DTU Library

Data, Models, and Strategies in Human Information Processing

Rasmussen, Jens

Publication date:
1976

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Rasmussen, J. (1976). *Data, Models, and Strategies in Human Information Processing*. Risoe-M No. 8

RISØ

April 1976

N-8-76

Research Establisment RISØ

ELECTRONICS DEPARTMENT

Data, Models, and Strategies

in Human Information Processing

by

Jens Rasmussen

Table of Contents

INTRODUCTION 1
DATA PROCESSING TASKS 2
Data-data transformation 2
Data-model transformation 3
Model-model transformation 3
Model-rule transformation 3
DATA PROCESSING MODELS 4
DATA 5
Individual representation of variables: 6
PROCEDURES 6
CRITERIA & CONSTRAINTS 7
ANALYSIS OF MENTAL ACTIVITY 34
AN ILLUSTRATIVE EXAMPLE 36

INTRODUCTION

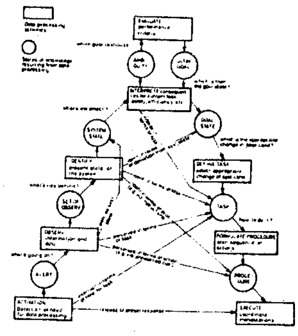
Previously the mental activity of a process operator performing the task of analysing the operating condition of a plant and planning the appropriate actions has been described as a sequence of "states of knowledge" at different levels of abstraction. The "states of knowledge" are connected by data transformation processes and it has been proposed to describe the mental activity of the operator - what he does - in terms of the basic features of the data process which are taken to be;

- The data, i.e., the representations of the state of the variables of the physical system considered
- The model, i.e. the explicit representation of system structure and properties which constrain or control the interrelationship between system variables
- The procedures or rules which are used to control the data processes. Depending upon the complexity of the model available or used the procedures may be general, i.e ' system independent, or specific, implicitly representing system properties and structure.

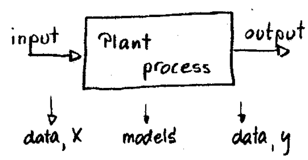
When the mental activity has been described in this way, the choice of the operator, - <u>why</u> he performs in a specific way - has to be analysed by correlating features of the typical data processes to the available mental mechanisms and their facilities and limitations.

DATA PROCESSING TASKS

The "states of knowledge" used in the "ladder og abstraction to describe the steps in the operator's mental activity during a subtask should be considered to be the operators conclusive statements regarding his knowledge of the state of the external physical system and of the appropriate plans and procedures for manipulating the system.

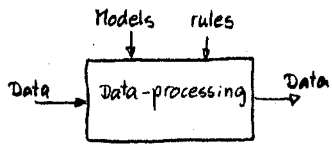


The data processes needed to connect these "states of knowledge" vary fundamentally depending upon the skill of the operator in the specific task. In very familiar tasks states can be connected by a one-step associative leap. In infrequent or unfamiliar situations,the data process needed to connect two of the states of knowledge included in the "ladder of abstraction" can be a complex sequence of elementary data processes connecting "substates" of knowledge regarding intermediate results or regarding states and procedures related to the operators internal data processes.



The basic features of the processes vary widely depending upon the purpose or goal, e.g., whether it **is** part of an analytical or a planning phase of the task. Some of these basic features are different roles of data, models and procedures during the processes.

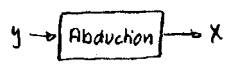
Different types of processes are illustrated by the following classes of transformations:



Data-data transformation

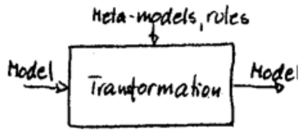
<u>- Deduction;</u> derivation of dependent data from other data; e.g. causal forward transformation to predict output response of system to given input or to derive not-observable information from observations.

<u>- Abduction;</u> reverse transformation, determining input data to system as possible causes of observed output set, e.g., interpretation of instrument readings in terms of internal process variables, explain system output data in terms of unobservable disturbances.



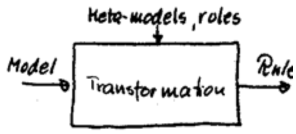
Data-model transformation

- Induction; derivation of information on system structure and properties from data; e.g. find system properties which satisfy a specified input-output relation during planning or diagnosis.



Model-model transformation

- Model transfer; e.g. to transfer to a domain or type of model for which standard procedures or solutions are known; e.g. transfer to known, analogic systems, to standard engineering "equivalent diagrams" or to mathematical representations.



Model-rule transformation

Planning, determination of a sequence of operations, i.e. internal data processes or external actions, for a specific task.

These transformations are only examples of processes related to the discussion of models & procedures. Other mental processes related to values,'90als & criteria such as choices and judgements are of course of great importance.

DATA PROCESSING MODELS

A physical system is a set. of interacting physical elements, which act as carriers of sources emitting information. The interrelationships in the information from the different sources are constrained by the physical interaction of the elements. In data processing such physical constraints, determined by the structure of the system and the properties of the elements, can be explicitly represented by a model of the system.

The data processes are controlled by rules or procedures, which may implicitly represent system properties and structure.

System structure and properties can be modelled by different means:

- Physical models; identifying potential data sources and their spatial relations.

  The mapping preserves the physical structure of the system, i.e. information-sources can be aggregated into sets representing physical objects, i.e., objects are sets of information sources frequently met in different contexts. Objects are formed in the model domain by closed, non-overlapping boundaries. Level of aggregation can be changed by rearranging such boundaries according to the need in the specific data process.

  Examples: Scale models, topographic maps, anatomical diagrams.

- Functional model; identifying potential sources of information together with rules specifying the relations between data available from the sources.

  The rules can be related to the system, e.g. the rules can be causal physical laws, or to the data process, e-g., they can be calculating algorithms.

  The functional model can be unstructured, monolithic, specifying relation between variables at the source level, i.e. the system is considered one integrated whole, one component, a black box.

However, generally a functional model is structured;

- into objects and rules for their behaviour and interaction, e.g. burners, boilers, pumps.
- or into functions; i.e. the data sources are aggregated into sets according to frequently useful relations between information from the sources, e.g. heat balance, mass transport, feed-back loops.

- State model; re-presenting the potential sources of information together with consistent sets of possible values of the variables related to the sources. System properties are implicitly represented by sets of "snap-shots" of variables.

  State models represent frequently used system states and are typically used to identify or verify such states. Therefore, the models are generally labelled by verbal statements referring to:

  - specified operational states:

    "air system normal"

    "its burning well"

  - or to intended operations

    "boiler ready for start"

- Behavioural model - dynamic representation of system behaviour; animate state models. The model **is** formed by storing dynamic representation of input-output relations. Structured in typical objects and their generic patterns of behaviour.

  This model is an "active" model only possible in a parallel processor.

Data processing models can be of a very different nature, depending upon the level of generalization, the selection of properties needed and modelled etc.

The following figures illustrate the content of different types of models. They are in a way models of data processing models, and should only be taken as illustrative examples of possible model content, not as direct representations of the models illustrated.

DATA

In mental data processing the "data" are representations' of the variables characterizing the actual state of the physical system considered. The "variables" of a physical system are here taken to be the measurable, unidimensional magnitudes which generally are used to characterize the state of a physical system in a more strict engineering treatment, and which typically are the primary magnitudes measured in process plants by the instrumentation, such as position, temperature, pressure, velocity, etc.

The physical variables of a System are represented in mental processes in different typical ways:

Individual representation of variables:

- continous, 'analog', representation found e.g., in pulse-density coding in basic neural processes underlying perception and subconscious sensorymotoric functions. Animate imagery in "visual thinking".
- digitized representation, used in calculating algorithms.
- discrete representation, factual statements in natural language reasoning, e.g., high, low, hot.

Collective representations of sets of variables related to objects or functions, states or actions etc. This representation implies "chunking",' a change in level of abstraction based on a stored state model.

A close relation between the type of model underlying a data process, and the type of data used in the process can be expected.

Use of collective representations typical for common sense or natural language reasoning is probably related to functional models structured in components. The variables are attributes of the components or objects, which interact or are means for action.

In formal reasoning based on individual representations of variables a functional model structured in relations is used. Objects or components are means for transforming variables and they dissolve into relations between variables - which become the "objects" or elements of the model.

Following Bugelsky, one could suspect in both cases unsupported verbal reasoning to be a verbal control and/or a verbal expression of the response of an internal, dynamic behavioural model. This model may simulate the behaviour of a system of interacting real objects, as well as of a system of artificial objects, i.e., signs & symbols.

PROCEDURES

The procedures, decision rules, which are used to control the data processes, depend in type and complexity upon the task and the model used for the task.

If a complex active model is used in parallel processing, e.g. analog simulation by a behavioral model, the process is completely specified by the model, if properly initialized and activated. The problem is transferred to another physical system and process control left to nature.

Data processing by a single channel processor implies sequential processes controlled by a procedure, a set of rules, and a stationary model. The content of the rules depend upon the task and the model available. If a complex, structured model is used, the rules can be very general; if only a rudimentary model is used, the rules will reflect implicitly the structure and properties of the system:

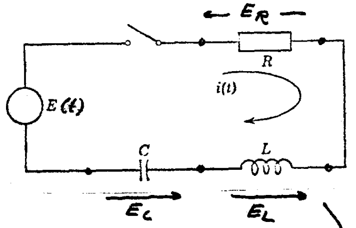
- general rules related to system properties, i.e. physical laws, are used in deductions by causal functional models.
- general rules related to the type of task are found as heuristic rules e.g. in search sequences.
- rules may be related to the structure of the model, e.g., conventions for manipulation of sets of equations
- rules may be related to a specific task and system as it is the case for instructions, cook book recipes based on physical system models only.

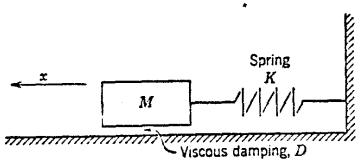
CRITERIA & CONSTRAINTS

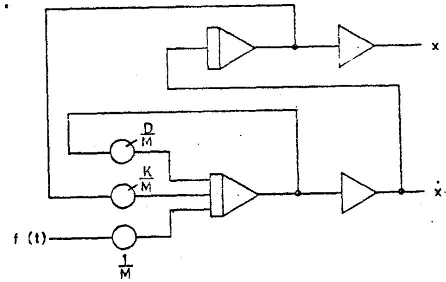
As a specific task can be solved by the use of different' models & procedures the choice of the process used in the task must be controlled by performance criteria related to the process itself. The performance criteria used **will** vary with the specific work situation, and can be related to personal emotions and needs, to internal constraints such as mental capacity limitations or to external constraints such as amount of information or time available.

Structured, functional models of 2' order systems

Physical Models ; Map of elements & structure of 2' order physical systems.



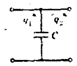




Functional Models ; 'Monolithic' models of components. Relations represented by math. equations.

$E_L = L\frac{d^2q}{dt^2}$

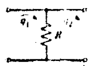
L = inductance



$E_C = \frac{1}{C}(q_1 - q_2)$

C = capacitance

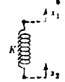
$i = \frac{dq}{dt}$



$E_R = R\frac{d(q_1 - q_2)}{dt}$

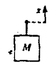
R = resistance
q = charge

generalize



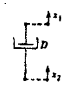
$f = K(x_1 - x_2)$

K = spring constant



$f = M\frac{d^2x}{dt^2}$

M = inertia

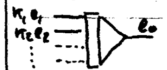


$f = D\frac{d(x_1 - x_2)}{dt}$

x = displacement
D = damping coefficient

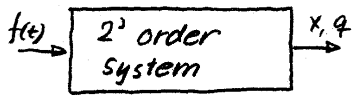
solve by analogy. (behavioural model sol.)

$e_o = -(k_1e_1 + k_2e_2 + \dots k_ne_n)$



$e_o = -e_o(0) - \int_0^t (k_1e_1 + k_2e_2 + \dots\dots\dots k_ne_n)dt$

Monolithic (generalized) functional model



$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{q}{C} = f(t)$$

$$M\frac{d^2x}{dt^2} = -D\frac{dx}{dt} - \dot{K}x$$

$$M\frac{d^2x}{dt^2} + D\frac{dx}{dt} + K_sx = f(t)$$

Process rules for Monolithic Model of 2' order system.

$f(t) \rightarrow$ **2' order System** $\rightarrow x$

$$M\frac{d^2x}{dt^2} + D\frac{dx}{dt} + K_s x = f(t)$$

Rules related to type of model

Cook book recipe for specific output (task). (ω_0, ζ).

$$p^2 + \frac{D}{J}p + \frac{K}{J} = 0,$$

$$\sqrt{\frac{K}{J}} = \omega_0 = \text{undamped natural frequency},$$

$$\frac{D}{2\sqrt{KJ}} = \zeta = \text{damping factor}.$$

Use of Laplace Transform

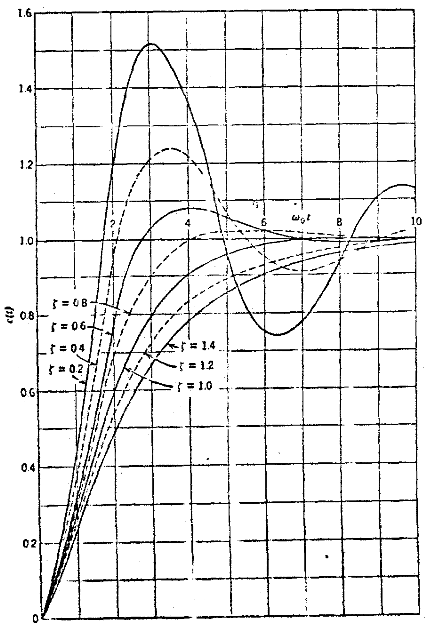
The work involved in using the classical approach to the solution of linear differential equations may be simplified to a routine process through the use of the Laplace transform and its inverse, which uses the same approach to obtain both transient and steady-state solutions

