Pneumatische besturingen

Practicumverslag BARS3a – Besturingstechniek

Naam: Jurre Lievaart
Vak: BARS3a – Besturingstechniek
Opleiding: TIS Delft
Datum: 2/10/2026

1. Inleiding

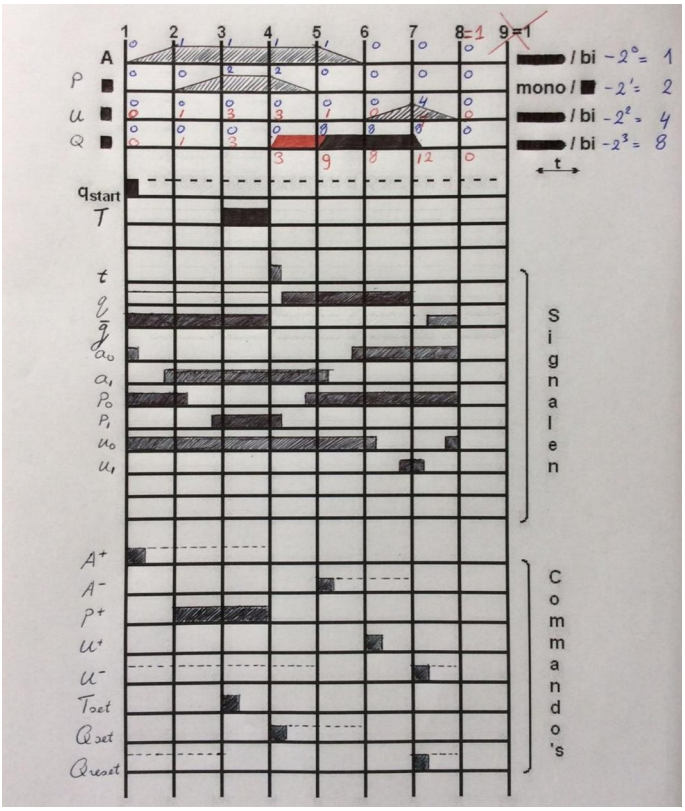
Pneumatische machines in de industrie worden vaak volledig pneumatisch bestuurd: eindstandmelders, logische ventielen en geheugens bepalen samen de volgorde van de cilinderbewegingen. In dit practicum is zo'n besturing stap voor stap ontworpen en gebouwd.

Het verslag behandelt de vier deelopdrachten: besturingsformules afleiden uit een gegeven WST-diagram (1), zelf een WST-diagram en formules opstellen (2), dit ontwerp simuleren in Fluidsim (3) en het bouwen op het Festobord (4). Alle deelopdrachten zijn tijdens de geroosterde practicumsessies beoordeeld:

2. Deelopdracht 1 – Persinstallatie voor lagerbussen

Cilinder A (bistabiel) schuift het product onder perscilinder P (monostabiel) en houdt het vast. P perst 5 s en loopt terug. Daarna gaat A terug en werpt cilinder U (bistabiel) het product uit. Er is een startgeheugen met start- en stopknop.

Controle WST-diagram



Figuur 1 – Gegeven WST-diagram persinstallatie (uit de opdracht)

Het gegeven WST-diagram is gecontroleerd aan de hand van de opdrachtomschrijving. De volgorde A+, P+ (5 s persen), P–, A–, U+ en U– klopt, en ook de eindstandmelders, de timer en het ongelijkmaakgeheugen (gezet door t, gereset door u_1) zijn juist weergegeven. De toestand $a_0 \cdot p_0 \cdot u_0$ komt voor in stap 1 en in stap 6. Die twee worden door q van elkaar onderscheiden, dus één geheugen is genoeg.

Besturingsformules

In de laatste kolom staat de vereenvoudigde vorm.

Stap	Commando	Volledige formule	Vereenvoudigd
1	A+	$q_{start} \cdot a_0 \cdot p_0 \cdot u_0 \cdot \overline{q}$	$\mathbf{q_{start} \cdot a_0 \cdot p_0 \cdot u_0 \cdot \overline{q}}$
2	P+	$a_1 \cdot p_0 \cdot u_0 \cdot \overline{q}$	$\mathbf{a_1 \cdot \overline{q}}$
3	T set	$a_1 \cdot p_1 \cdot u_0 \cdot \overline{q}$	$\mathbf{p_1}$
4	Q set	t	**t**
5	A–	$a_1 \cdot p_0 \cdot u_0 \cdot q$	$\mathbf{p_0 \cdot q}$
6	U+	$a_0 \cdot p_0 \cdot u_0 \cdot q$	$\mathbf{a_0 \cdot q}$
7	Q reset	$a_0 \cdot p_0 \cdot u_1 \cdot q$	$\mathbf{u_1}$
8	U–	$a_0 \cdot p_0 \cdot u_1 \cdot \overline{q}$	$\mathbf{u_1 \cdot \overline{q}}$

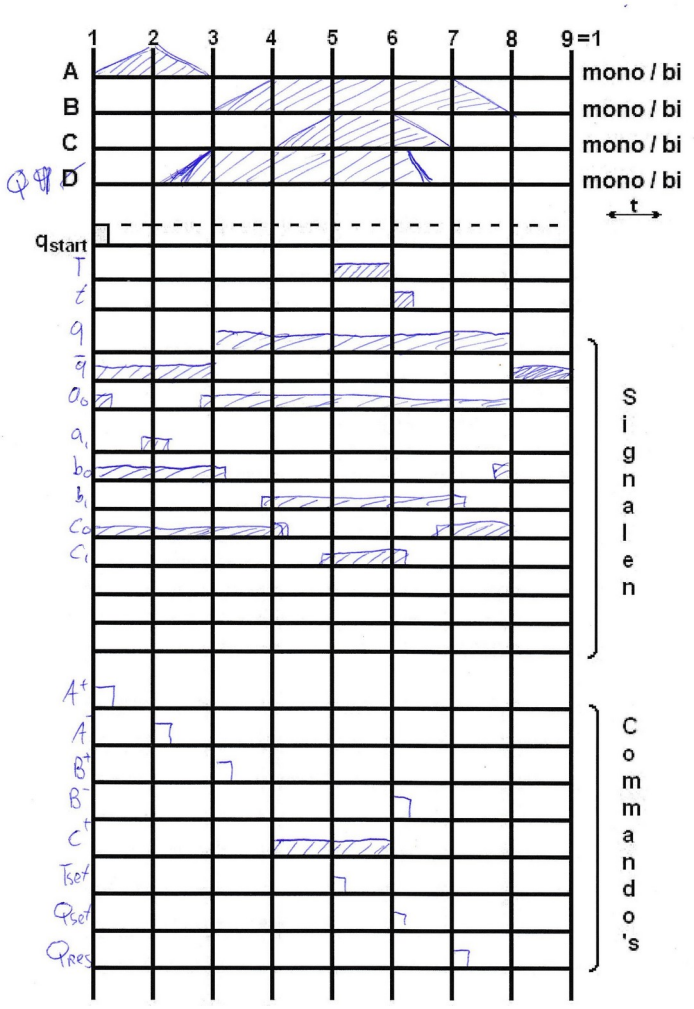
Vereenvoudiging. A+ bevat alle beginstand-elementen, zodat de pers niet start als een cilinder verkeerd staat. P is monostabiel, dus P+ moet blijven staan tot het persen klaar is: $a_1 \cdot \overline{q}$ blijft actief van stap 2 tot Q wordt gezet. Zodra t de q zet, valt P+ weg en loopt P terug. Daardoor valt ook p_1 weg en reset de timer vanzelf. U– wacht op $\overline{q}$, zodat U+ ($a_0 \cdot q$) en U– nooit tegelijk op het bistabiele ventiel staan.

3. Deelopdracht 2 – Overzetstation voor verpakkingen

Cilinder A (bistabiel) schuift de verpakking op de tiltafel en gaat retour. Cilinder B (bistabiel) heft de lift. Cilinder C (monostabiel) kiept de tafel en houdt dit 5 s vast. Daarna gaat eerst C retour en dan B.

WST-diagram

Figuur 2 toont het zelf opgestelde WST-diagram. De cyclus loopt in acht stappen: A schuift uit en terug (stap 1–3), B heft de lift (stap 3–5), C kiept 5 s (stap 4–6), daarna gaat C terug en als laatste B (stap 6–8).



Figuur 2 – WST-diagram overzetstation

Signaal q is het ongelijkmaakgeheugen. Het wordt gezet tijdens A– en gereset door het timersignaal t. Daarmee zijn de stappen met dezelfde eindstanden van elkaar te onderscheiden.

Besturingsformules

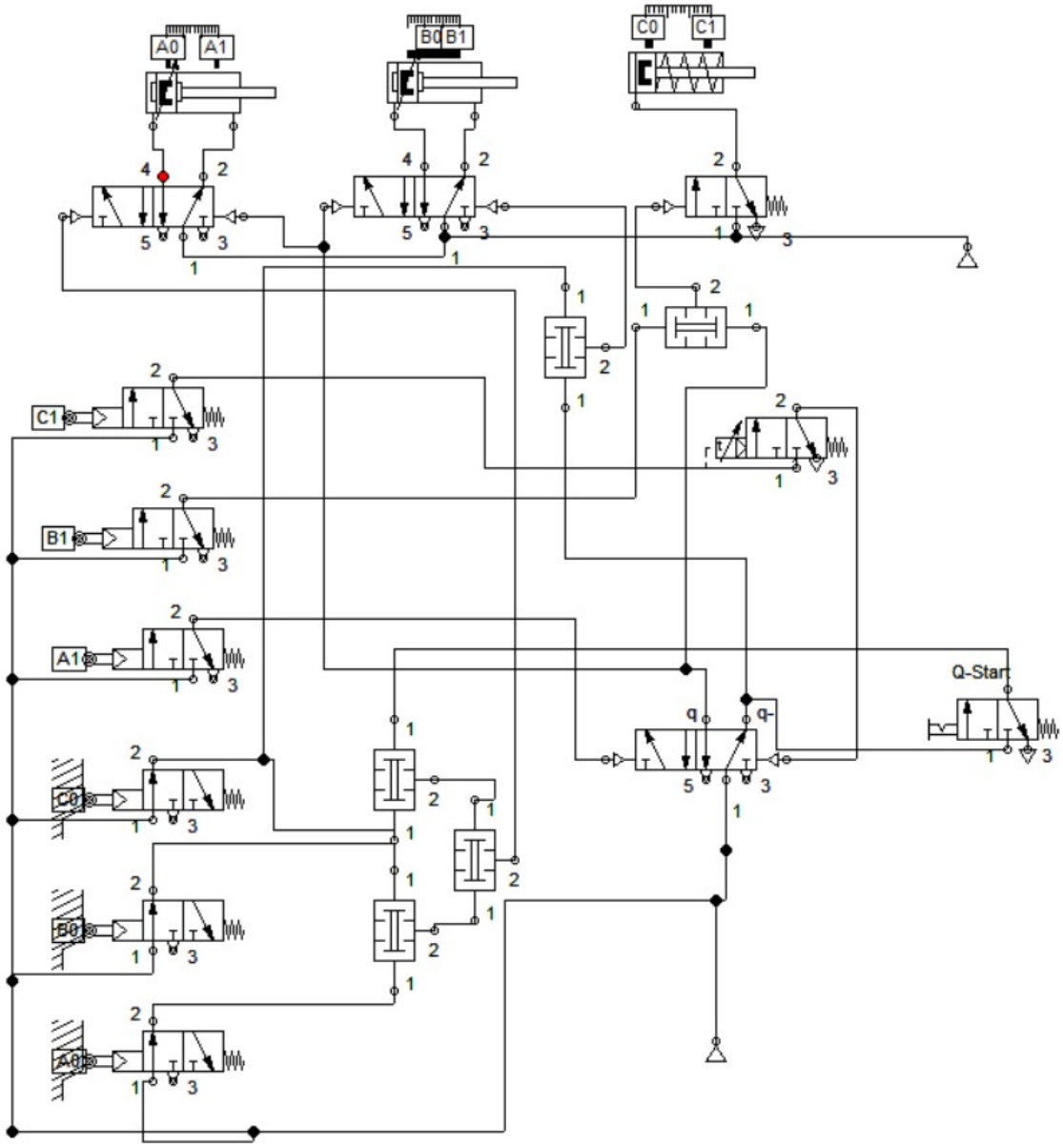
De volledige formules volgen uit het WST-diagram. In de laatste kolom staat de vereenvoudigde vorm die in de schakeling is gebruikt.

