

Kahendkoodidega seotud mõisted

♦ (*n*-järguline) **kahendvektor** on kahendnumbritena **0** ja **1** esitatud loogikaväärtuste ühemõõtmeline jada pikkusega *n*.

Vektori **pikkus** on tema 2ndjärgude arv ehk *n*-järgulise 2ndvektori pikkus on *n*.

☞ näide:

Järgneval real on esitatud 6 erineva pikkusega **kahendvektorit** :

00101101 010 11011 10 1 000101

Kahendvektoril pole mingit seost füüsikast tuntud vektori mõistega.

Erinevalt *kahendarvust* ei tohi kahendvektoris ära jätta algusnulle:

$$000101 \neq 101$$

Kahendvektori järkudel pole *järgukaalu*. Tema sarnasuse tõttu 2ndarvudega osutub mõnes rakenduses siiski kasulikuks ja vajalikuks vaadelda teda *kahendarvuna* ehk 2ndvektori järkudele omistatakse vajadusel 2ndsüsteemi loomulikud järgukaalud :

..... 16 8 4 2 1

See võimaldab kahendvektorit kompaktsemalt esitada talle vastava 2ndarvu väärtuse abil.

♦ **lähisvektorid** (*lähiskoodid*) on võrdse pikkusega kahendvektorid, mis erinevad teineteisest ainult ühes kahendjärgus.

☞ näide: järgnevad 2 vektorit on teineteise *lähisvektorid* : **1011 1001**

☞ näide: ka need kaks 2ndvektorit on teineteise *lähisvektorid* :

1 1 0 1 0 1 1 0 1 0

1 1 0 1 0 0 1 0 1 0

Iga *n*-järguline kahendvektor omab seega *n* tk. lähisvektoreid.

♦ **intervall** on võrdse pikkusega kahendvektorite hulk võimsusega 2^n ($n \in \mathbb{N}$), milles iga hulgaelemendi jaoks leidub samas hulgas täpselt *n* lähisvektorit.

☞ näide:

Järgnev kahendvektorite hulk on *intervall*, kuna ta sisaldab $2^2 = 4$ kahendvektorit ja igaüks nendest omab selles hulgas 2 lähisvektorit :

{ 000 001 010 011 }



?... kontrolli, mitu **lähisvektorit** leidub selles hulgas tema iga **2ndvektori** jaoks ?

Suvaline üksik kahendvektor **{ 00111 }** moodustab samuti *intervalli*, kuna sellises üheelemendilises hulgas on 2^0 elementi ja hulga ainus 2ndvektor omab samas hulgas 0 lähisvektorit:

(seega 2^n tk. 2ndvektoreid moodustavad intervalli ka $n = 0$ korral)



♦ intervalli **olulisteks järkudeks** (olulisteks muutujateks) on tema vektorite need 2ndjärgud, mille väärtus on kõikidel vektoritel kogu intervalli ulatuses konstantne:

{ 000 001 010 011 }

(**mitteolulised järkud** muutuvad kõikvõimalikes kombinatsioonides)

oluliste järkude ja **mitteoluliste järkude ARV** :

Kui intervallis on 2^n *m*-järgulist vektorit, siis on intervallil

(*m* - *n*) **olulist** järku ja *n* **mitteolulist** järku.

☞ näide:

intervallis **{ 0100 0110 }** on 2^1 tk 4-järgulisi vektoreid — misjuhul sellel intervallil on **4 - 1 = 3** olulist järku ja **1** mitteoluline järk **{ 0100 0110 }** siin : kolmas järk on mitteoluline

♦ Intervalli kompaktseks esituseks sobib kasutada **intervalli vektorestitust** sümbolitest **0 1** —, kus intervalli *olulised* (ehk konstantsed) järkud on

tähistatud nendesamade konstantidega **0 1** ja *mitteolulised* järgud on tähistatud sümboliga — .

Üle-eelmise näitena toodud intervalli vektorestitus on **0 — —** :

{ 000 001 010 011 } = 0 — —

Eelmise näiteintervalli vektorestitus on **0 1 — 0** :

{ 0100 0110 } = 0 1 — 0

LOOGIKAELEMENDID — digitaalsignaali töötlevad

Kahendkoode (ehk nende koosseisu kuuluvaid loogikaväärtusi **0 1**) töötlevat elektriskeemi nimetatakse **digitaalskeemiks**.

Iga digitaalseadme elementaarseteks koostisosadeks on **loogikaelemendid**, mis teevad loogikaväärtustega **0** ja **1** lihtsaimaid loogikatehteid — ehk töötlevad "ühtesid" ja "nulle" (uuteks) "ühtedeks" ja "nullideks".

Loogikaelementide omavahelisel kokkuühendamisel saadakse **loogikaskeem / loogikalülitus / digitaalskeem / digitaalseade** .

Iga digitaalseade koosneb seega loogikaelementidest ja ta töötleb **1**-de ja **0**-de kogumeid. Digitaallülituse igal signaaliliinil, sisendil, väljundil (igal "traadil") on igal hetkel olemas kumbki alternatiivne väärtus **0** või **1**, mis välistavad teineteist.

püstitame küsimuse:

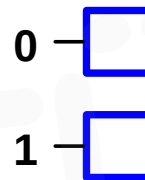


? Millised on **lihtsaimad võimalikud** (elementaarseimad) digitaalsignaali töötlustelemendid (loogikaelemendid), millest saaks koostada suuremaid / keerukamaid digitaalseadmeid ?

Vaatleme esmalt väga lihtsaid "digitaalkomponente" , millel on:

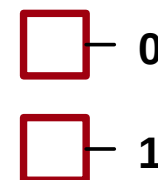
- **1 sisend** ja **mitte ühtegi** väljundit;
- **1 väljund** ja **mitte ühtegi** sisendit;

ainult sisend olemas :



kasutu, kuna "töötulemust" ei väljastu

ainult väljund olemas :

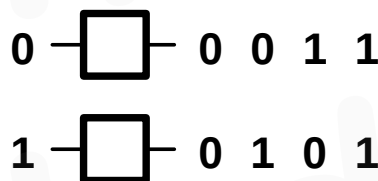


konstandi generaatorid (kuid ikkagi nad ei ole **töötlevad** elemendid)

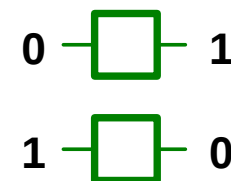
Sellised elemendid ei osutu digitaalsignaali lihtsateks töötlejateks — seega nad ei paku meile huvi, kuna oleme otsimas digitaalväärtusi (**1** ja **0**) **töötlevaid** lihtsaimaid võimalikke komponente/seadmeid.

Läheme sammuvõrra keerulisemaks: vaatleme seadet millel on **1 sisend** ja **1 väljund**.

Võtame seda seadet **2 eksemplari** (vaatleme sama seadme kahte identset koopiat):



4 erinevat võimalikku reaktsiooni



kasulik töötlustulemus

Seega eksisteerib täpselt **4** erinevat võimalikku seadet/elementi, millel on **üks sisend** ja **üks väljund**.

neljast erinevast võimalikust reageerimisviisist osutub **töötlevaks** reageerimiseks ainult üks — **inverteerimine**;

ülejäänud 3 on "sisendsignaali (loogiliselt) mittetöötlevad" reageerimisviisid:

- *konstant 0 generaator*
- *konstant 1 generaator*
- *sisendsignaali taaskordaja ehk repeater*

oleme saanud esimese kasuliku, lihtsaima digitaaltöötlustelemendi: