

Afdeling der Elektrotechniek

Aard : Afstudeerverslag
Omvang : 67 bladzijden + 30 bladz. bijlagen
Datum : oktober 1981

Lab/Afd. : Automatische Verkeerssystemen

Opdracht-
nummer : dK/1980/4

Auteur : A.J.M. KLEIN BRETELER

Titel : Het leiden en begeleiden van treinen
over een spoorwegknooppunt.

Opdracht-
nummer : dK/1980/4

Korte
inhoud : Het verslag beschrijft een vervolg onderzoek in het kader van het projekt Blauwkapel. Er is een communicatie tot stand gebracht tussen de regelcomputer en het beheersgebied zodat een NX-CVL-achtig systeem is ontstaan. Treinen worden door de computer gevolgd door het gebied. Als de treinbeweging niet aan de verwachtingen voldoet wordt dit op een beeldscherm gemeld. Hierbij is rekening gehouden met ergonomische eisen. De treinen zijn te volgen via een sporentablo dat wordt aangestuurd door een microprocessor.

EEN ONDERZOEK NAAR PROGRAMMATUUR VOOR

HET LEIDEN EN BEGELEIDEN

VAN TREINEN OVER EEN

SPOORWEG KNOOPPUNT

AFSTUDEERVERSLAG VAN

A.J.M. KLEIN BRETELER

DELFT, 1981

Summary.

The Blauwkapel project is a cooperation between Delft University of Technology and the Netherlands Railway System, with the purpose of developing a general regulatory-program for a railway junction. This report deals with a continued investigation into a program on the regulating processor, and describes the program developed for the input of steering instructions in the computer. These steering instructions can be coded into code-telegrams which are sent to the control area. This communication is compatible with the railway transmission system. In answer to these steering instructions, the control area sends back its state reports to the regulating processor. The program can receive and decode these state reports, after which the processor checks if the actual state of the control area conforms to the instructed one. As a result of a steering instruction, a train can start riding across the control area, and be followed through that area by means of these state reports. Now the actual progress of the train can be compared with the expectations. Irragularities noted here are reported to the operator on a terminal screen. Such presentation takes account of the ergonomic requirements.

A rail tableau is available to present the state of the area to the operator in a convenient way. This report describes the microprocessor program that makes it possible to show the state of the area on the railtableau according to the state reports received.

The program developed now collects the information which can form the basis for the proper regulatory program.

Samenvatting.

Het projekt Blauwkapel is een samenwerking tussen de T.H.-Delft en de Nederlandsche Spoorwegen, met als doel, te komen tot een algemeen bruikbaar regelprogramma voor spoorweg-knooppunten. Dit verslag behandelt een vervolgonderzoek naar programmatuur voor de regelprocessor. Het verslag beschrijft de ontwikkelde mogelijkheden voor het invoeren van stuuroopdrachten in de processor. Deze opdrachten moeten vervolgens van de processor naar het beheersgebied worden overgebracht. Dit gebeurt door de opdrachten te coderen in code-telegrammen. De hiervoor ontwikkelde programmatuur is volledig afgestemd op de spoorweg apparatuur, zodat de regelprocessor direkt aan de bestaande N.S.-transmissiesystemen kan worden aangesloten.

Als antwoord op deze stuuroopdrachten, zendt het beheersgebied toestandsmeldingen, via het transmissiesysteem, terug naar de regelprocessor. De ontwikkelde programmatuur is in staat deze meldingen te ontvangen en te decoderen. Daarna wordt gecontroleerd of de gemelde toestand van de baan overeenkomt met de opgedragen toestand. Als gevolg van een stuuroopdracht, kan er een trein door het gebied gaan rijden. Door middel van de toestandsmeldingen van de baan, is deze trein nu door het gebied te volgen. Daarbij wordt gecontroleerd of de voortgang van de trein volgens verwachting is. De onregelmatigheden die bij de diverse controles worden geconstateerd, worden aan de treindienstleider gemeld via een beeldscherm. Bij de manier waarop deze informatie op het beeldscherm wordt gepresenteerd, is rekening gehouden met de eisen die vanuit het ergonomisch vakgebied hieraan te stellen zijn.

Om de toestand van de baan overzichtelijk aan de treindienstleider te presenteren, is een sporentablo aanwezig. In dit verslag wordt de programmatuur behandeld, die ontwikkelt is om dit sporentablo, met behulp van een microprocessor, aan te sturen overeenkomstig de, van de baan ontvangen, toestandsmeldingen.

Met de nu ontwikkelde programmatuur, worden de gegevens verzameld die, bij verder onderzoek, als basis kunnen dienen voor het ontwikkelen van het eigenlijke regelprogramma.

AFDELING DER ELEKTROTECHNIEK

Vakgroep Automatische Verkeerssystemen

Mekelweg 4 - 2628 CD Delft

AFSTUDEEROPDRACHT

Te verrichten door : A.J.M. KLEIN BRETELER

De opdracht zal worden uitgevoerd in/bij :
Laboratorium voor Automatische Verkeerssystemen

Mentor(en) : ir. M.W.K.A. Breur
ir. R. van Bloemendaal (NS)
ir. L. Zigterman (NS)

Omschrijving van de opdracht :

Het afstudeerwerk maakt deel uit van het projekt "regeling van de treinenloop met behulp van rekenmachines". Binnen dit projekt wordt onderzoek verricht aan de regeling van het spoorwegverkeer op en rond de kruising Blauwkapel. De opdracht omvat een voortgezet onderzoek naar de programmatuur waarmee het proces rond een dergelijke spoorwegkruising bestuurd kan worden. In dit stadium dient de programmatuur ontwikkeld en beproefd te worden waarmee de procesregelcomputer de afwikkeling van het proces kan volgen en kan alarmeren wanneer deze afwikkeling niet volgens plan geschiedt. De mens-machine communicatie dient bij dit onderzoek ruime aandacht te krijgen. Het te besturen proces wordt nagebootst door een tweede minicomputer. De programmatuur mag geen onderscheid maken tussen de afhandeling van het nagebootste proces en de afhandeling van het werkelijke proces. De opdracht omvat tevens het verbeteren van het sporentableau waarmee de werking van de regelprocessor zichtbaar gemaakt kan worden.

Aanvang

afstudeerperiode :
januari 1980

Opdracht-no : dK/1980/4

Delft, januari 1980

De Hoogleraar,

Prof. ir. J.L. de Kroes.

Lijst van afkortingen

Ascii	= American Standard Code for Information Interchange.
CVL	= Centrale Verkeers Leiding.
DOS	= Disk Operating System
DR11-C	= Device Register, interface tussen de PDP11 computer en andere apparaten.
EPROM	= Erasable Programmable Read Only Memory
Led	= Licht Emiterende Diode.
TH	= Technische Hogeschool.
TNI	= TreinNummerIndicator.
NS	= N.V. Nederlandse Spoorwegen.
NX	= eNtrance eXit
PDP 11	= Programmable Digital Processor nr.11, minicomputer.
PIA	= Paralel Interface Adapter.
RT11	= Real Time operating system.
R50	= RADIX 50 notatie

5.5.2	treinnummer informatie	26
5.5.3	lamp/led informatie en knipperen	27
5.5.4	de hoofdlus en de opdrachtverwerking	28
5.5.5	de lammptest	29
5.5.6	uitbreiding van de aansturingstabel	29
5.6	de bediening van het tablo en de microprocessor	29
5.7	een testversie van het tablo programma	30
6	de mens-machine communicatie	31
6.1	verantwoording	31
6.2	de uitwerking van de ergonomische richtlijnen	32
6.3	het uitvoeren van gegevens	33
6.4	de uitvoer	33
6.4.2	de tekstbuffer uitvoer routines	36
6.4.3	de uitvoer van een lineprinter buffer	37
6.5	de keuzemogelijkheden	38
6.6	het uitwissen van regels	38
7	de communicatie met het beheersgebied	40
7.1.1	inleiding	40
7.1.2	het geven van opdrachten	41
7.2	het verzenden van opdrachten	41
7.2.1	het commando	41
7.2.2	de zendprocedure	42
7.3	het coderen van een commandowoord	44
7.3.1	het initieren	44
7.3.2	het coderen van een commandowoord	44
7.4	het ontvangen van een toestandsmelding	45
7.4.1	de signalering	45
7.4.2	de signalerings ontvangst	45
7.5	het verschil tussen de regelprogrammatuur en de CVL apparatuur	47
7.6.1	element aanvragen	48
7.6.2	een wisselaanvraag	48
7.7	het aanvragen van rijwegen	49
7.8	het verwerken van route aanvragen	51
8	het volgen van de treinen	52
8.1	inleiding	52

Voorwoord.

Volgens de normen van de afdeling Elektrotechniek van de T.H.Delft, behoort een afstudeerwerk 6-9 maanden in beslag te nemen. Dat dit voor mij meer dan het dubbele is geworden, komt onder andere doordat er maar weinig rekentijd beschikbaar was en er geprogrammeerd moest worden in assembler taal. Deze wijze van programmeren is zo omslachtig, dat men bij tijd en wijlen het hele doel van het programma uit het oog verliest ! Tevens bleken sommige zaken veel gecompliceerder te liggen dan zich in eerste instantie liet aanzien, bijvoorbeeld de uitvoer van tekst naar het beeldscherm. Voeg hierbij de nodige hardware problemen en op die manier loopt het afstudeerwerk zeer ver uit op het tijdschema.

Toch ligt dan nu uiteindelijk het afstudeer verslag voor u. Het behandelt een vervolgonderzoek naar programmatuur voor het regelen van treinverkeer op een spoorwegknooppunt. De ontwikkelde programmatuur moet de basis vormen waarop een volgend afstudeerder het onderzoek naar het eigenlijke regelprogramma kan baseren. Een aantal mensen wil ik danken voor hun bijdrage tot het uiteindelijke resultaat:

Professor de Kroes en de mentoren Zigterman, v.Bloemendaal en Breur, die mij tijdens de regelmatig gehouden tussenbesprekingen bijstuurden als ik met mijn onderzoek op een dwaalspoor dreigden te raken.

Het personeel van de vakgroep AVS en met name de heer Stikkel, die altijd klaar stonden als er problemen waren.

Dick Schriek die afstudeerde op het nabootsingspakket

Tot slot wens ik de volgende afstudeerders aan dit projekt, een succesvolle afstudeerperiode toe. Ik hoop dat zij aan dit verslag voldoende steun hebben om het vervolgonderzoek succesvol en in een redelijke tijd af te ronden.

Voorburg, 11-sept-1981, Loek Klein Breteler.

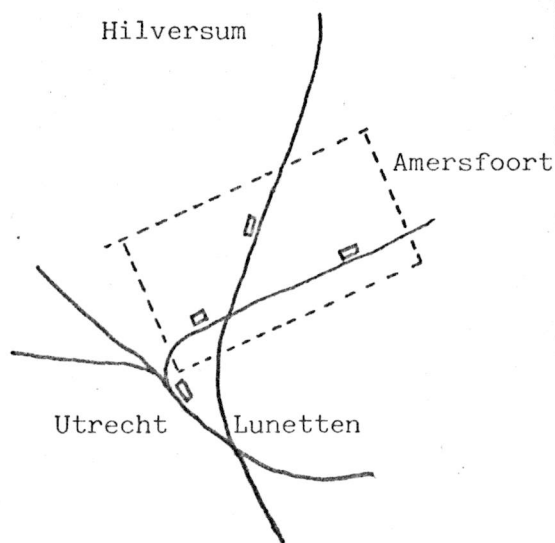
HOOFDSTUK 1: INLEIDING

1.1 Verantwoording

De Vakgroep Automatische VerkeersSystemen van de T.H. in Delft onderzoekt de mogelijkheden voor het automatiseren van verkeerssystemen. Zo is men sinds 1976 bezig met het ontwikkelen van een regelsysteem voor het treinverkeer. Met adviezen van de Nederlandsche Spoorwegen werkt de vakgroep aan het ontwikkelen van het prototype van een verkeersregelprogramma en de benodigde testprogramma's.

Als studie object is gekozen voor het kruispunt Blauwkapel. Dit is gelegen ten noorden van Utrecht en vormt de kruising van de lijnen Utrecht-Amersfoort en Lunetten-Hilversum.

Zie figuur 1.



Figuur 1: Het bestudeerde gebied.

Hoofdstuk 2 De probleemstelling.

2.1 inleiding

Het uiteindelijke doel van het projekt Blauwkapel is, te komen tot een algemeen programma, voor een minicomputer, dat in staat is om zelfstandig het verkeer op een spoorwegknooppunt optimaal te regelen. Of dit regelen ook werkelijk zelfstandig door de computer mag worden uitgevoerd, is een politieke keuze. Voorlopig ligt de eindverantwoording voor elke individuele regelaktie bij de treindienstleider. Deze moet voor elke regelaktie toestemming geven, waarna de computer de regelaktie kan uitvoeren.

Hierbij is het de taak van de computer om de mogelijke regelakties te onderzoeken op hun doelmatigheid. De uitkomsten van dit onderzoek worden, met de voor- en nadelen, gepresenteerd aan de treindienstleider, die dan een keuze maakt. De administratieve werkzaamheden van de treindienstleider kunnen ook door de computer worden uitgevoerd, zoals het bijhouden van een logboek.

Wil men de computer het verkeer zelfstandig laten regelen of mogelijke regelakties laten berekenen, dan zijn de volgende gegevens nodig:

- a) gegevens van de baan: Waar bevinden zich welke wissels, seinen e.d. in het gebied en wat is hun stand? Welke secties zijn er en zijn deze bezet, vrij of gereserveerd?
- b) gegevens van de treinen: Welke treinen zijn er gedefinieerd en waar bevinden deze treinen zich? Wat is hun route door het beheersgebied? Op welk tijdstip horen ze het beheersgebied te verlaten?
- c) Criteria voor het regelen van het verkeer.

Dient de treindienstleider een keuze te maken uit een aantal gepresenteerde regelakties, dan komen er nog enkele eisen bij:

- d) Een duidelijke presentatie van de gegevens aan de treindienstleider.
- e) Bedieningsorganen voor de computer.

Deze punten zijn in de volgende paragrafen nader toegelicht.

2.4 Het regelen van de treinenloop

Met de verzamelde gegevens van de treinen, de baanelementen en de tijden is genoeg bekend om te kunnen bepalen welke treinen voorrang verdienen op de kruising. Voorbeeld:

- a) Een posttrein zal moeten wachten op een sneltrein.
- b) Een beladen goederentrein zal, uit energetisch oogpunt, voorrang krijgen op een andere trein.
- c) Een trein die voor ligt op zijn dienstregeling zal moeten wachten op een andere trein.

Al dit soort situaties zal moeten worden gevangen in een regel-algoritme. Steeds als een trein het knooppunt nadert, zal moeten worden berekend wat de prioriteit van deze trein is. De trein met de hoogste prioriteit zal steeds als eerste toegang tot het knooppunt moeten krijgen. Dit alles slechts als voorbeeld.

2.5 De presentatie van de gegevens aan de treindienstleider

Om de treindienstleider een overzichtelijk beeld te geven van de situatie, is het gebruikelijk om het gehele beheersgebied schematisch weer te geven door middel van een sporentablo. Hierop worden alle gegevens met behulp van lampjes zichtbaar gemaakt. De gegevens voor het sporentablo worden afgeleid uit de ontvangen signaleringen en verzonden opdrachten.

Aanvullende informatie kan worden verstrekt via een beeldscherm of een printer van de computer.

2.6 Schematische weergave van het systeem

De verschillende stadia waarin het project kan worden opgedeeld, zijn grafisch weergegeven in figuur 2.1.

Hoofdstuk 3: Het al gerealiseerde deel van het projekt.

Het spoorwegproces is opgebouwd te denken uit drie delen:

1. Het beheersgebied. Dit omvat de beveiligings-apparatuur en de baan-elementen, zoals seinen, wissels en secties.
2. Het regelcentrum. Van hieruit regelt de treindienstleider de treinenloop door het beheersgebied.
3. Het communicatie systeem tussen het beheersgebied en het regelcentrum. Dit is het CVL-Quicktrol systeem.

Voor het regelen van het treinverkeer zal gebruik gemaakt worden van een PDP 11-20 minicomputer.

In 1976 is L. Zigterman het onderzoek begonnen met het verzamelen van gegevens over de geografie van het gebied (zie Lit. 1). Hij heeft ook datastructuren ontwikkeld die deze gegevens makkelijk toegankelijk maken voor de computer. Alle verzamelde gegevens zijn samengebracht in drie ascii-definitietabellen die met de hand worden opgesteld (zie ook Lit. 4,H2). Deze tabellen bevatten de gegevens van de rijwegen en de gegevens van de code-telegrammen, die in twee richtingen via het communicatiesysteem te verzenden zijn. Uit deze definitietabellen worden via conversieprogramma's, off-line, de voor de computer bruikbare datatabellen afgeleid. Hierdoor werken de programma's onafhankelijk van het gedefinieerde gebied. Door het wijzigen van de definitie-tabellen is het programma ook te gebruiken voor andere knooppunten.

Op de band opgenomen code-telegrammen (de registraties), waren met een hiervoor ontwikkeld hoofdprogramma te decoderen en te verwerken in de tabellen. L. Zigterman heeft ook het sporentablo ontworpen. Dit tablo wordt aangestuurd met behulp van een microprocessor die net als de regelprocessor, werkt met datatabellen die zijn afgeleid van de drie definitietabellen. De gegevens van de registraties konden worden zichtbaar gemaakt op het sporentablo.

In 1978 heeft R.v.Looveren het onderzoek voortgezet (zie lit. 2). Hij heeft vooral onderzoek gedaan naar de mens-machine-communicatie en in het bijzonder de invoer van opdrachten in de regelcomputer.

Hoofdstuk 4: Inleiding op de te verrichten werkzaamheden.

4.1 Inleiding op het verrichte onderzoek.

Uitgaande van het in hoofdstuk 3 beschreven uitgangspunt, is de volgende stap in het projekt, de verwerking van toestandsmeldingen, die door het programma zelf zijn ontvangen van het beheersgebied. Tot hier toe werd steeds met, op de band opgenomen, registraties gewerkt. Om het spoorwegproces te kunnen beïnvloeden, dienen de ingetypte stuuroopdrachten naar de baan te worden verzonden. Dit alles vereist programmatuur die volledig is aangepast aan het N.S.-transmissie systeem. Daartoe ook, moeten de ingetypte stuuroopdrachten gecodeerd worden in code-telegrammen, de commando's. Deze commando's moeten aan het N.S.-transmissie systeem worden aangeboden.

Van de ontvangen toestandsmeldingen, de signaleringen, kan dan gecontroleerd worden of ze aan de verwachtingen voldoen. Indien, als gevolg van een commando, een trein door het gebied gaat rijden, kan gecontroleerd worden of deze treimbeweging volgens verwachting verloopt. Om de onregelmatigheden in het spoorwegproces te kunnen melden, tegelijk met het intypen van opdrachten, dienen eisen aan de uitvoer te worden gesteld.

Om het sporentablo te kunnen gebruiken diende het tablo-aansturingsprogramma te worden aangepast aan de nieuwe hardware.

Daarmee is het projekt dan gevorderd als aangegeven in figuur 4.1.

De eisen die aan deze onderwerpen zijn te stellen, zijn in de volgende paragrafen verder uitgewerkt. De behandeling van de onderwerpen zelf is te vinden in de daarop volgende hoofdstukken.

4.2 De aanpassing van de tabloprogrammatuur.

De eisen die aan de nieuwe tabloprogrammatuur gesteld worden, zijn dezelfde als die welke golden voor de oude programmatuur:

4.3 De in- en uitvoer van gegevens.

De invoer van gegevens gebeurt, in dit prototype van een regelprocessor, door middel van een gewoon toetsenbord. De treindienstleider moet de tekens die hij intypt op het beeldscherm zien verschijnen. Daar er tijdens het intypen van een opdracht, uitvoer op het beeldscherm kan worden geschreven, dient hiertussen een scheiding te worden aangebracht, om te voorkomen dat de ingetypte tekens tussen de tekst komen te staan. De te gebruiken lettertekens voor het invoeren van opdrachten, dienen ook aan ergonomische eisen te voldoen.

Ook de presentatie van gegevens op het beeldscherm dient te voldoen aan de ergonomische eisen die hieraan te stellen zijn.

Daar de treindienstleiders, die gewend zijn met een NX-CVL toetsenbord bediening te werken, de beschikking hebben over een tussenbuffer om, alvast ingetypte opdrachten, te bewaren, dient deze faciliteit ook op het beeldscherm aanwezig te zijn.

Wijzigingen in de manier van werken met de programmatuur, dienen via het toetsenbord te kunnen worden aangebracht.

4.4 De communicatie met het beheersgebied.

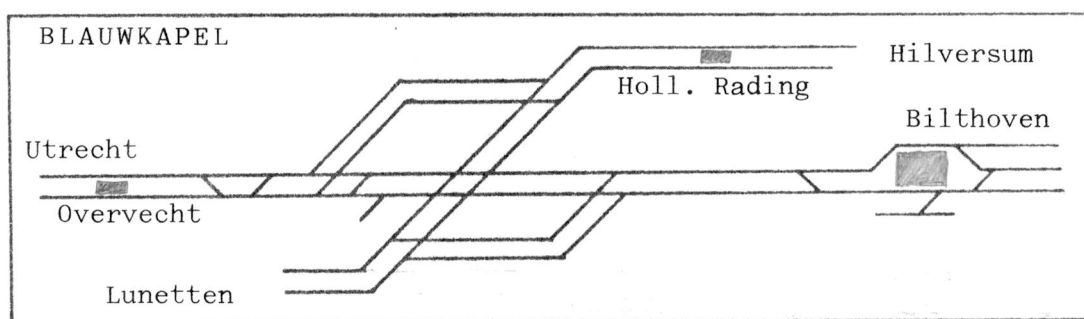
De communicatie met het beheersgebied bestaat uit het verzenden van stuuropdrachten, de commando's, en het ontvangen van toestandsmeldingen, de signaleringen. Deze communicatie dient geheel te zijn aangepast aan het N.S.-transmissiesysteem. Dit geldt zowel voor de manier van communiceren als voor de inhoud van de verzonden boodschappen. De ingetypte opdrachten dienen dus te worden gecodeerd tot commando's met de door de N.S. gebruikte codering.

Tevens dient de programmatuur te werken volgens het NX-CVL principe. Dit houdt in dat de trein door het beheersgebied kan worden geleid,

HOOFDSTUK 5: Het sporentablo.

5.1.1 Inleiding

Als hulpmiddel voor de treindienstleider is er een sporentablo aanwezig. Dit is een grafische weergave van het beheersgebied, het spoorweg-emplacement, waarop door middel van lampjes wordt aangegeven waar zich treinen bevinden. Ook de stand van wissels en seinen is zichtbaar. Zie figuur 5.1.



figuur 5.1: het sporentablo.

De treindienstleider kan nu, zittend voor het tablo, voor zich zien hoe de situatie buiten op de baan is. Aan de hand van deze gegevens kan hij nu de treinen door het beheersgebied leiden. Op het sporentablo kunnen de volgende zaken zichtbaar gemaakt worden:

- een wisselstand: Normal of Reverse, door middel van een witte lamp.
- een wisselvergrendeling: met een groene led.
- een seinstand: stopstand met een rode led, rijstand met een gele led.
- een sectiereservering: groen branden van de sectie.
- een sectiebezetting: geel branden van de sectie.

Ook complete rijwegen kunnen zichtbaar gemaakt worden;

Een rijweg is een aaneengesloten rij secties, wissels en grensels, die begint en eindigt met een sein.

5.2. De hardware van het tablo.

Voor een nauwkeurige beschrijving van de hardware wordt verwezen naar het verslag van Hans Vink (lit: 5) en bijlagen E1. Voor de aansturing zijn de volgende gegevens van de hardware van belang:

Er is voor de aansturing van het tablo een 16 bits interface beschikbaar. 8 Bits hiervan, adres genoemd, selekteren een groep lampjes/leds of een halve treinnummerindicator. De overige 8 bits, data genoemd, selekteren een lampje of led uit die groep of vormen de inhoud van de halve treinnummerindicator in BCD-code. Alle elementen zijn voorzien van een geheugen-element dat de informatie bewaart.

Verder is er nog een kloklijn die het moment bepaalt waarop een geselecteerd geheugenelement informatie over moet nemen van het interface.

Het tablo wordt bestuurd door een M6800 microprocessor.

In navolging van de methode die is gebruikt bij het eerste ontwerp van het tablo en de programmatuur daarvan, is de bedrading van het tablo weer volkomen willekeurig aangebracht. De bedoeling is dat alle lampjes, leds e.d. gewoon worden aangesloten, elk op een eigen adres, en dat daarna pas gekeken wordt op welk adres alles terecht is gekomen. Dit laatste gebeurt met een speciaal programma (zie H5.3).

5.3. Inventarisatie van de elementen en de bedrading.

Daar de bedrading van alle tablo-elementen volkomen willekeurig is, moet eerst worden uitgezocht op welk adres alle elementen terecht zijn gekomen. Hiervoor is het programma PONS68 geschreven dat draait op de M6800 microprocessor, die het tablo moet gaan aansturen.

Dit programma stuurt, in volgorde, een voor een alle adressen aan en vraagt de naam van het betrokken element in te typen, gevolgd door een soort aanduiding en de kleur van het element. Voor

is dan ook verdubbeld.

Daar sommige secties nu in 2 delen worden aangestuurd, worden er 2 sectie-tabellen opgebouwd. In de eerste tabel staan alle eerste delen van de secties en in de tweede tabel staat de extra informatie (indien aanwezig). Voor een sectie moeten nu dus steeds twee tabellen worden geraadpleegd.

Mochten er nog eens secties komen die in meer dan twee delen moeten worden aangestuurd, dan bespaart het geheugenruimte als men overgaat op een heel ander type tabel.

Aanbevolen wordt, om dan een tabelstructuur te gebruiken zoals die van de rijweggegevens. Dus een adrestabel met de adressen van de betrokken gegevens en 2 datatabellen met alle gegevens. Een datatabel met de gegevens van de gele lampen en een van de groene lampen. In deze datatabellen staan alle gegevens van een element achter elkaar, met 00 achter de gegevens als afsluiting. Hiermee is het aantal aansturingsadressen van een sectie vrijwel onbeperkt. Alle programma's moeten echter wel veranderd worden om met deze nieuwe tabellen te kunnen werken.

Bij dit alles gaat het steeds om informatie in ascii-code. Om deze informatie voor de M6800 microprocessor toegankelijk te maken, moeten de tabellen eerst nog vertaald worden door de assembler.

5.5.3 Lamp/led informatie en het knipperen

De lampjes en leds moeten ook kunnen knipperen en de informatie knipperen/vast branden/doven moet dus ook ergens worden bewaard. Dit is gedaan door een nieuwe tabel op te bouwen, de aansturings-tabel. Hier staat alle informatie voor de lampjes en de leds met het tablo-adres erbij, in een vorm die geschikt is om direkt naar het tablo te verzenden. De structuur van deze tabel is als volgt:

tabloadres, periode 1 databyte, periode 2 databyte, tabloadres, periode 1 databyte, periode 2 databyte, enzovoort.
--

De routines die deze tabel opbouwen vinden deze tabloadressen door alle element-tabellen af te zoeken die zijn opgebouwd door het conversieprogramma CONDAT. In deze element-tabellen staan alle adressen die in gebruik zijn voor het aansturen van lampjes en leds. Deze adressen worden in de aansturings-tabel gezet. In de element-tabellen komt nu als adres het low-byte van het adres van de plaats in de aansturings-tabel te staan, dat het betrokken element vertegenwoordigt.

De weg die gevolgd moet worden om van het opgegeven elementnummer het tablo aan te sturen, is nu als volgt:

Het nummer van het element verwijst naar een plaats in de element-tabel. Deze plaats in de element-tabel bevat het databit van het element en verwijst naar de plaats in de aansturings-tabel waar dit databit ingevuld moet worden om het element te laten branden. Nadat er een nieuw databit is ingevuld in de aansturings-tabel wordt de hele tabel naar het tablo verzonden.

Achter het tabloadres in de aansturings-tabel zijn twee bytes gereserveerd voor het invullen van de databits. Dit biedt de mogelijkheid om een element te laten knipperen. Wanneer namelijk in het periode 1-byte het databit van een element wordt ingevuld en in het periode 2 byte niet, dan heeft het afwisselend verzenden

5.5.5 De lampentest

Er zijn 4 lamptest routines opgenomen in het programma TABLO4.M68. Een test voor alle gele lampen, een voor alle groene lampen, een voor alle witte wissellampen, de seinleds en de vergrendelleds en een test voor alle treinnummerindikatoren. Deze testroutines zijn aan te roepen via de regelprocessor door het verzenden van een speciaal routinenummer. Vanaf de regelprocessor kan het tablo ook gewist worden. Dit kan door een routinenummer voor het wissen van het tablo.

Na het wissen van het tablo worden ook de aansturings-tabel en de TNI-tabel schoongemaakt en de interfaces opnieuw geïntialiseerd.

5.5.6 Uitbreiding van de aansturings-tabel

Daar er nu drie bytes per tabloadres nodig zijn, kan het gebeuren dat bij een grote uitbreiding van het tablo de aansturings-tabel te groot wordt om in een byte te adresseren. De tabel neemt nu 144 bytes in beslag van de maximale 256 bytes. Als oplossing voor dit soort ruimtegebrek kan men de tabel opdelen in drie tabellen. Een tabloadres-tabel, een periode 1-tabel en een periode 2-tabel. Om de drie gegevens van een element te bereiken heeft men dan steeds hetzelfde low-adres nodig (1 byte), maar voor elke tabel een ander high-adres. Op deze manier kan men met slechts geringe programma aanpassingen drie maal zoveel adressen aansturen.

5.6 De bediening van het tablo en de microprocessor

Het opstarten van de microprocessor en het tablo vereiste nogal wat handelingen. Daar dit allemaal standaard procedures zijn, kunnen deze allemaal door de microprocessor zelf uitgevoerd worden. Het monitorprogramma van de microprocessor is nu zo aangepast dat na het inschakelen van de spanning, de micropro-

Hoofdstuk 6 De Mens-Machine communicatie.

6.1 Verantwoording.

Onder de mens-machine communicatie wordt over het algemeen verstaan, de dialoog van een mens met een computer. Voor een goede communicatie zijn twee zaken nodig, te weten:

1. de technische middelen om te kunnen communiceren.
2. de ergonomische eisen die te stellen zijn om de communicatie mensvriendelijk te houden.

Punt 2 is, volgens mij, zo'n specialistisch onderwerp dat dit door een ergonoom moet worden uitgewerkt en niet door iemand met een elektrotechnische opleiding.

Punt 1 houdt in, het ontwikkelen van de gereedschappen die de ergonoom nodig heeft om zijn werk te kunnen doen. Dit is werk dat door iemand met een elektrotechnische achtergrond kan worden gedaan. Om dit te doen is wel enig inzicht nodig in de ergonomie. Men moet namelijk weten wat voor eisen een ergonoom zoal aan de communicatie kan stellen. In de ergonomische literatuur (lit 8: H5) komen de volgende punten naar voren:

- a) informatie van zeer verschillend gewicht, dient in verschillende deelruimten te worden gepresenteerd.
- b) de informatie dient overzichtelijk geplaatst te zijn.
- c) de gemiddelde zoektijd voor een tekst is kleiner bij een gestructureerde presentatie.
- d) verwerkte informatie moet verwijderd kunnen worden.

Als invoer orgaan wordt een toetsenbord gebruikt. Voor de invoer hiermee is uitgegaan van de syntax zoals die is ontworpen door R.v.Looveren, die hier enig onderzoek naar heeft gedaan (lit 2: H11).

Voor nadere bijzonderheden, zie bijlage F.

Opgemerkt dient nog te worden dat als het systeem in een echte CVL-post zal worden ingezet, er niet met een beeldschermterminal zal worden gewerkt voor de normale bedieningshandelingen van de treindienstleider. Het beeldscherm zal dan dienen voor het geven van aanvullende informatie, terwijl de eigenlijke CVL bedieningsfuncties op een speciaal bedieningsorgaan zijn ondergebracht. Het

voegd. Dit gebeurt met merktekentjes voor de regels. Als alle tekst bekeken is kunnen deze tekentjes worden verwijderd. Punt d leidt tot de mogelijkheid om regels te kunnen wissen van het beeldscherm. Om een bepaalde regel te kunnen selekteren, zijn aan alle regels regelnummers toegevoegd.

6.3 **Het invoeren van gegevens.**

De invoer is toegespitst op het uitvoeren van een aantal vooraf geprogrammeerde opdrachten. Deze opdrachten worden direkt uitgevoerd zodra het daarmee corresponderende teken door de computer is herkend. Dit teken wordt onderzocht in de interruptroutine die de tekens, die op het toetsenbord worden ingetypt, invoert in de computer. Is een van de standaard tekens herkend dat wordt een vlag gezet. Vanuit de hoofdloop wordt steeds in de routine SYNTAX gekeken of deze vlag opstaat. Is door middel van deze vlag aangegeven dat er een opdracht is ingetypt, dan wordt via een sprongtabel naar de betreffende verwerkingsroutine gesprongen. Deze routine controleert de syntax van de opdracht en voert de opdracht uit.

Sommige opdrachten resulteren in een vraag en antwoordcyclus. De opdrachtcode leidt dan tot het stellen van een vraag door de computer. Het ingetypte antwoord wordt dan verwerkt na het intypen van de RETURN-toets.

Tijdens het intypen worden de ingetypte tekens geechoed op de onderste regel van het beeldscherm en eventueel ook op de lineprinter. Als een opdracht is uitgevoerd, wordt deze weer van het scherm verwijderd. Als de computer een vraag gesteld heeft, wordt deze van het scherm verwijderd zodra deze beantwoord is.

6.4 **De uitvoer.**

Voor het bewaren van de teksten die in de diverse deelruimten van het beeldscherm staan, is een aantal tekstbuffers beschikbaar. Deze zijn zo groot, dat precies alle regels van een deelruimte in het

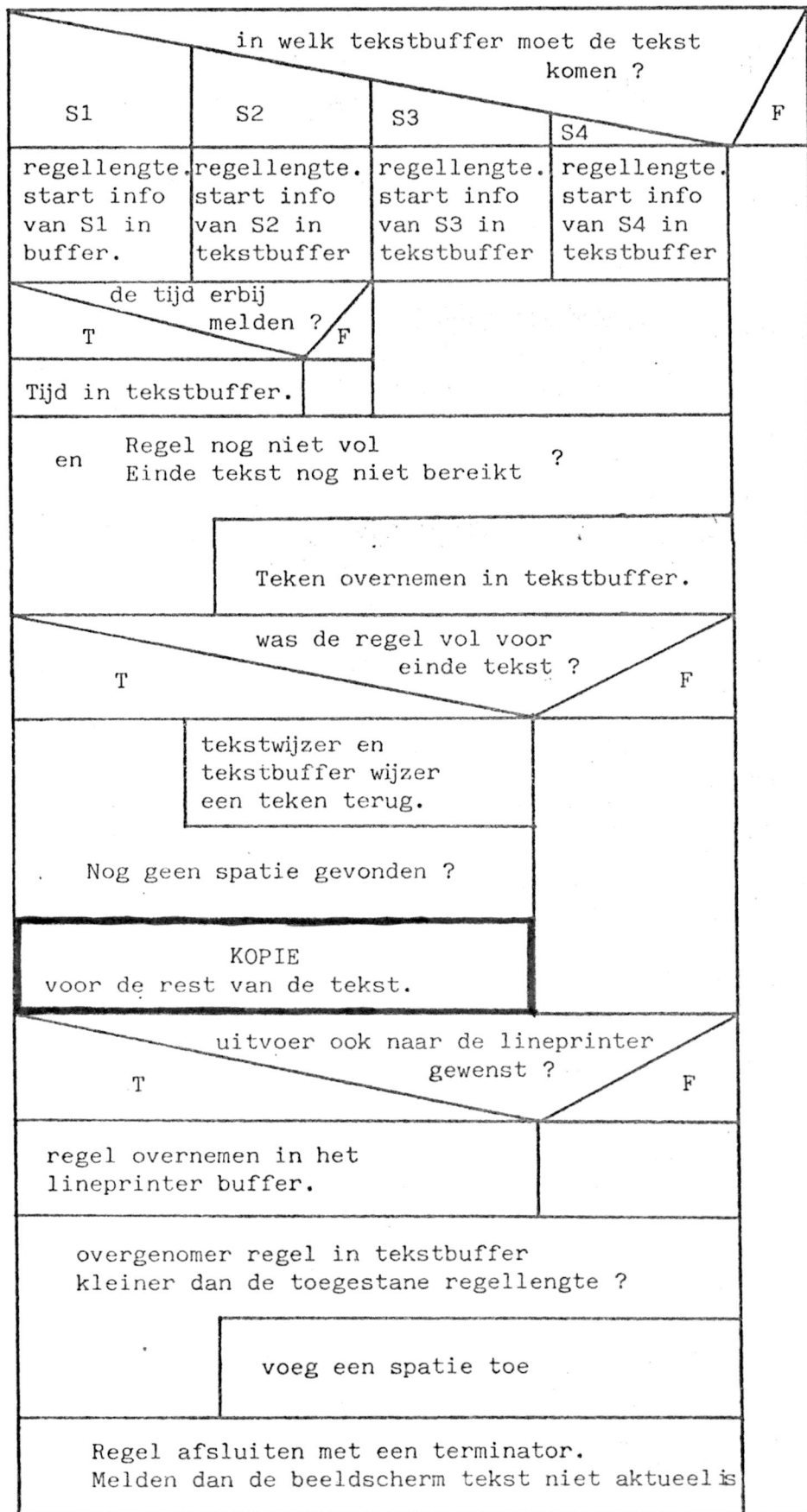


Diagram 6.1: De routine KOPIE

In de deelruimten van het beeldscherm staat de laatst ingevoerde tekst steeds onderaan. Als er dus enkele regels zijn toegevoegd, dienen alle andere regels op te schuiven. Dit gebeurt door het opnieuw nummeren van de regels en het vernieuwen van de informatie die de plaats van de regel op het beeldscherm bepaalt. Hierna wordt het tekstbuffer regel voor regel verzonden.

6.4.3 De uitvoer van een lineprinter-buffer.

Voor de uitvoer naar de lineprinter zijn twee buffers in gebruik. Terwijl het ene buffer wordt weggeschreven naar de printer, kan het andere buffer worden gevuld met nieuwe informatie. Vanuit de hoofdplus wordt steeds de routine UITLP aangeroepen die de buffers kan verwisselen. De invoerwijzer en de uitvoerwijzer worden steeds verzet als de buffers van functie wisselen. (zie diagram 6.4).

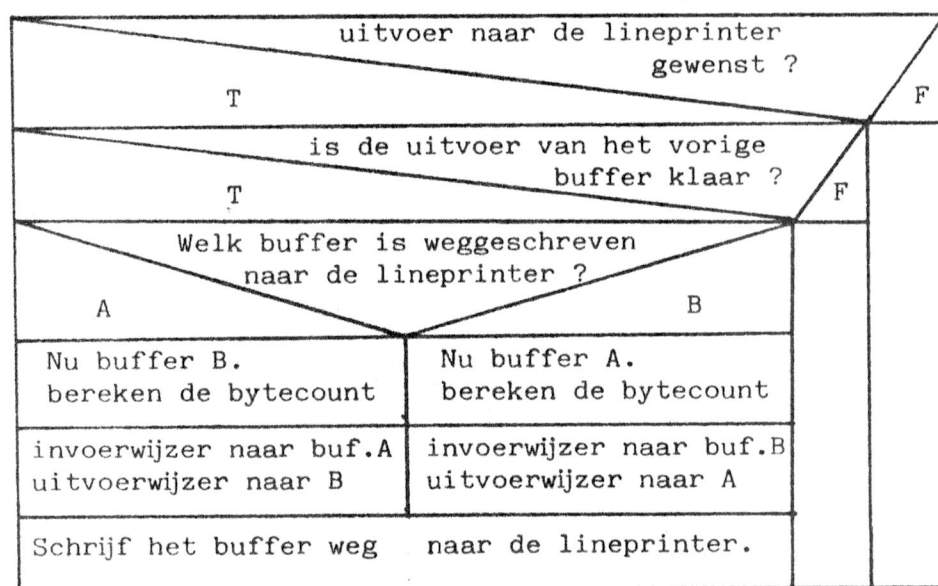


Diagram 6.4: de routine UITLP.

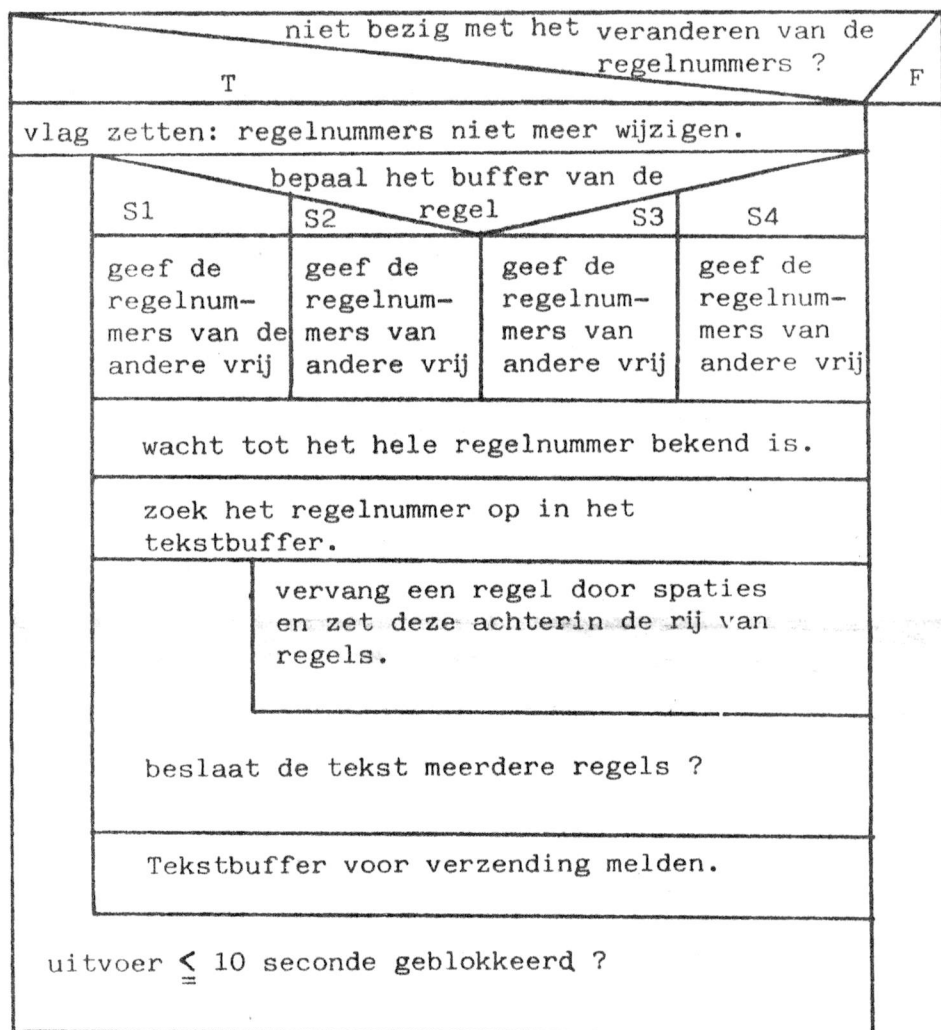
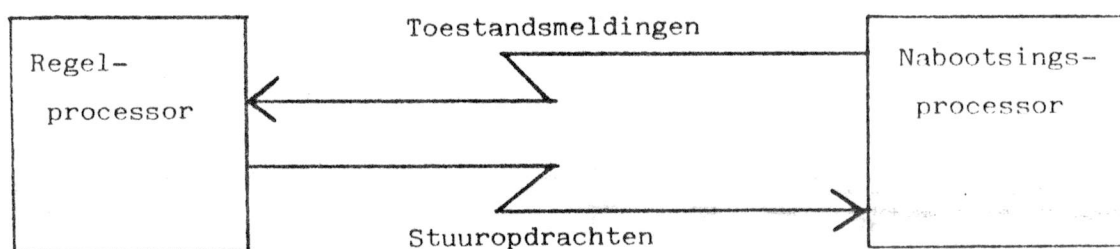


diagram 6.5: de routine WISBEW.

Hoofdstuk 7. De communicatie met het beheersgebied.

7.1.1 Inleiding.

Het contact tussen de CVL-post en het buitengebeuren wordt verzorgd door het Quicktrol-transmissie systeem. Dit systeem is aan de CVL kant verbonden met de apparatuur van de treindienstleider, hetgeen in dit geval wordt gerepresenteerd door de PDP 11-20 regelprocessor. Aan de andere kant bevindt zich de baan-apparatuur met o.a. de beveiligingssystemen. In het laboratorium wordt dit buitengebeuren gesimuleerd door de PDP 11-03 nabootsingsprocessor (zie fig. 7.1).



Figuur 7.1: De communicatie.

Als eis aan deze processoren is gesteld dat ze zo zijn aangepast aan de werkelijke systemen dat ze hiervoor in de plaats gezet kunnen worden. Dat wil zeggen dat de regelprocessor zonder meer in de plaats gezet moet kunnen worden van de CVL-apparatuur. De uitvoer van de regelprocessor moet dan ook aangepast zijn aan het Quicktrol transmissiesysteem. De van de treindienstleider ontvangen opdrachten moeten aan dit transmissiesysteem worden doorgegeven en de terugmeldingen van de baan moeten worden ontvangen via dit systeem. Op deze manier maakt het niet uit of de regelprocessor is verbonden met een echt spoorweg-emplacement of met de nabootsingsprocessor.

In paragraaf 2 t/m 5 zal deze communicatie met de buitenwereld worden behandeld, te beginnen met het verzenden van de eigenlijke code-telegrammen. Daarna zal worden ingegaan op de manier waarop deze telegrammen worden gecodeerd. Paragraaf 4 gaat over het ontvangen van code-telegrammen van de baan en in paragraaf 5 zal worden ingegaan op het nog bestaande verschil met de werkelijkheid.

Hierna volgen 13 codestappen van 90 ms elk, die + of - zijn. Stap 15, de stopstap, is weer -. Tussen twee commando's in zit minstens 90 ms rust na de stopstap (zie lit. 7). De codestappen bestaan uit een adres deel en een informatie deel. In het adresdeel, dat direkt na de startstap komt, staat de stationscode van de onderpost waarvoor het commando bestemd is. Dit zijn meestal 3 stappen. Daarna volgt een groepscode van 3 stappen. De daarop volgende informatie-stappen geven de gevraagde stand aan voor evenzoveel elementen.

Voor de onderpost Blauwkapel-A is het aantal informatie-stappen zo groot dat deze niet meer in een commando passen. Deze zijn dan ondergebracht in twee gekoppelde commando's. In het eerste commando wordt dan aangegeven dat het gaat om een gekoppeld commando. In het tweede commando bestaat de stationscode dan uit een stap met de waarde 0. Dit is tevens de enige stap waarin, bij Blauwkapel, de waarde 0 wordt gebruikt. Hieronder volgen nog enkele voorbeelden van commando's:

Onderpost Blauwkapel Groep A:	+ + + -	+ + + +	+ + + + +	-
	+ 0 - + -	+ + - -	+ + - + -	-

Onderpost Blauwkapel Groep B:	+ + + +	- - - -	+ + - + -	-
-------------------------------	---------	---------	-----------	---

Onderpost Bilthoven:	+ + - + -	+ - - +	+ + + - -	-
----------------------	-----------	---------	-----------	---

* Zie ook hoofdstuk 11.

7.2.2 De zendprocedure.

De verbinding van de PDP 11/20 regelprocessor met de onderposten loopt via twee bits van het DR11-C interface. Hiervoor zijn bit 0 en bit CSR1 gebruikt. Een te verzenden commando staat in een ringbuffer klaar. De zendroutine ZENDEN zoekt het eerste te verzenden commando op in het ringbuffer en zet vervolgens wekkers die aflopen op alle tijdstippen dat het commando van stap verandert (zie diagram 7.1). De wekkers lopen dus af na 150 ms, 240 ms, 330 ms, 420 ms, ... (zie lit. 7). Vervolgens wordt de waarde van de startstap via het DR11-C interface uitgezonden. Iedere keer als er een wekker afloopt, wordt direkt in de interrupt routine de volgende stap verzonden. Deze actie kan niet

7.3 **Het coderen van een Commando woord.**

7.3.1 **Het initieren.**

Als een aanvraag moet worden verzonden, moet deze eerst worden gecodeerd in een commandowoord. Daarbij moet bekend zijn welk commandowoord voor een bepaald element gecodeerd dient te worden. Dit wordt uitgezocht door de routine INICOM. Alvorens het regelprogramma met de verwerking van gegevens begint wordt deze routine aangeroepen. INICOM zoekt alle commandowoorden af. Van alle elementen die daarin voorkomen worden de statuswoorden opgezocht in de betreffende element-tabellen. In het tweede statuswoord van elk element wordt het nummer van het commandowoord ingevuld, waarin het element voorkomt. OP deze manier is, na deze routine, van elk element bekend met welk commandowoord een aanvraag voor dit element gecodeerd kan worden. In INICOM worden tevens alle wissels aangegeven als Normal in de statuswoorden. Voor nadere bijzonderheden, zie bijlage G of de programma's.

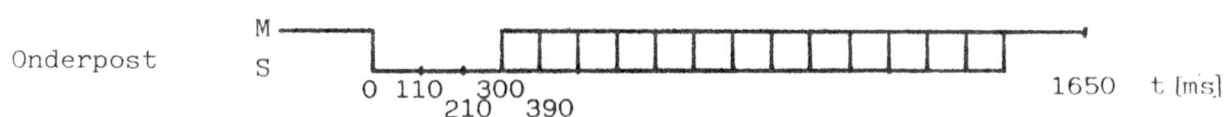
7.3.2 **Het coderen van een commandowoord.**

Indien een enkel element wordt aangevraagd kan in het tweede statuswoord van dit element het nummer worden gevonden van het commandowoord dat gecodeerd dient te worden. Dit coderen gebeurt in de routine CODCOM. Hier worden eerst de startstap en de stappen van de stations- en groepscode ingevuld. Deze stappen worden in het ringbuffer gezet dat bestemd is voor het opslaan van commando's. Daarna wordt van alle elementen die voorkomen in dit commandowoord het statuswoord opgezocht. De betreffende commandostap wordt dan gecodeerd overeenkomstig de toestand van het element zoals deze in het statuswoord staat aangegeven. Bijvoorbeeld:

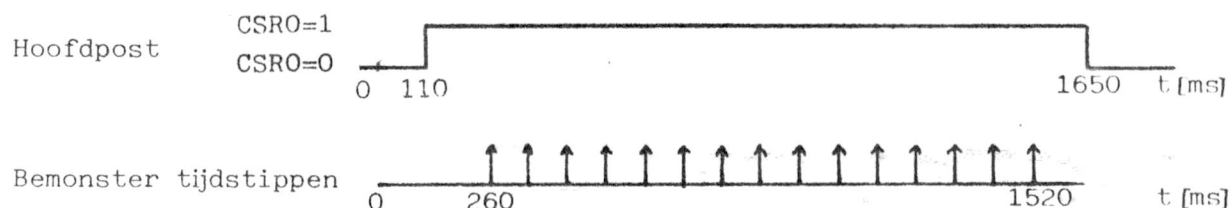
- a) Een wissel Normal wordt als + gecodeerd.
- b) Een wissel aanvraag Reverse wordt als - gecodeerd.

Tot slot wordt de stopstap toegevoegd en wordt het commando aangemeld als verzendingsgereed. Zie bijlage G.

Dit is noodzakelijk daar alleen de middelste 45 ms van een stap significant zijn en de stap in die tijd dus bemonsterd moet worden. Deze stap wordt als byte in een ringbuffer gezet. Als de hele signalering is ingelezen wordt dit gemeld en kan de signalering verwerkt worden via de reeds bestaande routines in het hoofdmodule. Indien gewenst kunnen de ontvangen signaleringen worden getoond op het beeldscherm.



Figuur 7.3: de signalerings structuur.



Figuur 7.4: De ontvangst procedure.

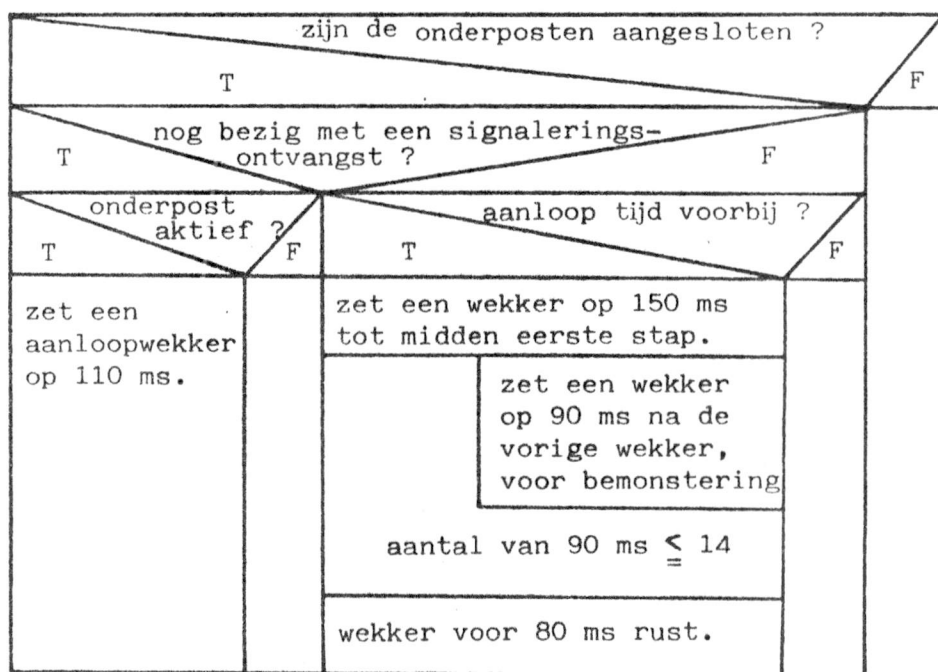


Diagram 7.2: De routine OPMELD.

7.6.1 Element-aanvragen.

Als een aanvraag is gecontroleerd op de juiste syntax, wordt het element opgezocht in de element-tabel. Hier staan de status van het element en het nummer van het commando-woord waarmee de aanvraag kan worden verzonden. In het status-woord wordt gecontroleerd of het element al de gevraagde stand heeft. Zoja dan hoeft er niets te worden gedaan. Heeft het element een andere stand, en is het niet vergrendeld, dan wordt de aanvraag genoteerd in het statuswoord (zie diagram 7.3). Voor de gegevens over de statuswoorden van de diverse elementen wordt verwezen naar het verslag van L.Zigterman (lit: 1, H15.5).

