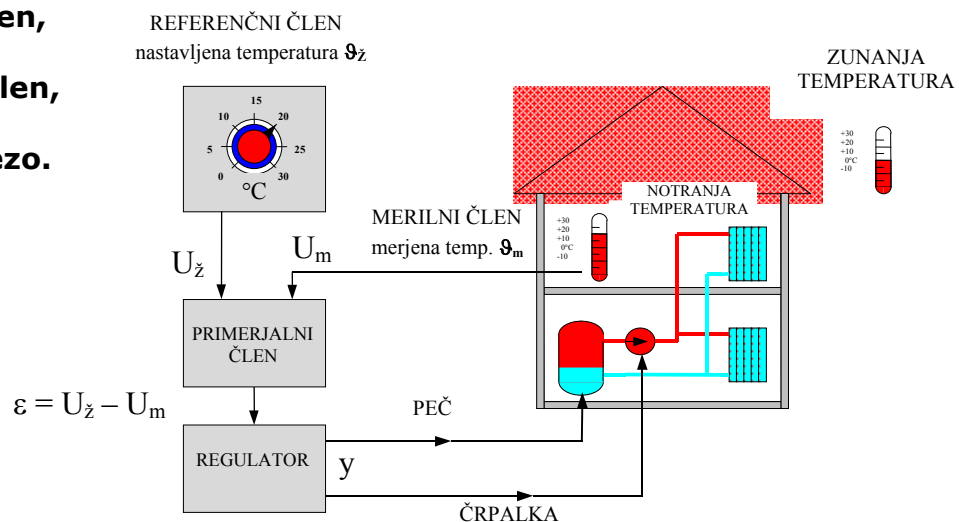
## REGULACIJE

Če želimo krmiljeno veličino (temperaturo) nadzorovati in v primeru odstopanja od želene vrednosti posredovati, jo moramo **meriti**. Dobljeno vrednost moramo nato primerjati z nastavljeno želeno vrednostjo in v primeru odstopanja naj poseben **dodaten sistem** poskrbi za izravnavo (popravo). Tak sistem imenujemo **regulacijski sistem**.

Za spreminjanje krmilnega v regulacijski sistem moramo dodati nekatere funkcije, (referenčni člen, meritev regulirane izhodne veličine  $x$ , primerjanje merjene in želene vrednosti ter sistem, ki bo sposoben posredovati v primeru neskladja).

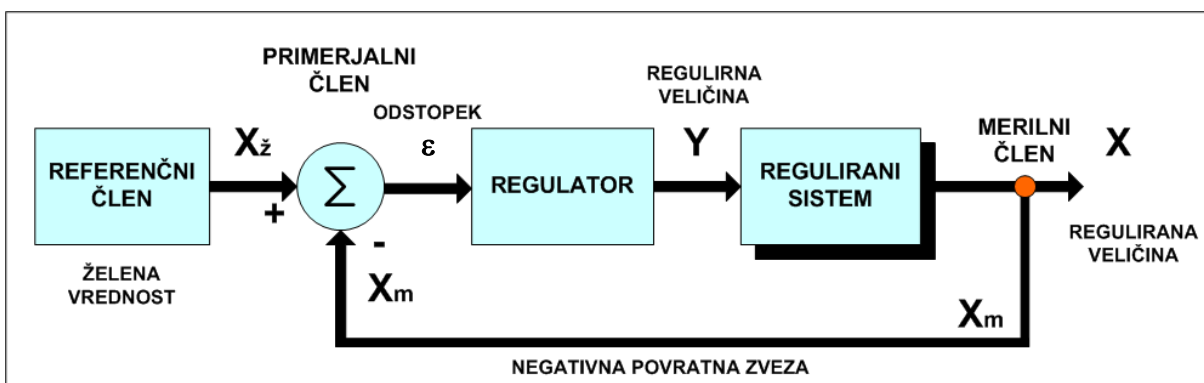
Da bi dosegli regulacijo (ne pa krmiljenje), potrebujemo naslednje dodatne člene:

- referenčni člen,
- merilni člen,
- primerjalni člen,
- regulator,
- povratno zvezo.



Željeno vrednost temperature  $\vartheta_z$  lahko sedaj nastavimo z **referenčnim členom**. Z **merilnim členom** merimo temperaturo v prostoru  $\vartheta_m$ . Oba podatka o temperaturi vodimo na **primerjalni člen**, ki primerja želeno vrednost  $\vartheta_z$  z izmerjeno vrednostjo regulirane veličine  $\vartheta_m$ . Na svojem izhodu nam primerjalni člen ustvarja **regulacijsko odstopanje**  $\varepsilon$ , ki ga **vodimo v regulator**. **Regulator** na svojem izhodu ustvarja **regulirano veličino**  $y$  sorazmerno regulacijskemu odstopanju  $\varepsilon$  (večji  $\varepsilon$  - večji  $y$ , manjši  $\varepsilon$  - manjši  $y$ ). Z regulirano veličino  $y$  deluje regulator na regulirani sistem (peč in črpalko) tako, da regulirana veličina  $x$  sledi nastavljeni željeni vrednosti.

**Regulacijski sistem lahko predstavimo z blokovno shemo.**



**Referenčni člen** daje na svojem izhodu **želeno vrednost**  $x_z$  zato opravlja funkcijo dajalnika želene vrednosti.

**Primerjalni člen** primerja želeno vrednost  $x_z$  z izmerjeno vrednostjo regulirane veličine  $x_m$  in na svojem izhodu ustvarja regulacijski odstop  $\varepsilon$ , za katerega velja, da je:  $\varepsilon = x_z - x_m$

**Regulator** je krmiljen z regulacijskim odstopanjem  $\varepsilon$ . Na svojem izhodu ustvarja regulirno veličino  $y$ , s katero deluje (preko izvršilnih členov (relejev, servomotorjev, hidravličnih in elektromehanskih sistemov)) na **regulirani sistem S** tako, da regulirana veličina sledi nastavljeni želeni vrednosti.

**Merilni člen** meri trenutno vrednost **regulirane veličine**  $x$  in jo preko negativne povratne zveze posreduje **primerjalnemu členu**.

## Delitev regulacij

### a) glede na konstrukcijsko izvedbo:

- **analogne** (uporabimo analogne elemente: dajalnik stanj - potenciometer, primerjalnik in regulator operacijski ojačevalnik ...)
- **digitalne** (uporabimo digitalne komponente: tipkovnica, A/D in D/A pretvornik, mikroprocesor )

### b) glede na izvedbo regulatorja in izvršilnega člena:

- **zvezne** (regulator generira zvezne signale npr. 0 do 10 V, s pomočjo katerih krmilimo ustrezne izvršilne člene - generatorji impulzov, tiristorski mostiči, frekvenčni pretvorniki ...)
- **nezvezne** (lahko uporabimo regulatorje, katerih regulirna veličina lahko zavzame le dve vrednosti in sicer:  $Y = 0$  in  $Y = Y_h$ . To so nezvezno delujoči regulatorji; glede na način delovanja so lahko dvopoložajni in tripoložajni)

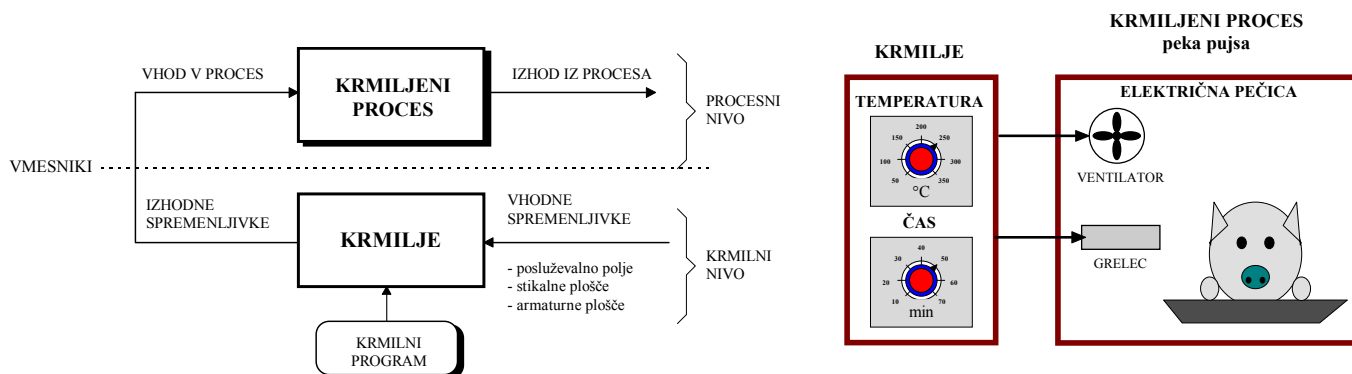
### c) glede na želeno vrednost

- **sledilne regulacije** - Pri tej zvrsti regulacij sledi regulirana veličina spremembam želene vrednosti. Primere teh regulacij najdemo pri numerično krmiljenih obdelovalnih strojih, kjer se na primer v skladu z zahtevano obliko obdelovanca spreminja želena pozicija orodja.
- **časovno programirane regulacije** - V bistvu so to sledilne regulacije, pri katerih se želena vrednost spreminja po programu, ki je časovno odvisen. Takšen regulacijski sistem najdemo na primer v avtomatskih sistemih za ogrevanje prostorov, kjer je višina želene temperature odvisna od dnevnega in tedenskega programa.
- **regulacije s konstantno želeno vrednostjo** - V teh regulacijskih sistemih je želena vrednost časovno neodvisna in konstantna. Zgled teh regulacij so pri regulacijah napetosti, števila vrtljajev, pretoka, gladine in podobno, to je v vseh primerih, kjer želimo obdržati regulirano veličino konstantno in neodvisno od zunanjih vplivnih veličin.

## KRMILJE

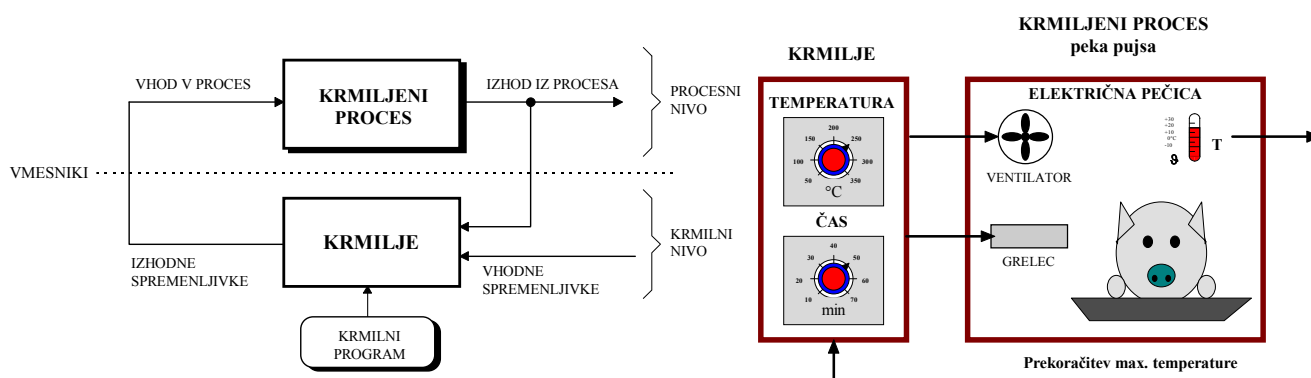
**Povezava krmilja s procesom, ki ga krmilimo,** je lahko izvedena po naslednjih načelih:

- Krmiljenje v odprti zanki**



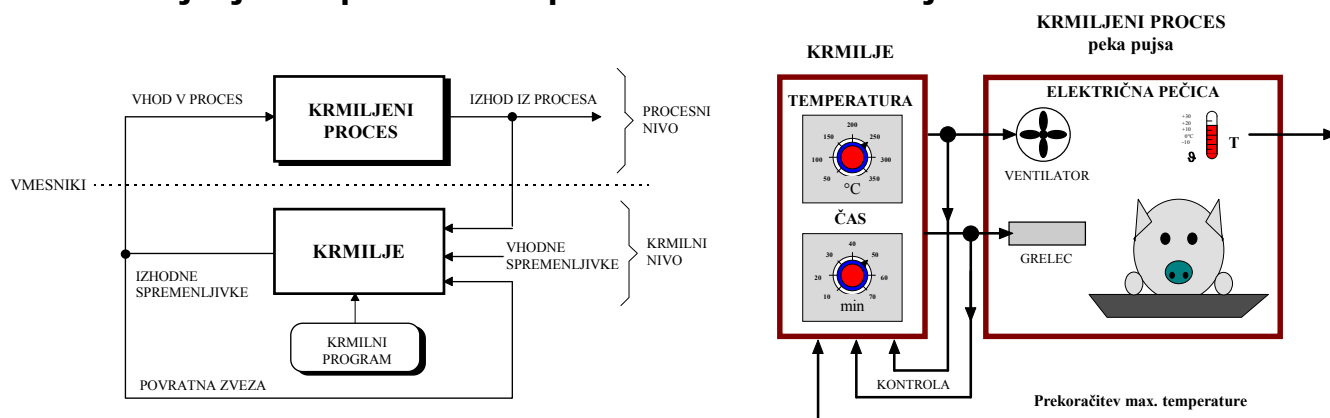
Krmiljenje v odprti zanki je značilnost logičnih krmilij, pri katerih kombinacija vhodnih spremenljivk (različna tipkala za ročno posredovanje) prek krmilnega programa aktivira krmilne izhodne signale.

- Krmiljenje v zaprti zanki**



Krmiljenje v zaprti zanki zaznava prek senzorjev stanja v procesu. Ta stanja se uporabljajo v krmilju kot dodatni vhodni pogoji za izvedbo naslednjega krmilnega programa ali za ponovitev določene krmilne frekvence. Tak sistem krmiljenja poznamo pri koračnih krmiljih.

- Krmiljenje v zaprti zanki in povratno zvezo v krmilju**



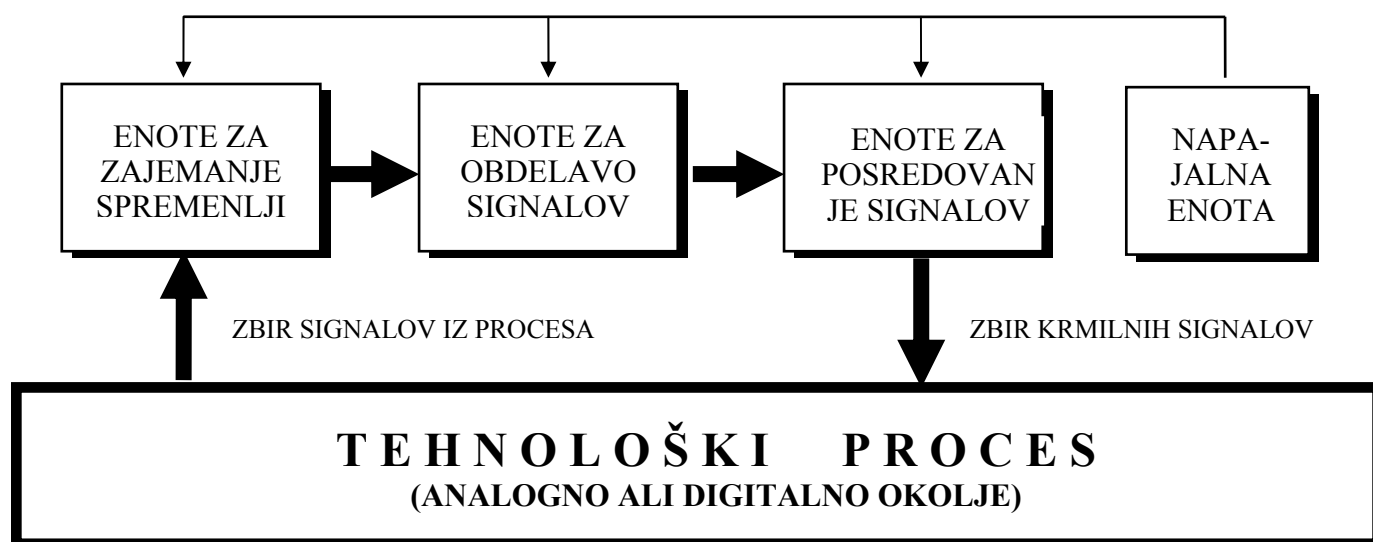
Povratna zveza v krmilju omogoča nadzor nad izhodnimi signali iz krmilja in njihovo testiranje na meji dopustne vrednosti. Take možnosti izrabljamo pri računalniško podprtih krmiljih.

### Funkcijske enote krmilja

Funkcijske enote krmilja, ki tvorijo krmilni sistem in omogočajo povezovanje s procesnim okoljem, lahko razdelimo na:

- enote za zajemanje procesnih spremenljivk,
- enoto za obdelavo signalov,
- enote za posredovanje krmilnih signalov v procesno okolje,
- napajalno enoto.

Če navedene enote povežemo v sistem in procesno okolje, dobimo strukturni model krmilnega sistema.



**Vhodne enote** (enote za zajemanje procesnih spremenljivk) so namenjene zajemanju signalov iz procesnega okolja, ki so potrebni za krmiljenje in imajo funkcijo vhodnih spremenljivk krmilja. Te signale posredujejo stikala, tipkala, različni kontakti in senzorji. Standardne vhodne enote ali enote za zajemanje imajo 8 ali 16 vhodov. Na isti krmilni sistem lahko priključimo več vhodnih enot; vsaki je prirejen določen naslov, prek katerega komunicira s centralno enoto.

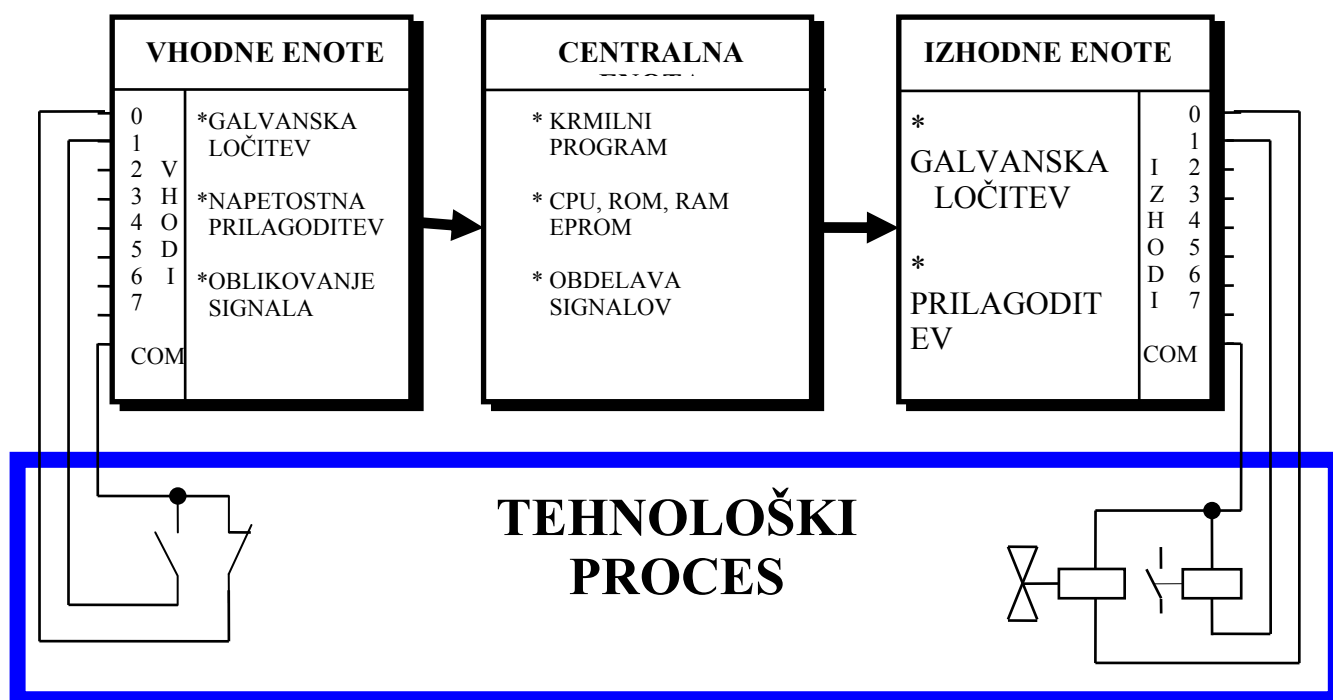
**Enota za obdelavo signalov** ali **centralna enota** je v trajno ožičenih sistemih logično vezje, ki opravlja funkcijo krmilja, v sistemih s prostim programiranjem pa je to centralna procesna enota (CPE ali CPU) ali mikroprocesor s svojimi perifernimi enotami, kot so pomnilnik in vhodno-izhodni vmesnik. Ta enota mora, v skladu s krmilnim programom, ustvariti ustrezne krmilne signale glede na dano kombinacijo vhodnih spremenljivk. Pri trajno ožičenih krmiljih je krmilni program v ožičenju centralne enote. V sistemih s prostim programiranjem pa program sestavlja niz ukazov, ki so vpisani kot binarne vrednosti v pomnilniku centralne enote.

**Enote za posredovanje** krmilnih signalov ali izhodne enote so namenjene oblikovanju takšnih signalov, s katerimi se lahko aktivirajo izvršilni členi, kot so na primer pomožni releji, elektromagnetni ventili, koračni motorji. Standardne izvedbe izhodnih enot so

prirejane za osem izhodov; obstajajo pa tudi enote s 16 izhodi. Na sistem lahko povežemo tudi več izhodnih enot; vsaka mora biti na svojem naslovu.

**Napajalna enota** oskrbuje krmilja z napajalnimi napetostmi. Standardne izvedbe uporabljajo enosmerne napetosti s 24 V nivojem.

Če povzamemo navedene funkcije posameznih enot krmilja, lahko to predstavimo s spodnjo sliko.



### Primerjava centralno procesnih enot krmilnih sistemov

Primerjava centralne enote trajno ožičenih krmilij s centralno enoto krmilij s prostim programiranjem nam da naslednje značilnosti:

**Program krmilja v centralni enoti trajno ožičenih sistemov je izveden:**

- z žičnimi povezavami posameznih komponent krmilnega sistema,
- s povezavami na tiskalnem vezju ali
- s cevnimi povezavami pri pnevmatskih ali hidravličnih krmilnih sistemih.

**Program krmilja v centralni enoti sistemov s prostim programiranjem je dan:**

- z vsebino programskega pomnilnika.

Iz navedenega lahko povzamemo, da so sistemi s trajnim ožičenjem primerni za krmilja v okoljih, kjer se tehnološki proces ne spreminja. Pomnilniško programirana krmilja pa so primerna za okolja, kjer se pogoji in zahtevne spreminjajo; saj je mnogo lažje preprogramirati EPROM pomnilnik kot pa predelati vso centralno enoto.