Stap	Commando	Volledige formule	Vereenvoudigd
1	A+	$q_{start} \cdot a_0 \cdot b_0 \cdot (\overline{q} \cdot c_0)$	$\mathbf{q_{start} \cdot a_0 \cdot b_0 \cdot \overline{q} \cdot c_0}$
2	Q set, A–	$(\overline{q} \cdot c_0) \cdot a_1 \cdot b_0$	$\mathbf{a_1}$
3	B+	$q \cdot a_0 \cdot b_0 \cdot c_0$	$\mathbf{a_0 \cdot q}$
4	C+	$q \cdot a_0 \cdot b_1 \cdot c_0$	$\mathbf{b_1 \cdot q}$
5	T set	$q \cdot a_0 \cdot b_1 \cdot c_1$	$\mathbf{c_1}$
6	Q reset	t	**t**
7	B–	$a_0 \cdot b_1 \cdot (c_0 \cdot \overline{q})$	$\mathbf{\overline{q} \cdot c_0}$

Vereenvoudiging. Het eerste commando A+ bevat alle beginstand-elementen, zodat de installatie niet start als de cilinders niet in de beginstand staan. Bij de overige commando's is alleen het signaal nodig dat net is veranderd, eventueel aangevuld met q. De combinatie $a_0 \cdot b_0 \cdot c_0$ komt twee keer voor (stap 1 en stap 3); q onderscheidt die twee. Hetzelfde geldt voor $b_1 \cdot c_0$ bij C+ (q) en B– ($\overline{q}$). Zo is één ongelijkmaakgeheugen voldoende.

4. Deelopdracht 3 – Fluidsim-simulatie

De formules uit deelopdracht 2 zijn één op één omgezet naar een Fluidsim-model (figuur 3). Het model is opgebouwd in drie lagen: de sensoren onderaan, de besturingslogica in het midden en de ventielen en cilinders bovenaan. Zo lopen er weinig lijnen kruiselings.



Figuur 3 – Fluidsim-model overzetstation

Gebruikte componenten

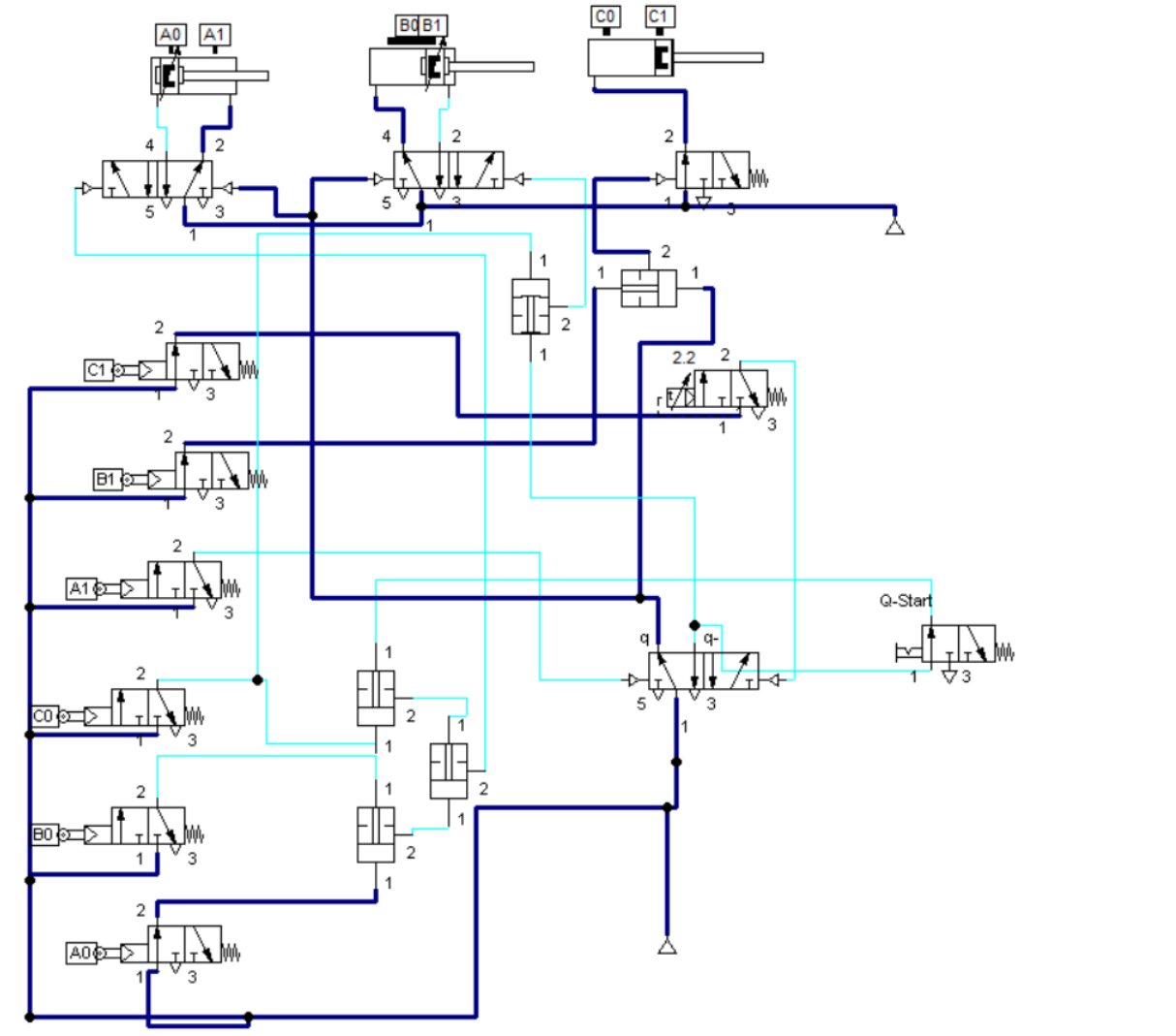
Functie	Component	Aantal
Cilinders A en B	Dubbelwerkende cilinder	2
Cilinder C	Enkelwerkende cilinder met veerretour	1
Sturing A en B (bistabiel)	5/2-impulsventiel, pneumatisch bediend	2
Sturing C (monostabiel)	3/2-ventiel, pneumatisch bediend, veerretour	1

Functie	Component	Aantal
Ongelijkmaakgeheugen q	5/2-impulsventiel (uitgangen q en $\overline{q}$)	1
Timer 5 s	3/2-vertragingsventiel, normaal gesloten	1
EN-functies	Tweedrukventiel	5
Eindstandmelders	3/2-rolventiel	6
Start/stop (q-start)	3/2-handventiel met vergrendeling	1

De term $\overline{q} \cdot c_0$ wordt één keer gemaakt en gebruikt voor zowel A+ als B−. Daardoor zijn zes tweedrukventielen genoeg: één voor $\overline{q} \cdot c_0$, drie voor A+, één voor B+ en één voor C+.

Werking

Met de startknop in stand ‘aan’ en alle cilinders in de beginstand schakelt A+. A schuift de verpakking op de tafel; a_1 zet q en stuurt A terug. Met q en a_0 heft B de lift. Bij b_1 kiept C de tafel. Zodra c_1 actief is start de timer; na 5 s reset t het geheugen. Dan valt C+ weg, gaat C terug en volgt B−. De beginstand is weer bereikt en de cyclus begint opnieuw zolang de startknop aan staat.



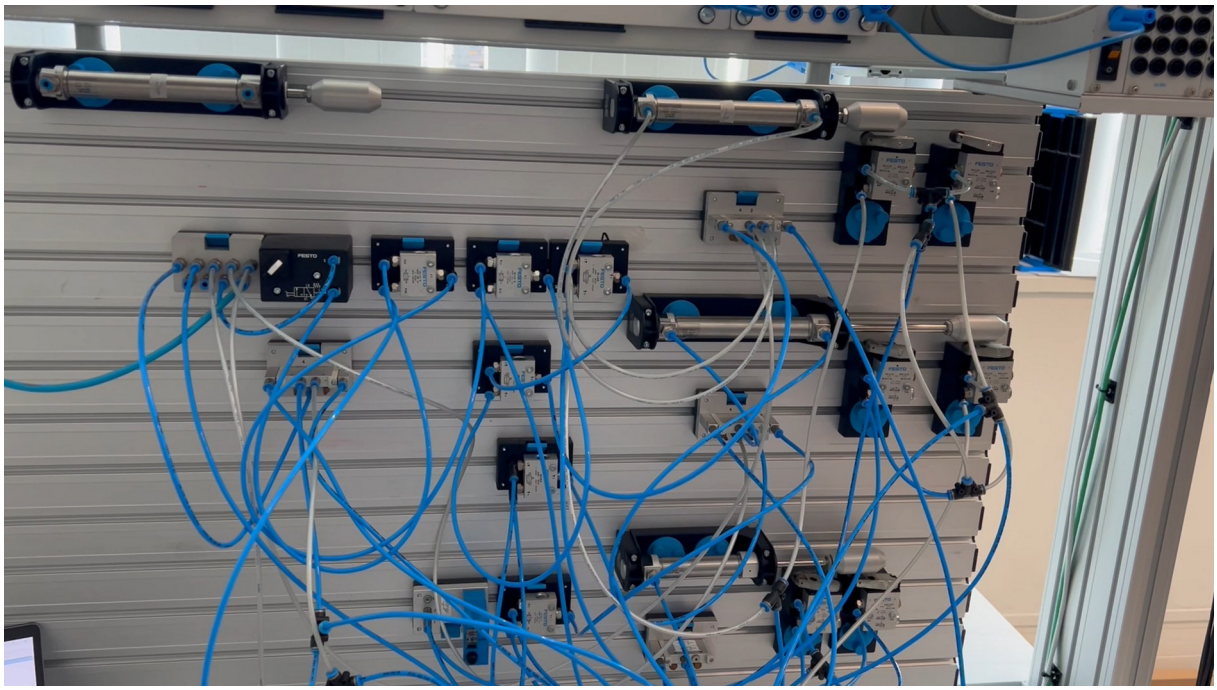
Figuur 4 – Simulatie tijdens de cyclus

Controle eisen

Eis	Hoe opgelost
Ongelijkmaakgeheugen en timer	Eén 5/2-impulsventiel als q, één vertragingsventiel
Stroombegrenzing per cilinder	Meter-out stroombegrenzer per cilinder
Bistabiel op bistabiel	A en B op 5/2-impulsventielen
Monostabiel op monostabiel	C op 5/2-ventiel met veerretour
Geen getrapte sensorventielen	Alle rolventielen en de startknop direct op de hoofdleiding
Eindstandmelders niet direct op actuatoren	Alle signalen lopen via logica naar de stuurventielen
Cyclus herhaalt zolang ‘aan’	q_{start} zit in A+