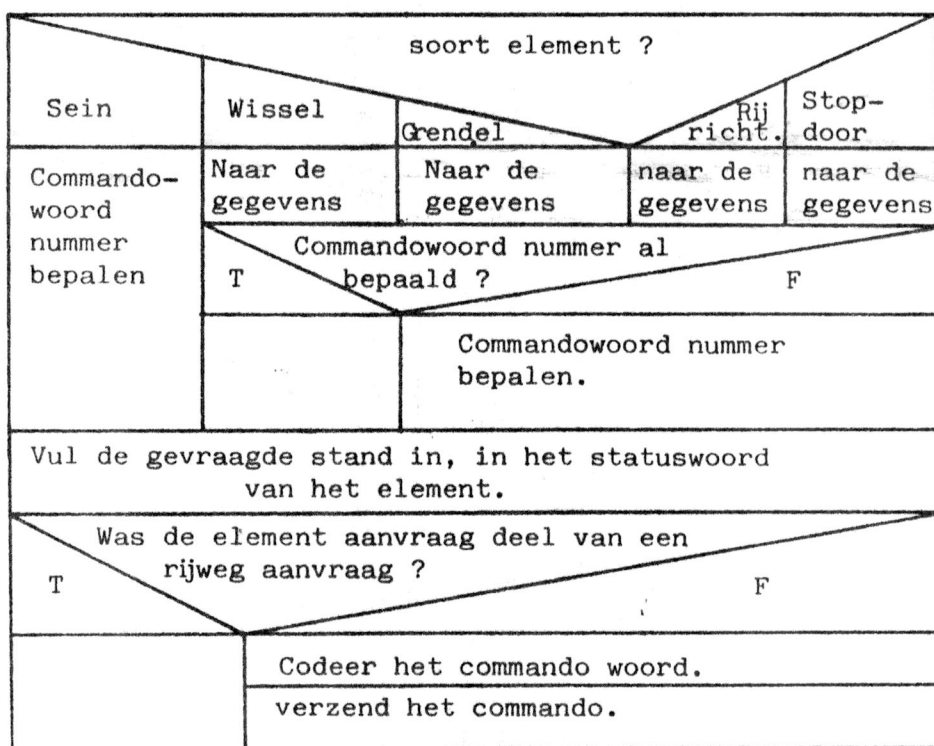


Diagram 7.3: De elementaanvraag routines.

7.6.2 Een wisselaanvraag.

Wisselaanvragen kunnen niet zonder meer worden gecodeerd, daar hier sprake is van twee manieren om een wissel aan te duiden. De

voorkomen, is het te coderen commandowoord niet zondermeer bekend. De NS heeft de commandowoorden echter zo opgebouwd dat alle elementen van een rijweg in een woord voorkomen. Een hele rijweg is dus met een (gekoppeld) commando aan te vragen. Het nummer van dit woord is te vinden is het tweede statuswoord van het toegangssein van de rijweg. Dit is eenduidig bepaald daar elk sein maar in een commandowoord voorkomt. Het aanvragen van een rijweg gaat nu als volgt (zie diagram 7.4): Na de syntaxcontrole van de aanvraag wordt in de routine RA gekeken of de rijweg instelbaar is. Dat wil zeggen, of er geen conflicterende rijwegen zijn ingesteld of aangevraagd. Is de rijweg instelbaar, dan wordt het nummer van de rijweg in een buffer gezet voor verdere verwerking. In de routine SYNRR worden alle elementen van de rijweg in de voor de rijweg noodzakelijke stand aangevraagd. Dit gebeurt door de routines die de enkele elementen aanvragen (zie H7.6) als subroutine aan te roepen. Zijn alle elementen afgehandeld, dan wordt het betreffende commandowoord gecodeerd in de routine CODCOM en verzonden.

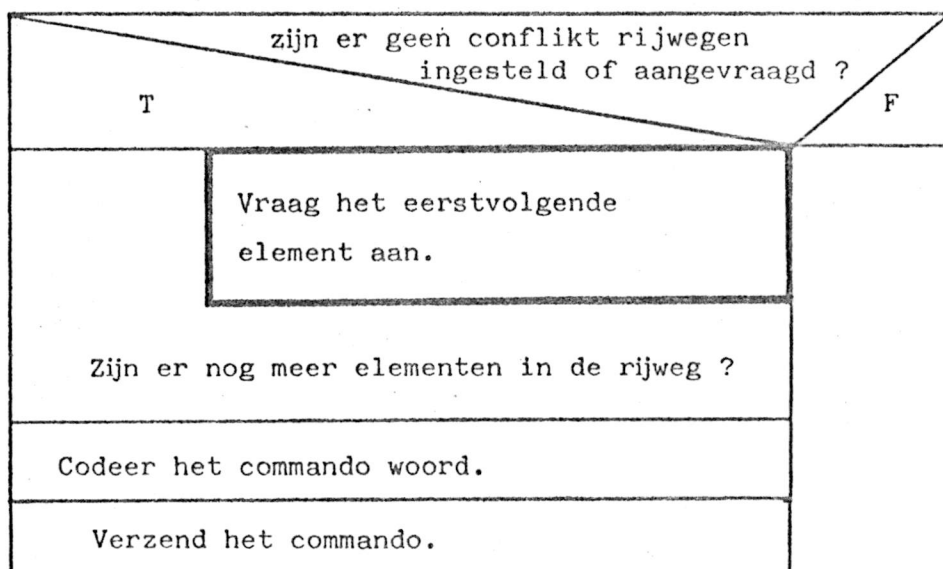


Diagram 7.4: De rijwegaanvraag routine.

Hoofdstuk 8 Het volgen van de treinen.

8.1 Inleiding.

Om het uiteindelijke doel, het regelen van het treinverkeer, te kunnen verwezenlijken, moet de computer kunnen beschikken over voldoende informatie. Deze informatie dient de gegevens te bevatten over de toestand waarin de baan zich op een bepaald moment bevindt. Dit is onder te verdelen in de toestand waarin ieder afzonderlijk element zich bevindt. Daar de verschillende elementen met elkaar samenhangen, dient de computer tevens te weten of de toestand van de elementen samen volgens verwachting is. Ook de tijd speelt bij het treinverkeer een grote rol. De tijden worden bepaald door de treinen. Deze hebben een bepaalde snelheid, wat bijvoorbeeld resulteert in een bepaalde bezettingstijd van secties. De verschillende treinen hebben ook een onderlinge samenhang. De treinen en hun samenhang zijn gedefinieerd in een dienstregeling.

Daar alle informatie afhankelijk is van de treinen, zijn de gegevens ook zoveel mogelijk opgeborgen bij de treinen. Hiertoe is voor elke trein een treinblok gedefinieerd dat alle informatie van de trein bevat, zoals de naam, de tijd, de plaats en de weg (zie bijlage H). Om goed op onverwachte situaties te kunnen reageren, moet de computer steeds weten of alles nog volgens verwachting verloopt. Daarbij wordt er van uitgegaan, dat er niets vanzelf gebeurt. Overal moet opdracht voor gegeven zijn door de treindiensleider. Zo'n opdracht zet een reeks gebeurtenissen in gang die van te voren bekend zijn. Uit de ontvangen signaleringen is nu af te leiden hoe alles verloopt. De zaken die dan door de computer gecontroleerd dienen te worden, zijn de volgende:

1. Het uitvoeren van de opdrachten van de treindiensleider.
2. een treimbeweging. De volgorde van bezetraken en vrijkomen van de diverse elementen.
3. De voortgang van de treimbeweging.

Deze controles worden behandeld in de volgende paragrafen.

Deze gegevens zijn niet verder uitgewerkt. De route bepaalt de aankomstrijweg van de trein.

Deze gegevens worden on-line geconverteerd door de routine CONTRN. Deze routine bouwt een treinblok op voor elke trein.

Hierin staan de gegevens, in een voor de computer hanteerbare vorm. In de routine INIDST die aan het begin van het hoofdprogramma wordt aangeroepen, worden alle treingegevens geconverteerd en samengesteld tot treinblokken. deze treinblokken worden daarna aan elkaar gelinkt. (zie diagram 8.1). Deze linken bepalen de volgorde van de treinen. Tevens maken zij het mogelijk een extra trein later tussen te voegen. Dit is mogelijk daar de treingegevens on-line geconverteerd worden en het gevormde treinblok door het wijzigen van enkele linkadressen is intevoegen in de treinrij. Voor de structuur van een treinblok wordt verwezen naar bijlage H.

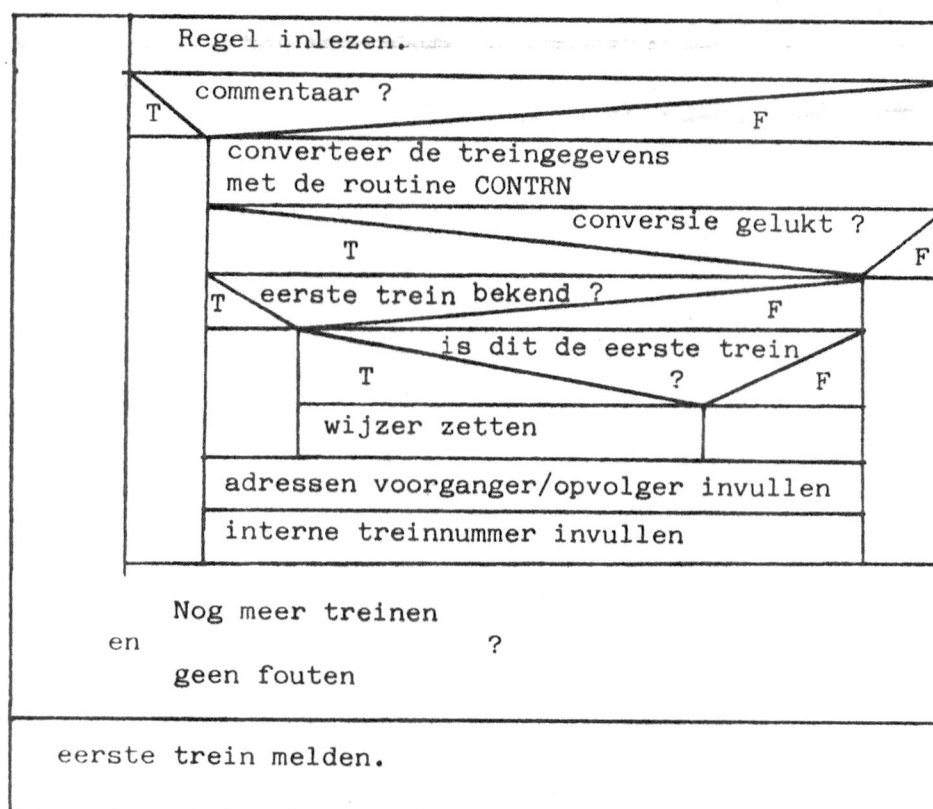


Diagram 8.1: De routine INIDST.

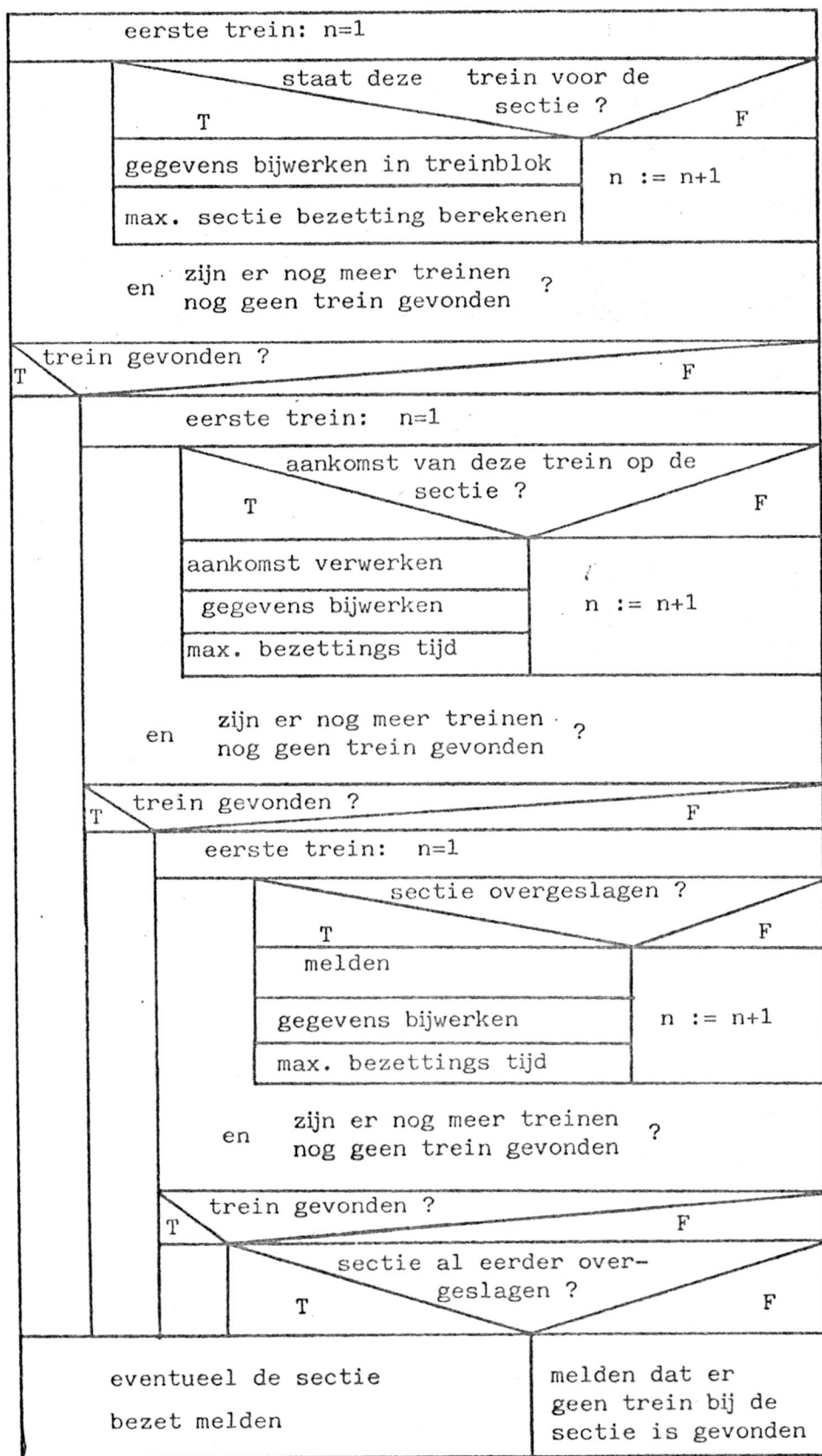


Diagram 8.2: De routine VLGSBS

8.6 Het vrijkomen van een sectie.

Bij de analyse van de ontvangen signaleringen, wordt in TK ook het vrijmelden van secties onderzocht. Aan dit gedeelte is de subroutine VLGSVS toegevoegd (zie diagram 8.4) die gelijkenis vertoont met de routine VLGSBS van de bezetmeldingen. Van alle treinen die in het gebied rijden, wordt in het treinblok gekeken of de staartwijzer van de trein naar de betrokken sectie wijst. Als dit het geval is, kan deze worden doorgeschoven naar de volgende sectie. Alle elementen die daarbij in de rijwegdefinitie worden gepasseerd, zoals wissels, dienen nu vrij te komen. Om dit te controleren worden wekkers gezet.

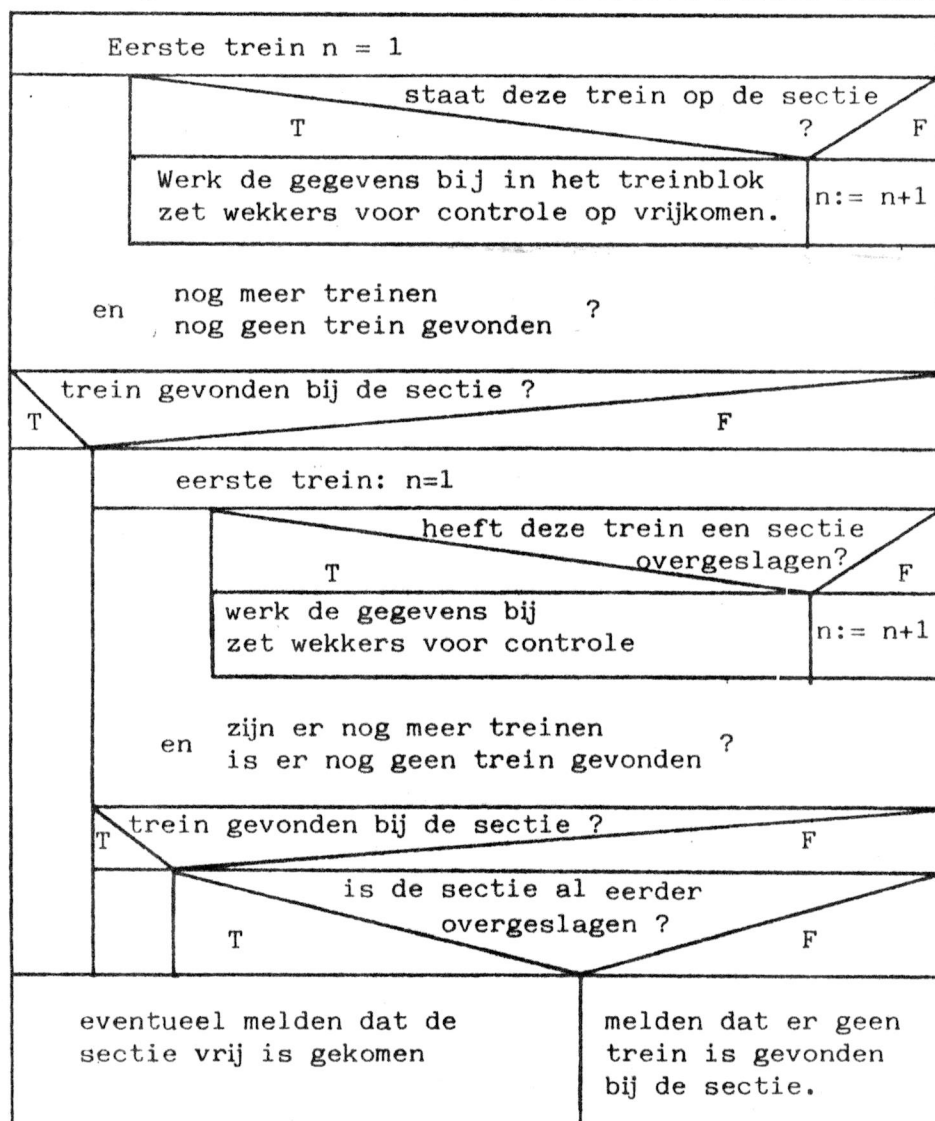


Diagram 8.4: De routine VLGSVS.

Hoofdstuk 9: Het nabootsingsprogramma op de PDP 11-03

Het nabootsingsprogramma dat op de PDP 11-03 draait en het beheersgebied nabootst, is ontworpen om te werken met het Foreground-Background monitor programma. Daar na het opstarten van de processor het Single-Job monitor programma wordt geladen, dient men alvorens het nabootsings programma te starten, over te gaan op de andere monitor. Dit kan met het commando:

.BOOT DXMNFB

Dit monitor programma bevat wekker macro's die niet in de andere monitor voorkomen.

Daar de interfacebus van de PDP 11-03 is uitgebreid, verloopt de communicatie van de PDP 11-20 met de PDP 11-03 nu via een ander interface adres. Hierdoor moesten enkele modules worden veranderd:

RECCOM	nieuwe versie: RECC02
SNDPD2	nieuwe versie: SNDPD3
STNAB0	nieuwe versie: STNAB2
LINK.COM	nieuwe versie: LINK2.COM

Het nieuwe programma heet NAB002 en staat op de werkschijf.

Hoofdstuk 10: Conclusies en aanbevelingen.

1) Het eerste dat ik uit mijn onderzoek kan concluderen, is het feit dat het programmeren in assemblertaal zeer tijdrovend is en volgens mij ongeschikt voor een omvangrijk projekt als Blauwkapel. Het grootste deel van de tijd is men nu bezig met zaken die met de problemen op zich niets te maken hebben.

Aanbevolen wordt dan ook om, alvorens verder te gaan met het projekt, eerst een onderzoek te doen naar een geschikte programmeertaal. Als voorbeeld worden de talen BCPL en C genoemd, die speciaal ontworpen zijn voor het schrijven van systeemprogramma's. Een combinatie van deze talen met assembler routines voor meer specialistische zaken is wellicht ook mogelijk. Het werken in een hogere taal bevordert bovendien de overdraagbaarheid van de programmatuur. Voor het nog te ontwikkelen regelprogramma kan dit van belang zijn.

2) De tabloprogrammatuur is zo gewijzigd dat het sporentablo weer bruikbaar is. De programmatuur is onafhankelijk van de geografie van het gebied, de bedrading van het tablo en de plaats in het geheugen. Door toepassing van een aangepast monitorprogramma in de microprocessor die het sporentablo aanstuurt, is het gebruik van het sporentablo tevens wat meer mensvriendelijk geworden.

3) De communicatie met het beheersgebied vindt plaats door middel van de bij de N.S. in gebruik zijnde coderingen en is aangepast aan het N.S.-transmissiesysteem. De verwerking van de ingelezen signaleringen gebeurt pas als de hele signalering is ontvangen. Dit in tegenstelling tot de N.S.-relaissystemen die een stap direkt verwerken als deze ontvangen wordt. Alvorens een commando te verzenden, wordt dit eerst in zijn geheel gecodeerd, terwijl de N.S.-relaisapparatuur een stap codeert op het moment dat deze verzonden wordt.

Aanbevolen wordt om verder geen tijd te besteden aan pogingen om de computer te laten werken als de relaissystemen, daar er geen duidelijke voordelen bij aanwezig zijn.

Hoofdstuk 11: Protocol van de examenzitting.

1) In H7.2.1 wordt gesproken over het verzenden van commando's. Een 0 op de lijn is voor meer zaken in gebruik dan voor het aangeven van een gekoppeld commando. Na enige tijd 0, beginnen alle onderposten met het verzenden van al hun signaleringen. Hierna is dus in de hoofdpst precies bekend hoe de aktuele situatie in het beheersgebied is. Het op de band opnemen van registraties dient dan ook te beginnen met enige tijd 0 op de lijn, zodat de beginsituatie gedefinieerd is.

2) In H8.3.2 wordt gesproken over de dienstregeling als treindefinities en in H2.3 staat dat er nooit onverwacht treinen door het gebied rijden. In de praktijk kan dit echter best wel eens voorkomen als er iemand vergeet een trein te definieren of een wijziging door te geven aan de computer. In de programmatuur moet hierin voorzien zijn. Voor de computer mag er niets onverwacht zijn. De computer kan dan bijvoorbeeld vragen om wat voor trein het gaat, waarna alsnog een treinblok voor de trein kan worden samengesteld.

3) In H6.6 wordt gesproken over de maximum tijd van 10 seconde voor het intypen van gegevens. Na deze 10 seconde verdwijnt de opdracht van het scherm als men nog niet klaar is en dat is niet mensvriendelijk.

Er mag nooit informatie verdwijnen die de treindienstleider nog niet heeft gelezen. Er dient altijd gewacht te worden op een teken van de treindienstleider dat de tekst verwijderd kan worden. Komt dit teken niet en is het scherm vol, dan dient de uitvoer ergens te worden opgeslagen (logboek of lineprinter). De treindienstleider kan de teksten dan later nalezen of terugroepen op het scherm.

4) Er is nog geen rekening gehouden met kopmakende treinen. Deze treinbewegingen zijn onder de rangeerbewegingen gerangschikt en hiervoor is nog geen programmatuur aanwezig.

5) De communicatie Regelpst-Beheersgebied is nog niet geheel geïmplementeerd. Rijrichting en Stopdoor elementen worden nog niet

Literatuurlijst

1. Zigterman, ir. L.
Gebruik van een processor voor de regeling van het
treinverkeer op een knooppunt.
Afstudeerverslag, vakgroep Automatische Verkeers-
systemen, Afdeling Elektrotechniek, TH-Delft.
Januari 1978, dK/1976/10.
2. Looveren, ir. R.S.M. van
Ontwerp van een besturingssysteem voor een spoorweg-
knooppunt.
Concept afstudeerverslag, vakgroep Automatische
Verkeerssystemen, Afdeling Elektrotechniek, TH-Delft.
Augustus 1979.
3. Coenraad, ir. W.J.
Nabootsing, een hulpmiddel voor het spoorwegproject
Blauwkapel.
Afstudeerverslag, vakgroep Automatische Verkeers-
systemen, Afdeling Elektrotechniek, TH-Delft.
December 1979, dK/1979/7.
4. Schriek, ir. D.
Nabootsing van het gedrag van een spoorwegknooppunt.
Afstudeerverslag, vakgroep Automatische Verkeers-
systemen, Afdeling Elektrotechniek, TH-Delft.
December 1980, dK/1979/17.
5. Vink, H.
Sporentableau "Blauwkapel", delen 1,2,3 en 4.
Stageverslagen, M.T.S. Alkmaar.
6. Holdert, R.
Treindienstleider of treindienstlijder ?
Amsterdam, 1973.

BIJLAGEN

INHOUD

		Blz.
Bijlage A	Waar de programma's te vinden zijn	1
Bijlage B	Hoe de programma's te starten zijn	2
Bijlage C	Het module HFDPGM	4
Bijlage D	Wijzigingen in bestaande programma's	6
Bijlage E.1	De tablo hardware	7
Bijlage E.2	Het M6800 programma PONS68	9
Bijlage E.3	Het conversie programma CONDAT	10
Bijlage E.4	Het monitor programma TABMON	11
Bijlage F	Het module KEYBLD	12
Bijlage G.1	Het module ERRCAR	19
Bijlage G.2	Het module SIGCOM	21
Bijlage G.3	De conversie van wissel informatie	23
Bijlage H	Het module VOLG	24
Bijlage I	Voorbeeld van de uitvoer van het programma	29

Bijlage A.

Waar de programma's te vinden zijn.

De benodigde modules zijn samengebracht in twee Bibliotheken van object modules. Bibliotheek LIB2.OBJ bevat de oude modules, die vaak zijn gewijzigd, in de volgende volgorde:

TABLO, WEKKER, DECTRT, DECODE, ZOEKBP, SCHRLS,
BURO, DGDITD, SPOORW, DECSIG.

De bibliotheek LIB1.OBJ bevat de nieuw ontwikkelde modules:
KEYBLD, SIGCOM, ERRCAR, VOLG.

Met behulp van de Linker zijn het hoofdmodule HFDPGM en de twee bibliotheken samen te linken tot een load module (1 global is niet gedefinieerd).

Op band A zijn de source versies van de LIB1 modules te vinden, alsmede HFDPGM, LIB1.OBJ, LIB2.OBJ en de LOAD versie van het programma.

Op band B staan de source versies van de LIB2 modules, de conversie programma's, de M6800 programma's en tabellen. Als de source programma's gewijzigd zijn, zijn deze voorzien van de extensie .LKB.

Band C is een demonstratie band met alle definitie files, registratie files en de LOAD versie van het programma.

Op schijf DK3: staan alle voor een demonstratie benodigde files onder de inlog code [100,70].

Band 43 is een back-up band. Deze bevat tevens het test-programma TABLO4.TST.

Bijlage B.

Hoe de programma's te starten zijn.

Als men het volledige programma wil demonstreren, werkend met het sporentablo en de PDP 11-03 Nabootser, dient men de volgende handelingen te verrichten in de aangegeven volgorde:

1. Speciale PDP 11 interface PIA's in de microprocessor aanbrengen.
Alsmede: 8K-RAM op adres 0, 2K-RAM op adres A000, CPU, PIA, NMI-Timer en de TABMON monitor.
2. twee kabels tussen het DR11-C2 interface en de micro aanbrengen.
3. Kabel tussen de microprocessor en het tablo aanbrengen.
4. twee kabels tussen de PDP 11-03 en het DR11-C interface van de PDP 11-20.
5. Spanning aan van de microprocessor (of een RESET).
6. Regelprocessor (PDP 11-20) opstarten met de volgende string in de UIC [100,70]:
\$BA DK3:BLAUW
7. Nabootsingsprocessor (PDP 11-03) opstarten:
 - a) Line klok aan (schakelaar links)
 - b) WERKschijf in DX1: plaatsen.
 - c) toets de volgende commando's in:
.BOOT DXMNFB
.RU NAB002
8. Beantwoord de vragen als volgt:
vraag1: 200 wekkers van 4 woorden.
vraag2:
vraag3: keuze 3
vraag4: nog even wachten met beantwoorden.
9. De vragen van de regelprocessor zijn met een RETURN te beantwoorden.
De begintijd van de dienstregeling is: 10:00:00:
Nog geen RETURN intypen.
10. Nu de nabootser starten door de tijd in te voeren:
-10:00:00: RETURN.

Bijlage C.

Het module HFDPGM

Het programma omvat een aantal delen. Als eerste het eigenlijke hoofdprogramma dat alle definitie files inleest en de tabellen opbouwt waarmee in de rest van het programma gewerkt wordt. Na het initieren van het tablo en de wekkerrouines, worden vragen gesteld, aan de treindienstleider, over het programma gedeelte waarmee gewerkt moet worden (zie tabel C.1). Na het intypen van het gewenste nummer of de RETURN-toets, worden vragen gesteld over de uitvoer naar het beeldscherm en de line-printer (zie tabel C.2). Na een RETURN wordt de tijd gevraagd. In het geval van registratieverwerking wordt ook nog de gewenste registratie file gevraagd. Hierna worden de uitvoer, de commandoverzending en de dienstregeling geïnitieerd.

De hoofdlus.

Nu gaat het programma naar de hoofdlus. Deze lus wordt doorlopen tot het einde van het programma. In de hoofdlus staan 10 subroutine-aanroepen. Elke subroutine werkt een programma onderdeel af. Als enkele routines erg veel rekentijd vragen, kan eventueel twee maal de routine AFGEL worden aangeroepen, die de afgelopen wekkers afhandeld, om de wekker afhandeling niet te veel te vertragen.

De registratieverwerking.

Als tweede programmadeel volgt de routine COMAFH, die de registraties verwerkt. Na het zetten van een wekker voor de volgende registratieverwerking, wordt getest of het een signalering of commando betreft. Een signalering wordt verwerkt in de routine RGSIGN. Deze routine gaat, na het zetten van een wijzer, over in de routine SIGN, die ook de signalerings-verwerking verzorgt van de ontvangen signaleringen. De commando-verwerking gaat met behulp van de commando-woorden tabel COMMAN. Hierin staan steeds subroutine nummers gevolgd door element nummers. Via de subroutine tabel ROUTAB worden de

Bijlage D.

Wijzigingen in de bestaande programma's.

Voor alle modules geldt dat de uitvoer is aangepast aan het werken met een statisch beeldscherm. Dit houdt in dat de aanroep van de routine KOPIE is vervangen door KOPFLN of de nieuwe versie van KOPIE. De uitvoer van tekst die plaatsvindt voordat het beeldscherm is geïnitieerd, gaat met behulp van de routine SCHRST. De teksten zijn gedefinieerd in het module KEYBLD en in de modules die deze teksten aanroepen zijn deze dus als global gedefinieerd. Uitvoer van losse tekens gaat via de wijzer TBFWYS.

In de modules zijn verder de volgende wijzigingen aangebracht:

WEKKER: klokfrequentie is nu 100 Hz. zodat de tijd in stapjes van 10 ms wordt opgehoogd. De afhandeling van de communicatie- wekkers gebeurt direkt in de interruptroutine omdat niet precies bekend is wanneer de wekkers worden afgehandeld. Er zijn nu 60 wekkers.

SCHRLS: De routine WRITKB is omgedoopt in SCHRST en wordt gebruikt voor uitvoer voor de beeldscherm initiering. Voor de uitvoer van losse tekens is nu KOPFLN in gebruik in plaats van KOPIE. KOPIE, UITVR1 en UITVR2 zijn overbodig en dus verwijderd.

SPOORW: De datatabellen zijn verdubbeld. De sectiedata tabel bevat nu 5 woorden per element en de verschillende secties zijn te vinden via de nieuwe SECADR tabel die de adressen bevat. Linkblok en buffer voor de file WISLRN zijn toegevoegd. Treinblok ruimte voor 17. treinen van 18woorden gereserveerd.

DECSIG: Door het verdubbellen van de datatabellen moeten deze nu worden aangeroepen via BP2WRD. De sectie-data is te vinden door eerst het adres op te zoeken in SECADR. De treinnummer routines zijn geschrapt. Volg-routine aanroepen toegevoegd.

BIJLAGE E.1

De tablo hardware schematisch weergegeven

Voor de aansturing van een element zijn 16 bits beschikbaar. Voor de lampjes en de leds zijn al deze 16 bit in gebruik om het element te selekteren en zijn dus als adres van het element te beschouwen. Een lamp/led gaat branden als het adres wordt aangestuurd als er een klok-enable signaal optreedt. Een lamp/led gaat uit als het betrokken adres niet is aangestuurd bij een klok-enable signaal.

Bij het aansturen van de treinnummerindikatoren dienen 8 bits als adres van een decoder en 8 bits vormen het treinnummer in BCD-code. Hierna zullen de 8 bits die bij de treinnummers als data bits in gebruik zijn, ook bij de lampen/leds als data worden aangemerkt.

De 8 adresbits zijn opgedeeld in 2 maal 4 bits. 4 bits worden in een Hoofd-demultiplexer gedecodeerd en dienen om 1 uit 16 demultiplexers te selekteren. De overige 4 bits vormen de ingangscode voor deze 16 demultiplexers. Met deze 8 adres-bits kan men zodoende 1 uit 256 lijnen selekteren. Deze lijnen dienen als klok-enable lijn voor de latch/decoder van een treinnummerindicator of voor een blok van 8 latches/buffers voor een lamp/led element. Zo'n element kan bestaan uit 1 led of een aantal lampjes behorend bij een element, bijvoorbeeld 5 lampjes van 1 sectie. Schematisch ziet dit alles eruit als in figuur E1.

Het sporentablo bestaat uit 3 kasten die elk zijn voorzien van een eigen Hoofd-demultiplexer die enkele demultiplexers kan selekteren. Hierdoor is het mogelijk de informatie gecodeerd aan de kasten aan te bieden. Elke kast is voorzien van een 25-polige plug. Hiermee kunnen de kasten direkt aan de microprocessor worden aangesloten om te worden getest. In de normale bedrijfs-situatie is de middenkast aan de microprocessor aangesloten en zijn de zijkasten met de middenkast verbonden.

Bylage E.2

het microprocessorprogramma PONS68.

Op de M6800 microprocessor die het sporentablo aanstuurt, draait, onder de AVSMON monitor, het programma PONS68. Dit programma zorgt voor de inventarisatie van de tablo-hardware. Na het initialiseren van het tablo-interface (een PIA), en het wissen van het tablo in de routine INIT, worden alle tablo-adressen een voor een aangestuurd, te beginnen bij adres 0. Bij elk adres wordt gevraagd om de naam en de soort van het element in te toetsen, volgens een in het programma vermelde syntax. Na het intoetsen van een komma, worden de ingetoetste tekens op ponsband vastgelegd. Het element wordt weer gedooft en het volgende element wordt aangestuurd. Op deze manier vormt de ponsband een ascii-definitie van de elementen van het tablo en de adressen waarop deze elementen zijn aan te sturen. Startadres A200.

De syntax staat in het programma vermeld.

Voor het testen van de tablo-elementen, zonder gebruik te maken van de PDP 11-20 regelprocessor, is een testroutine aanwezig. Deze stuurt het element aan waarvan het Hexadecimale adres is ingetoetst. Het startadres is A203.

Bijlage E.4

Het monitor programma TABMON.

Het in gebruik zijnde monitorprogramma TABMON, is afgeleid van de AVSMON monitor. De niet gebruikte gedeelten zijn geschrapt en vervangen door routines die speciaal voor het sporentablo van belang zijn.

Na een RESET begint het programma met het initieren van het tablo-interface (een PIA) en het wissen van het tablo. Daarna worden van de PDP 11-20 regelprocessor de programma's en de datatabellen ingelezen die voor de aansturing van het tablo nodig zijn. Na het inlezen, wordt het programma gestart en is het tablo gebruiksgereed.

Alvorens de microprocessor te resetten of de spanning aan te schakelen, dienen alle kabels te worden aangebracht:

2 naar de PDP en 1 naar het tablo.

BYLAGE F

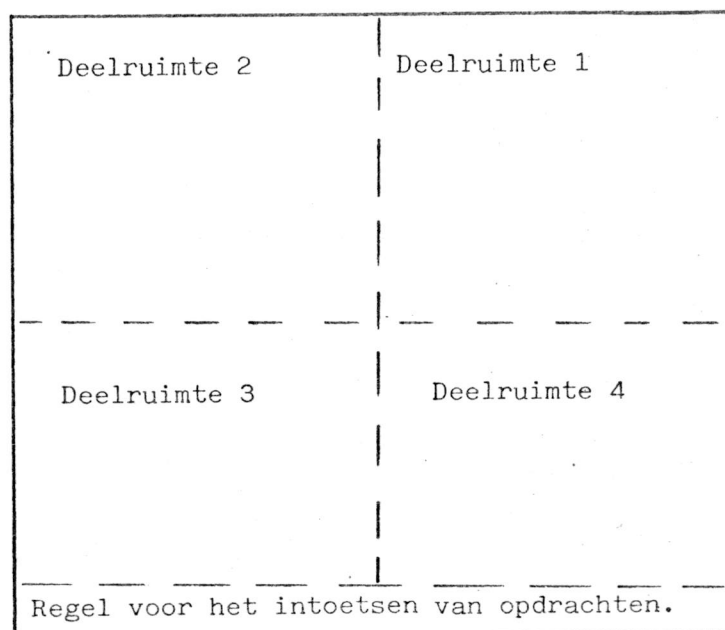
Het module KEYBLD.

Het module KEYBLD verzorgt de invoer via het toetsenbord en de uitvoer naar het beeldscherm en de lineprinter.

Het beeldscherm kan worden verdeeld in 4 deelruimten. Deze zijn te definiëren met behulp van parameters die aan het begin van het module staan. Alle regels in de deelruimten zijn genummerd. Ook deze nummers zijn gedefinieerd vooraan het programma.

het initieren.

De routine INITIO initieert de in- en uitvoer. De invoer gaat op interrupt basis en hiervoor worden de nieuwe interruptvectoren gezet. Hierna worden, overeenkomstig de waarden van de parameters, de scheidingen aangebracht tussen de verschillende deelruimten op het beeldscherm. De deelruimten zijn geplaatst als aangegeven in figuur F.1.



Figuur F.1: De deelruimten.

Het echoen gebeurt door het verzenden van het buffer ECHBUF, dat de informatie bevat van de onderste regel van het beeldscherm.

De uitvoer van tekst.

Het invoeren van teksten in het tekstbuffer gebeurt met de routine KOPIE die wordt aangeroepen als: JSR R5,KOPIE

tekstnaam

(zie diagram 6.1).

Via de subroutine OPENEN wordt, aan de hand van de plaats van de tekst in het module, bepaald in welk tekstbuffer de tekst moet worden opgenomen. Via de invoerwijzer SEKTix, die naar de laatst ingevoerde regel wijst, wordt de cursor informatie en het regelnummer van deze tekst, over de oudste regel van het tekstbuffer heengeschreven. In de routine PRTIJD wordt eventueel de tijd ingevuld, waarna in KOPIE de tekst wordt overgenomen. Indien de tekst langer is dan een regel, wordt, in de routine KOPIE, de routine KOPIE aangeroepen voor de rest van de tekst. Op deze vervolgregel wordt geen regelnummer en geen tijd ingevuld. Als de hele tekst is overgenomen, wordt de rest van de regel opgevuld met spaties en wordt de vlag LINESx gezet, om aan te geven dat de tekst op het beeldscherm niet meer aktueel is. (zie figuur F2). Via de subroutine LPKOPI wordt de ingevoerde regel overgenomen in een buffer voor uitvoer naar de lineprinter.

linkadres, escape F,cursor adres,regelnummer,>/tekst/terminator.

Figuur F2: De regel opbouw.

De uitvoer naar de Lineprinter.

In de routine UITLP, die vanuit de hoofdloop wordt aangeroepen, wordt de uitvoer van een lineprinter-buffer naar de lineprinter verzorgd. Eerst wordt gekeken of het vorige buffer geheel is weggeschreven. Als dat het geval is, wordt getest welk van de twee buffers is weggeschreven. Er zijn twee buffers voor de lineprinter die afwisselend

Na elke vraag wacht de computer tot de gebruiker zijn keuze heeft kenbaar gemaakt door het intoetsen van nummers en/of een RETURN. In de routine WERKBY worden deze nummers vervolgens opgezocht in de tabel BLDTAB. Hier staat achter elk nummer het bijbehorende subroutine adres. In deze subroutines wordt het overeenkomstige bit in het KEUZE register bijgewerkt. Hiermee wordt de werking van het programma bepaald.

Het invoeren van opdrachten.

De routine ECHO, die vanuit de interrupt routine van het toetsenbord wordt aangeroepen, test of het ingetoetste teken een van de tekens is die in de tabel KARSET voorkomen. OP deze tekens wordt direkt actie ondernomen zonder een RETURN af te wachten (zie tabel F.3). Als het om een van deze tekens gaat, wordt er een vlag gezet, de CRVLAG, die aangeeft dat de opdracht verwerkt kan worden. Dit gebeurt in de

EX	=	Programma afsluiten en verlaten
N	=	Instellen van een Route/Rijweg
H	=	Hulptekst of Herroepen van Rijweg
L	=	Wissel linksleidend leggen
R	=	Wissel Rechtsleidend leggen
K	=	Keuze vragen van uitvoer en runmode
^	=	Opdracht opbergen in Tussenbuffer
W	=	Wissen van een tekst
B	=	Bewaren van een tekst
C	=	Gezien toets: merktekens weghalen

Tabel F.3: De actie-tekens uit de tabel KARSET

In de routine WISSEN wordt de bedoelde regel, en eventuele vervolg-regels, uit de gelinkte rij regels verwijderd en achteraan ingevuld. Dat gebeurt door het wijzigen van de linkadressen. Tevens wordt de regel gevuld met spaties. Vervolgens wordt gemeld dat het tekstbuffer niet meer aktueel is.

Het tussenbuffer.

De routines INTUS en TUSSEN regelen de in- en uitvoer van opdrachten in het tussenbuffer. Dit gebeurt via het hulpbuffer S4BUF. Het invoeren gebeurt door met KOPIE het S4BUF in het tekstbuffer over te nemen. Als een opdracht uit het tussenbuffer wordt aangeroepen via het regelnummer, wordt eerst de regel opgezocht. De opdracht wordt vervolgens overgenomen in het S4BUF, waarna de regel uit het tussenbuffer wordt verwijderd met de routine WISSEN.

De uitvoerKeuze.

Via de routine KIEZEN worden vragen gesteld, aan de treindienst-leider, net als in de routine START. Hiermee is tijdens het draaien van het programma en de manier van werken te veranderen.

Bijlage G.1

Het module ERRCAR.

In het module ERRCAR staan de Element, Rijweg en Route Controle en Aanvraag Routines.

Deze routines controleren een aanvraag op de juiste syntax en het nut van de aanvraag en kijken of de aanvraag mogelijk is. Wat niet mogelijk is wordt gemeld, met opgaaf van redenen. Als een aanvraag mogelijk is, wordt deze genoteerd in de statuswoorden en gemeld aan de codeer routines.

Het module bevat de volgende routines:

De routine RA voor de rijweg en route aanvragen (zie diagram 7.4). Deze routine kijkt eerst of het een rijweg betreft en daarna wordt in RART gekeken of het een route is. Bij een route aanvraag wordt het routenummer in het buffer ART gezet voor verdere verwerking.

In de routine RACON wordt van de rijwegen gecontroleerd of er conflict rijwegen zijn aangevraagd of ingesteld. Dit wordt gemeld via R3. Als de rijweg instelbaar is, wordt de aanvraag in RWINST genoteerd in de rijwegstatus en wordt het rijwegnummer in het buffer ARW gezet voor verdere verwerking. In MLDCON worden de conflictrichtingen gemeld.

De routine RH kijkt of een rijweg herroepen kan worden. Het rijwegnummer wordt daarna in het buffer ARH gezet voor verdere verwerking.

De routine SYNRR zorgt voor de verdere verwerking van de aanvragen voor rijwegen en routes. Het nummer van een aantevragen rijweg wordt uit het buffer gehaald. Daarna zoekt de routine alle afzonderlijke elementen op. Deze worden allen als aangevraagd gecodeerd in het statuswoord. Dit gebeurt in de routines voor het aanvragen van individuele elementen. Deze routines hebben twee ingangen. Een voor het aanvragen van slechts een element. Hierin wordt de syntax van de aanvraag gecontroleerd en het nut van de aanvraag. Hierna volgt het algemene deel van de routine dat alleen de aanvraag in het statuswoord invult (zie diagram 7.3). De volgende routines

Bijlage G.2

Het module SIGCOM.

Het module SIGCOM bevat routines voor de signalerings-ontvangst en commando codering en verzending.

De routine INICOM zet bij alle elementen in het tweede statuswoord het nummer van het commandowoord dat voor dat element gecodeerd moet worden. Daartoe wordt het hele commandowoorden-tabel COMMAN doorlopen. Hierin staat steeds een subroutine-nummer gevolgd door 0, 1 of 2 elementnummers. Via de subroutine-tabel INITAB springt het programma naar de bedoelde subroutine. Daarin worden de elementen voorzien van het nummer van het commandowoord waarin ze voorkomen. Dit nummer wordt ingevuld in het tweede statuswoord van elk element. Dit tweede statuswoord is bij elk element, speciaal voor dit doel, toegevoegd aan de element-tabellen. Bij wissels wordt tevens in het high-byte van het tweede statuswoord het ascii karakter L of R ingevuld. Dit geeft aan of bij het wissel de Normal stand als linksleidend of als rechtsleidend wordt aangeduid. Deze informatie is te vinden in de tabel WISLRN.BNR (zie bijlage G.3). Tevens worden wissels gecodeerd als liggend in de Normal stand. Dit gebeurt in het eerste statuswoord.

Het coderen van commandowoorden gaat via de routine CODEER. Deze routine zoekt uit of het gaat om een gekoppeld commandowoord. In dat geval wordt ook het gekoppelde commando gecodeerd.

Het eigenlijke coderen gebeurt in de routine CODCOM. Dit is een routine als INICOM. De commandowoorden-tabel COMMAN wordt doorlopen tot het gewenste commandowoord gevonden is. Het nummer van dit woord is bekend en wordt gewoon afgeteld. De onderweg gevonden stations-codebits en de reservebits worden bewaard. In het gevonden commandowoord staan weer subroutine nummers gevolgd door elementnummers. Via de subroutine-tabel SYNTAB, wordt naar de subroutines gesprongen die de stappen van de betrokken elementen coderen overeenkomstig de aangevraagde stand van het element. Als er geen aanvraag is gedaan,

Bylage G.3

Conversie van de wisselinformatie.

Om vast te leggen of een wissel Linksleidend of Rechtsleidend genoemd wordt in de Normal stand, wordt met de hand een definitietabel opgesteld. Dit is de tabel WISLRN.ASC, die de volgende syntax heeft:

17L,19R,20R,42L,...

Deze tabel wordt door het conversieprogramma CONLRN geconverteerd naar de binaire tabel WISLRN.BNR. Deze tabel bevat voor elk element een woord met daarin een L of een R. Deze woorden staan op de nummervolgorde van de bijbehorende wissels. Hierdoor is de tabel voor het hoofdprogramma makkelijk toegankelijk.

Bijlage H.

Het module VOLG.

In dit module staan routines die de treinen kunnen volgen die door het gebied rijden. Tevens staan hierin de routines die de treindefinities, de dienstregeling, verwerken. De uitvoering van de verzonden commando's wordt gecontroleerd. De voortgang van de treinbeweging wordt gevolgd door een controle op de maximum bezettingstijd van secties uit te voeren. Ook de routines die de tijd bijhouden staan in dit module.

De initierings routines.

In de routine INITYD wordt om de begintijd gevraagd en wordt de 50 Hz klok, de KW11-L, gestart. Daarna wordt in de routine CONTYD de tijd, opgegeven in uren, minuten en seconden, geconverteerd naar kloktikken, 50 per seconde. Dit aantal staat in twee woorden onder het label TYDSEC. In de klokinterrupt-routine TYDINT wordt 50 maal per seconde de tijd opgehoogd met een in het buffer SECFRQ opgegeven waarde. De waarde 1 zorgt voor een real-time werking van de programma's.

De routine INIDST converteert de dienstregeling naar een voor de computer bruikbare structuur (zie diagram 8.1). Deze structuur en de dienstregeling zijn gegeven in tabel H.2 en H.3. Het opbouwen van de verschillende treinblokken gebeurt in de subroutine CONTRN. Tevens wordt de wijzer DSTWYS gezet naar de eerste trein die in het gebied aankomt.

Syntax: Tijd: Treinnummer, Treintype, Route

10:00:05: 0001,10,803881	;Sneltrain Utrecht-Amersfoort
10:00:20: 0002,46,66X804	;Beladen goederen Hilversum-Utrecht
10:00:35: 0003,12,882804	;Stoptrein Amersfoort-Utrecht
10:00:50: 0004,55,7AP704	;Lege goederentrein Lunetten-Hilversum

Tabel H.2: De dienstregeling

rijweg is ingesteld of aangevraagd. Dit wordt gemeld via R3.

Een aantal routines controleert het uitvoeren van opdrachten. Eerst wordt een wekker gezet op het tijdstip dat de controle moet plaatsvinden. In de controleroutine wordt gekeken of de opdracht is uitgevoerd. Het niet uitgevoerd zijn van de opdracht, leidt eerst tot een nieuwe controle op de instelbaarheid. Indien nog steeds instelbaar, wordt de fout gemeld en wordt een wekker gezet voor een routine die nogmaals controleert of de opdracht is uitgevoerd. Het niet uitgevoerd zijn leidt tot een nieuwe testwekker, het wel uitgevoerd zijn leidt tot de melding dat de opdracht nu is uitgevoerd. De volgende routines zijn in gebruik:

		controle	retry
wisselaanvraag:	VLGWNC	ALWCOM	RETRWS
	VLGWRC	ALWCOM	RETRWS
rijwegaanvraag:	VLGSSC	ALRWIN	RETRRW
	VLGSRC	ALRWIN	RETRRW

Sectie bezetting.

In de routine VLGSBS wordt een trein gezocht bij een bezetgemelde sectie (zie diagram 8.2). Eerst wordt gekeken of de trein de sectie als volgende sectie heeft. Daarna wordt gekeken of het om een aankomst gaat en vervolgens wordt gekeken of er een sectie is overgeslagen in de toestandsmeldingen. Van een gevonden trein wordt de kopwijzer doorgeschoven en wordt het treinnummer naar het tablo gezonden. Bij een aankomst wordt gecontroleerd of de trein meer dan een instelbare tijdsmarge te vroeg is. Dit wordt dan gemeld. De aankomst zelf wordt altijd gemeld. Tevens wordt de staartwijzer van de trein ingevuld.

Via de subroutine MAXBEZ wordt de maximale bezettingstijd van een sectie berekend. Dit gebeurt door de lengte van de sectie, die te vinden is in de uitgebreide sectiedatatabel (zie lit.4), te delen door de te definiëren minimum snelheid van de trein. Na deze tijd loopt een wekker af en wordt in de routine VLGSVW gekeken

deze niet is gevonden, wordt via de subroutines SCOVWR en SCOVRY gekeken of een sectie is overgeslagen door de toestandsmeldingen. Als dit zo is, wordt in de routine SCLOST een wekker gezet die in SCLVRY controleerd of de sectie na korte tijd toch nog vrij is gemeld. Dit gebeurt door het sectienummer op te zoeken in het NVRIBF. Als de sectie dan nog niet vrij is, wordt dit gemeld. later als de sectie alsnog vrijkomt wordt ook dit gemeld.

Indien de trein gevonden is, wordt de staartwijzer doorgeschoven naar de volgende sectie. Hierbij passeert deze wijzer alle tussenliggende elementen, zoals wissels, grendels, e.d.. Deze elementen moeten nu vrijkomen, hetgeen gecontroleerd wordt in routines die worden doorlopen na het aflopen van een wekker. Voor wissels wordt in de routine WISWEK een wekker gezet die in de routine ALWISV controleert of de wissel vrij is. Afwijkingen worden gemeld. Tevens wordt dan een wekker gezet voor een retry op het vrijkomen. Alsnog vrijgekomen wekkers worden gemeld.

Als bij een trein geen volgende staartsectie is gevonden, wordt gemeld dat de trein het gebied verlaat. Uit het treinblok worden de kop- en staartwijzers verwijderd en indien mogelijk wordt de wijzer DSTWYS, die het begin van de dienstregeling aangeeft, doorgeschoven. Als alle treinen van de dienstregeling het gebied hebben verlaten, wordt het programma verlaten.

```

> SYNTAX VOOR HET REGELEN VAN DE TREINENLOOP :
> .....N OF H = INSTELLEN EN HERROEPEN VAN ROUTES,
> .L EN R = LINKS EN RECHTS VAN WISSELS,
> .EN BS = OPSLAG IN TUSSENBUFFER EN EXIT.
>ROUTE 803881 WORDT AANGEVRAAGD !!
>RIJWEG 803819 WORDT AANGEVRAAGD !!
>819837
>ROUTE 66X804 WORDT AANGEVRAAGD !!
>RIJWEG 66X731 WORDT AANGEVRAAGD !!
>731824
>10:00:12 RIJWEG 803819 IS NIET INGESTELD.
>10:00:12 AANKOMST VAN TREIN 1 OP SECTIE 033
>10:00:15 RIJWEG 66X731 IS NIET INGESTELD.
>ROUTE 882804 WORDT AANGEVRAAGD !!
>RIJWEG 882870 WORDT AANGEVRAAGD !!
>870860
>10:00:22 803819 IS REEDS INGESTELD !!
>10:00:32 AANKOMST VAN TREIN 2 OP SECTIE 060
>ROUTE 7AP704 WORDT AANGEVRAAGD !!
>RIJWEG 7AP748 WORDT AANGEVRAAGD !!
>748740
>10:00:35 66X731 IS REEDS INGESTELD !!
>10:00:52 AANKOMST VAN TREIN 3 OP SECTIE 075
>10:01:12 AANKOMST VAN TREIN 4 OP SECTIE 023
4N
>RIJWEG 870860 WORDT AANGEVRAAGD !!
>860854
>10:01:36 TREIN 3 PASSEERT SEIN: 870
3N
>RIJWEG 731824 WORDT AANGEVRAAGD !!
>824816
>10:01:58 TREIN 2 PASSEERT SEIN: 731
4N
>RIJWEG 860854 WORDT AANGEVRAAGD !!
>854834
5N
>RIJWEG 748740 WORDT AANGEVRAAGD !!
>10:02:23 TREIN 3 PASSEERT SEIN: 860
>10:02:39 TREIN 4 PASSEERT SEIN: 748
5N
>RIJWEG 854834 WORDT AANGEVRAAGD !!
>834820
>10:02:48 TREIN 3 PASSEERT SEIN: 854
740728N
>RIJWEG 740728 WORDT AANGEVRAAGD !!
>10:03:21 TREIN 4 PASSEERT SEIN: 740
4N
>RIJWEG 824816 WORDT AANGEVRAAGD !!
>RIJWEG 816804 WORDT AANGEVRAAGD !!
>10:03:40 TREIN 2 BEZET
> SECTIE 017 TE LANG
>10:03:46 TREIN 2 PASSEERT SEIN: 824
>10:03:51 SECTIE 017 IS
> VRIJGEKOMEN

```