**Za podrobnejšo oceno sistemov moramo omeniti prednosti in slabosti. Trajno ožičeni sistemi imajo prednost predvsem v tem, da so:**

- manj dovzetni za zunanje motnje, če jih sestavljajo elektromehanske komponente;
- ugodnejši za vzdrževanje, ker imajo večjo tradicijo ter je vzdrževalno osebje zanje bolj usposobljeno in privajeno na sistem (izkušnje in poznavanje komponent ter karakterističnih napak).

**Slabosti teh sistemov se kažejo predvsem v tem, da:**

- zasedajo veliko prostora, kar v največji meri velja za elektromehanska in pnevmatska krmilja;
- so težko prilagodljivi spremembam v okolju, ki ga krmilijo, saj moramo za vsakršno spremembo v programu predelati krmilje tako, da spremenimo povezave med komponentami, ali celo zamenjati določene komponente.

**Prednosti sistemov s prostim programiranjem so:**

- velika prilagodljivost,
- visoka integracija funkcij,
- kratki preklopni časi,
- sposobnost pomnjenja podatkov,
- sposobnost izvajanja aritmetičnih operacij,
- sposobnost zaznavanja in javljanja napak,
- dokumentiranje dogodkov (mejne vrednosti, ročni posegi, spremembe vodenja ...).

**Slabe lastnosti teh sistemov so:**

- visoki začetni stroški,
- bolj zahtevni sistemi glede vstavljanja v obratovanje in vzdrževanje.

Razvoj krmilij kaže na to, da se vse bolj uveljavljajo krmilja s prostim programiranjem, zato bomo tudi v nadaljevanju dali več poudarka tem krmiljem.

**Vprašanja!**

1. Kako razdelimo krmilja glede na funkcijsko odvisnost med vhodnimi in izhodnimi spremenljivkami?
2. O logičnih krmiljih govorimo takrat, ko .....
3. O koračnik krmiljih govorimo takrat, ko ....
4. Glede na izvedbo delimo krmilja na: .....
5. Glavni gradniki elektromehanskih krmilij so: ....
6. Krmilja s prostim programiranjem sestavljajo: .....
7. Katere povezave krmilij s krmiljenim procesom poznaš?
8. Funkcijske enote krmilij so: .....

**Načini opisovanja krmilij**

Z razvojem krmilij je bilo razvitih tudi več pripomočkov za opisovanje in analizo krmilij. Obstaja več načinov, po katerih opišemo krmilje; katerega od teh načinov bomo v nekem primeru uporabili, je predvsem odvisno od funkcije krmilja - izvedbe krmilja. Za logična krmilja bomo uporabili drugačen način opisovanja kot za koračna krmilja, prav tako drugače opišemo elektromehanska trajno ožičena krmilja kot krmilja s prostim programiranjem. V nadaljevanju bomo navedli in opisali nekaj bistvenih načinov opisovanja krmilij.

Ti načini so:

- algoritem krmilja,
- časovni diagram,
- pravilnostna tabela,
- logična funkcija,
- krmilni načrt,
- funkcijski načrt,
- diagram stanj,
- koračna veriga,
- programski modul.

## Algoritem krmilja

Algoritem krmilja je v bistvu tehniški opis, kako naj krmilje deluje, katere količine iz procesa nastopajo kot vhodne spremenljivke, kateri pogoji morajo biti izpolnjeni za ustvarjanje določenih izhodnih krmilnih signalov. Pravilno formuliranje algoritma omogoča, da lahko hitreje in natančno izvedemo takšno krmilje, ki bo zadovoljevalo postavljene zahteve brez neprijetnih naknadnih popravkov. Dovolj natančen algoritem krmilja lahko nastane le v sodelovanju tehnologa, ki pozna tehnološki proces, z načrtovalcem krmilnega programa, ki pa pozna opremo in zmogljivost krmilnega sistema. Čim bolj je natančen algoritem krmilja, tem manj je kasnejših neprijetnih popravkov.

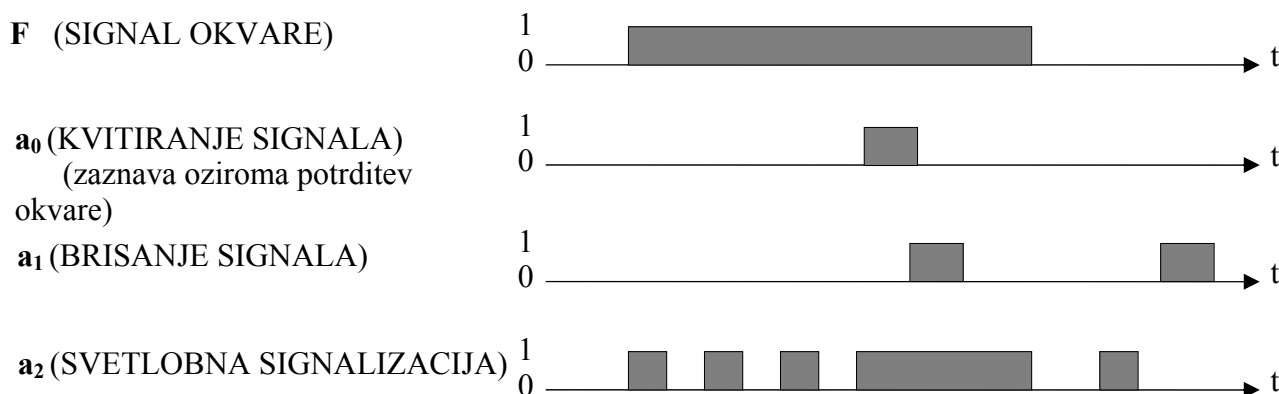
**Zgled:** Algoritem krmilja za krmiljenje dviznega transporterja:

- krmilje naj krmili pomik dvigala za transport izdelkov v skladišču;
- za aktiviranje pomika dvigala imamo tipkali, ki sta označeni z GOR oziroma DOL;
- pogon zaustavimo s pritiskom na tipkalo STOP;
- smer gibanja spremenimo tako, da pritisnemo tipko STOP in nato ustrezno tipko GOR ali DOL;
- krmilje naj bo tako zasnovano, da onemogoči spremembo smeri brez predhodnega aktiviranja tipke STOP;
- hkrati pritisk na tipki GOR in DOL predstavlja nelogični poseg, ki naj ga krmilje ignorira;
- končni legi transporterja se indicirata s končnimi stikali, ki delujejo na izklop pogona.

## Časovni diagram

Časovni diagrami dajejo časovne poteke posameznih vhodnih ali izhodnih spremenljivk ter njihove medsebojne časovne odnose. S časovnimi diagrami zato opisujemo krmilne sisteme, v katerih se stanja spreminjajo v določenih časovnih zaporedjih.

**Zgled:** Registriranje signala okvare



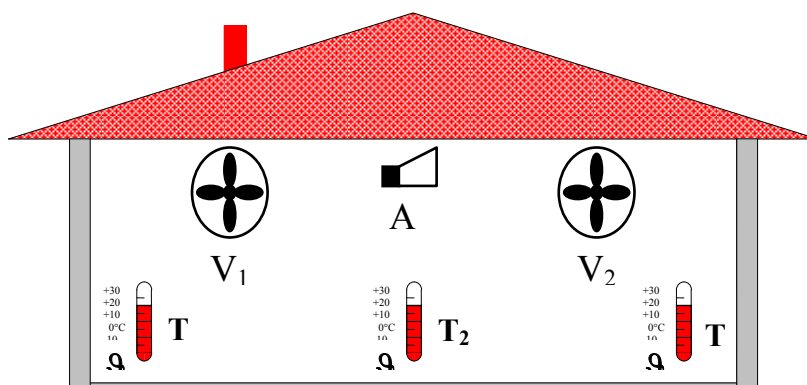
Signal **F** označuje trajanje okvare, ki jo registrira utripajoča luč. Po aktiviranju signala **a<sub>0</sub>** utripajoča luč preide v neprekinjeno svetlobo, dokler ne odpravimo okvare. Po odpravljenju okvare luč utripa s polovično frekvenco, dokler s signalom **a<sub>1</sub>** ne vzpostavimo osnovnega stanja. S signalom **a<sub>1</sub>** moramo resetirati svetlobne signalizacije, dokler je okvara.

## Pravilnostna tabela

Pravilnostna tabela je zelo ugodna za analizo in sintezo logičnih krmilij. Iz nje je razvidno, kdaj so aktivirane posamezne izhodne funkcije (gretje, hlajenje, motor levo, motor desno, dvig, spust, itd). V prvih stolpcih podamo vse kombinacije vhodnih vplivnih spremenljivk, v desnih stolpcih pa izhodne spremenljivke. V stolpcih izhodnih spremenljivk zapišemo logična stanja 1 pri tistih kombinacijah vhodnih spremenljivk - mintermov, kjer zavzame izhodna visoko stanje.

## Primer!

V prostoru merimo temperaturo na treh mestih.



$\vartheta_1 \Rightarrow T_1$  (temperaturni senzor 1)  
 $\vartheta_2 \Rightarrow T_2$  (temperaturni senzor 2)  
 $\vartheta_3 \Rightarrow T_3$  (temperaturni senzor 3)

VHODNE SPREMENLJIVKE

Krmilje naj bo izvedeno tako, da v primeru prekoračitve temperature na enem mestu vključi prvi ventilator.

V primeru povečane temperature na dveh mestih, pa vključi še drugi ventilator, tako da delujeta oba hkrati.

V primeru aktiviranja vseh treh senzorjev, pa se poleg obeh ventilatorjev vključi še alarm.

$V_1 \Rightarrow$  (prvi ventilator)  
 $V_2 \Rightarrow$  (drugi ventilator)  
 $A \Rightarrow$  (alarm)

IZHODNE SPREMENLJIVKE

| VHODNE SPREMENLJIVKE<br>(temperaturni senzorji) |                |                | IZHODNE SPREMENLJIVKE<br>(ventilatorja in alarm) |                |   |
|-------------------------------------------------|----------------|----------------|--------------------------------------------------|----------------|---|
| T <sub>1</sub>                                  | T <sub>2</sub> | T <sub>3</sub> | V <sub>1</sub>                                   | V <sub>2</sub> | A |
| 0                                               | 0              | 0              | 0                                                | 0              | 0 |
| 0                                               | 0              | 1              | 1                                                | 0              | 0 |
| 0                                               | 1              | 0              | 1                                                | 0              | 0 |
| 0                                               | 1              | 1              | 1                                                | 1              | 0 |
| 1                                               | 0              | 0              | 1                                                | 0              | 0 |
| 1                                               | 0              | 1              | 1                                                | 1              | 0 |
| 1                                               | 1              | 0              | 1                                                | 1              | 0 |
| 1                                               | 1              | 1              | 1                                                | 1              | 1 |

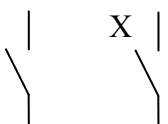
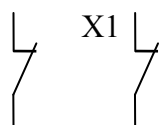
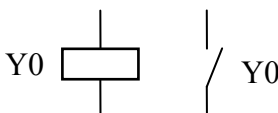
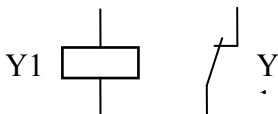
Pravilnostna tabela nam pomaga ne samo pri sintezi krmilja, temveč tudi pri analizi delovanja v primeru okvare.

## Logična funkcija

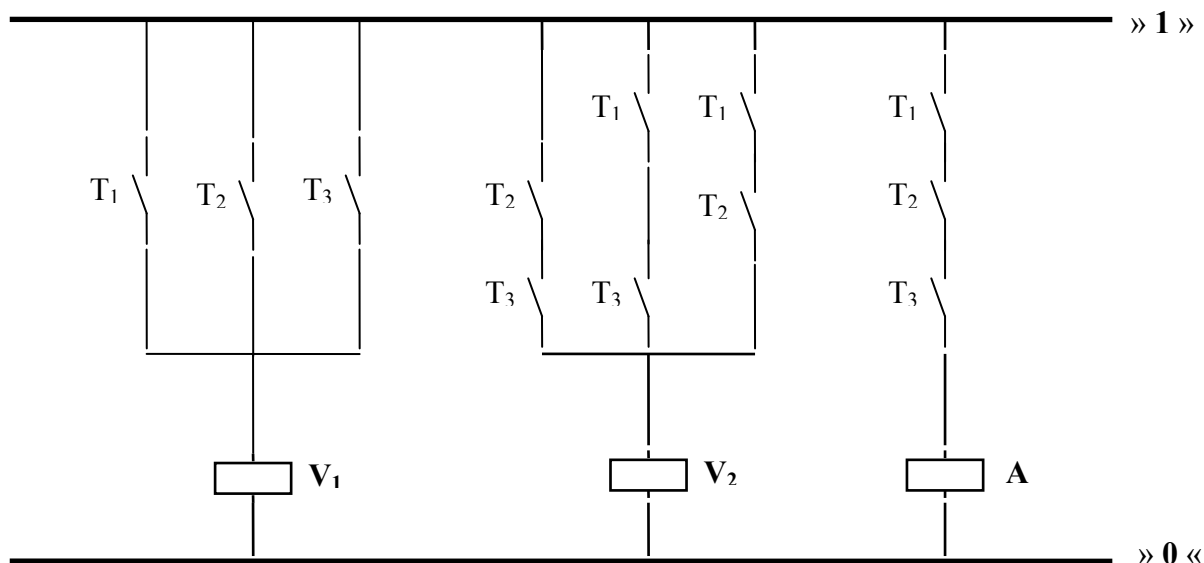
Logična funkcija nam podaja v obliki logične enačbe odvisnost posamezne izhodne spremenljivke glede na stanje vhodnih spremenljivk. V nekaterih primerih predstavlja vhodno spremenljivko tudi stanje kakšnega izhoda (v tem primeru govorimo o koračnih krmiljih). Logično funkcijo sestavimo iz pravilnostne tabele. Funkcijo lahko zapišemo v kanonični disjunktivni ali konjunktivni obliki.

## Krmilni načrt

Krmilni načrt predstavlja realizacijo logične funkcije s simboli elektromehanskih krmilij. V teh načrtih ponazarjamo negirane vhodne spremenljivke z odpiralnimi kontakti, nenegirane spremenljivke pa z zapiralnimi kontakti. Izhodne spremenljivke predstavljamo z releji.

| Pomen                       | Simbol v krmilnem načrtu                                                            | Komentar                                                                                                |
|-----------------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| <b>stikalo oziroma</b>      |  | <b>zapiralni kontakt</b><br>(v osnovnem stanju odprt, delovni kontakti, nenegirana stikala, tipkala...) |
| <b>vhodna spremenljivka</b> |  | <b>odpiralni kontakt</b><br>(v osnovnem stanju sklenjen, mirovni kontakti, negirana stikala tipkala...) |
| <b>navitje releja</b>       |  | <b>aktiviranje navitja vključi kontakt</b><br>(delovni kontakti na izhodu)                              |
| <b>ali izhodna funkcija</b> |  | <b>aktiviranje navitja izključi kontakt</b><br>(mirovni kontakt na izhodu)                              |

Kmilne načrte za funkcije  $V_1$ ,  $V_2$  in  $A$  iz zgleda v prejšnjem poglavju kaže spodnja slika.



Čeprav so moderna krmilja zasnovana ali na trajno ožičenih ali elektronskih vezjih ali kot krmilja s prostim programiranjem, je krmilni načrt še vedno močno v rabi. To si lahko razlagamo zaradi dveh razlogov:

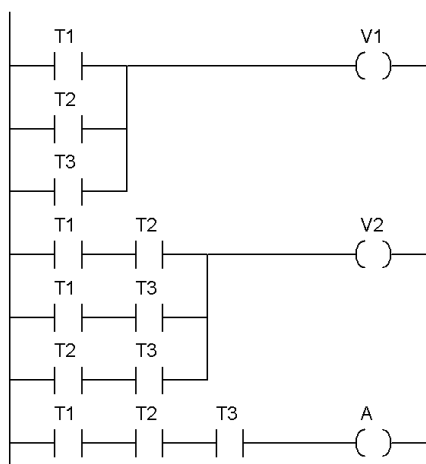
- 1) še vedno obstaja veliko krmilij, v katerih so uporabljeni elektromehanski elementi;
- 2) strokovni kader, ki se ukvarja s krmilji, je deloma še vedno vzgojen na sistemih, ki so bili zasnovani na elektromehanskih krmiljih.

## Kontaktni načrt

Kontaktni načrt je podoben krmilnemu načrtu in ga lahko rišemo ali neposredno iz logične funkcije ali iz krmilnega načrta. V kontaktnem načrtu uporabljamo posebne simbole za odpiralne ali zapiralne kontakte ter za izhodne relejske funkcije. Standardne simbole kontaktnih načrtov in primerjavo s simboli krmilnega načrta kaže spodnja slika.

| Pomen                       | Simbol v krmilnem načrtu | Simbol v kontaktnem načrtu | Komentar                                                                     |
|-----------------------------|--------------------------|----------------------------|------------------------------------------------------------------------------|
| <b>stikalo oziroma</b>      |                          |                            | <b>zapiralni kontakt</b><br>(delovni kontakti, negirana stikala, tipkala...) |
| <b>vhodna spremenljivka</b> |                          |                            | <b>odpiralni kontakt</b><br>(mirovni kontakti, negirana stikala, tipkala...) |
| <b>navitje releja</b>       |                          |                            | <b>aktiviranje navitja vključi kontakt</b><br>(delovni kontakti na izhodu)   |
| <b>ali izhodna funkcija</b> |                          |                            | <b>aktiviranje navitja izključi kontakt</b><br>(mirovni kontakt na izhodu)   |