Cyclus wordt afgemaakt na ‘uit’	q_{start} zit alleen in A+; een lopende cyclus loopt door

5. Deelopdracht 4 – Festobord

Het model uit Deelopdracht 3 heb ik op het Festobord nagebouwd met dezelfde componenten.



Figuur 5 – Opbouw op het Festobord

Test en demonstratie

Eerst is de beginstand gecontroleerd: alle cilinders ingetrokken en de startknop op ‘uit’. Daarna is de startknop aangezet en is de volgorde van de bewegingen vergeleken met het WST-diagram. Tot slot is de startknop midden in een cyclus uitgezet, om te controleren dat de cyclus wordt afgemaakt en de installatie in de beginstand stopt.

Foutzoeken. Ik had een plakkend Q ventiel maar na vervanging en hulp van de docent kon ik de leidingen terug volgen tot het ventiel. En de compressor stond eerst te hoog wat voor rare output zorgde maar door een smoorventiel toe te passen en later de oorzaak te vinden werd dit ook opgelost.

Video. Is samen met het verslag ingeleverd

6. Conclusie

Voor beide installaties zijn met een WST-diagram de besturingsformules opgesteld en vereenvoudigd. In beide gevallen was één ongelijkmaakgeheugen genoeg om stappen met dezelfde eindstanden te onderscheiden. Alleen het eerste commando bevat alle beginstand-elementen; de andere commando's hebben genoeg aan één of twee signalen.

De Fluidsim-simulatie van het overzetstation volgde het WST-diagram en het ontwerp is daarna op het Festobord gebouwd. Door de formules eerst te vereenvoudigen waren weinig componenten nodig, wat het bouwen overzichtelijk hield.