## Kontaktni načrt krmiljenja ventilatorjev in alarma



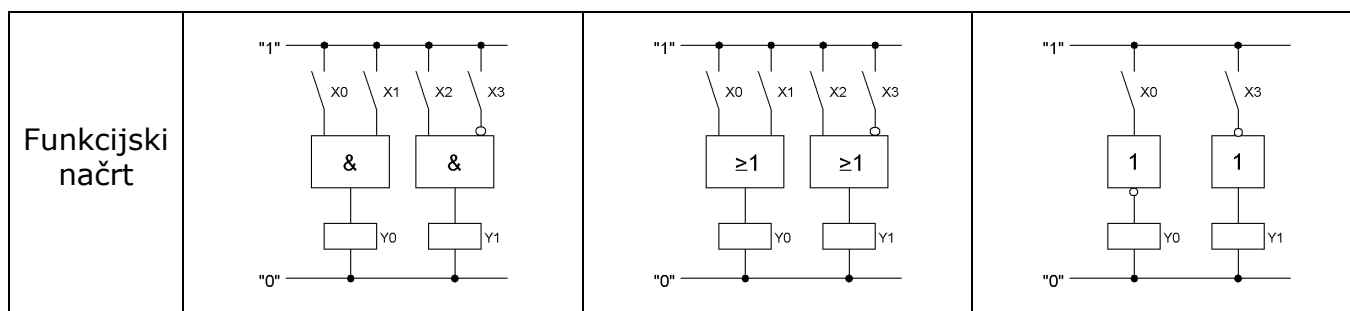
## Funkcijski načrt

Funkcijski načrt je procesno orientiran način prikazovanja krmilne funkcije. Simboli, ki jih uporabljamo v tem načrtu niso odvisni od konkretne izvedbe krmilja, ki je lahko pnevmatsko, elektromehansko, elektronsko ali s prostim programiranjem. Simbole, ki jih uporabljamo za predstavitev posameznih logičnih funkcij, rišemo v skladu z IEC standardom.

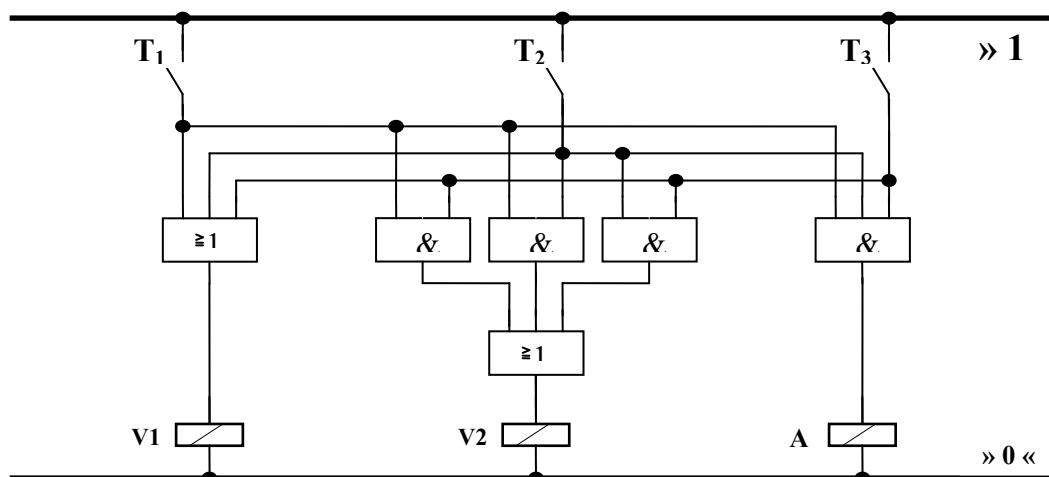
Vhodov ali izhodov simbolov normalno ne rišemo s puščicami, če je smer prehoda signala od leve proti desni ali od zgoraj navzdol. V primeru, ko je v risbi simbol postavljen tako, da prehaja signal od desne proti levi, opremimo vhode in izhode s puščicami, ki enoumno kažejo smer prehoda signala.

Na spodnji sliki imamo podan pregled osnovnih logičnih funkcij v krmilnem, kontaktnem in funkcijskem načrtu.

| Funkcija       | IN | ALI | NEGACIJA |
|----------------|----|-----|----------|
| Krmilni načrt  |    |     |          |
| Kontakti Načrt |    |     |          |



Funksijski načrt krmiljenja ventilatorjev in alarma



## ANALIZA VEZJA

Je ugotavljanje kaj vezje dela. To naredimo tako da:

- napišemo tabelo ( pri kombinacijskih vezjih )
- narišemo časovni diagram ( pri sekvenčnih vezjih )
- iz teh dveh dobimo vhode , izhode in notranja stanja oziroma delovanje vezja

## SINTEZA VEZJA

- imamo besedni opis ( tabele ,diagram )
- moramo narediti vezje ( program )
- analiziramo vezje in ugotavljamo če ustreza zahtevam

## Vprašanja!

1. Na katere načine lahko opišemo neko krmilje?
2. V katerem primeru bomo krmilje opisali s časovnim diagramom?
3. Katere minterme iz pravilnostne tabele vstavimo v logično funkcijo?
4. Kateri načrt predstavlja realizacijo logične funkcije s simboli elektromehanskih krmilij?
5. V krmilnem načrtu ponazarjamo vhodne spremenljivke z \_\_\_\_\_, izhodne pa z \_\_\_\_\_!
6. Nariši simbole za realizacijo IN, ALI in NE funkcije v krmilnem načrtu!
7. Po katerih standardih rišemo simbole v funkcijskem načrtu?
8. Funksijski načrt velja za realizacijo krmilij v naslednjih oblikah?
9. Nariši funkcijske simbole za realizacijo IN, ALI in NE funkcije!
10. Nariši simbol za tipkalo, katero prevaja, ko na njega pritisnemo.