Other set of generally applicable rules related to type of model

General Linear Differential Equations. A common type of system equation, the general linear integro-differential equation with constant coefficients may be written in terms of the input $x(t)$ and the driving function $y(t)$ (Ref. 13) as

$$(22)\quad a_0\frac{d^n x(t)}{dt^n} + a_1\frac{d^{n-1}x(t)}{dt^{n-1}} + \cdots + a_{n-1}\frac{dx(t)}{dt} + a_n x(t) + a_{n+1}\int x(t)\,dt + \cdots + a_{n+q}\int^q x(t)\,dt^q = y(t).$$

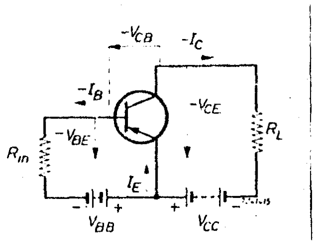
As a class, the homogeneous equation resulting from reducing the right-hand side of the equation to zero has as its general solution a linear combination of solutions of the exponential form $e^{p_n t}$, where p_n may be real or complex.



14. Response of second order system: $C(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}R(s)$

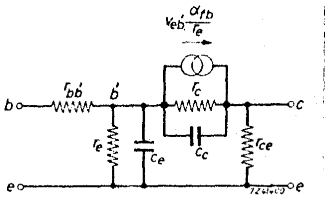
Generalized functional models = models related to generally useful repertoire of procedures for model transformation & data processing.

Functional Model of transistor amplifier stage.

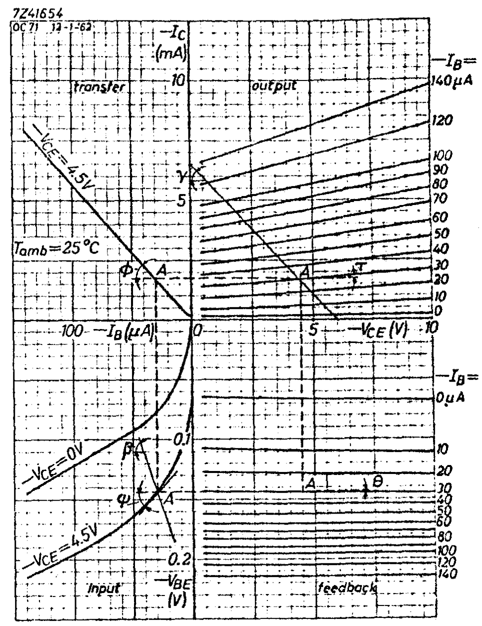


Schematic physical model

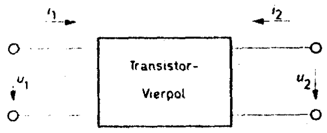
Functional model of transistor



Schematic map referring relations to standard electronic circuit conventions



Relations expressed in graphic diagram conventions.



$$u_1 = h_{11} \cdot i_1 + h_{12} \cdot u_2$$

$$i_2 = h_{21} \cdot i_1 + h_{22} \cdot u_2$$

Kennwerte bei $T_U = 25\ °C$

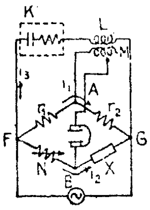
h-Parameter bei $U_{CE} = 5$ V, $I_C = 2$ mA, $f = 1$ kHz

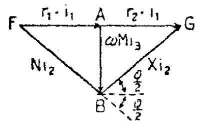
Stromverstärkung	h_{21e}	220 (125 . . 260)
Eingangswiderstand	h_{11e}	2,7 (1,6 . . 4,5)
Ausgangsleitwert	h_{22e}	18 (< 30)
Spannungsrückwirkung	h_{12e}	$1{,}5 \cdot 10^{-4}$

Functional model of transistor.

Relations expressed in standard mathematical conventions.

Functional Model of measuring bridge.





Schematic physical model

Relations expressed in geometric conventions.

Functional Models of capacitor circuits.

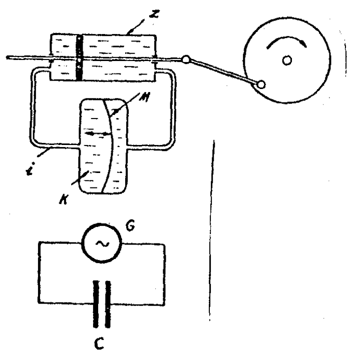


Abb. 26
Mechanisches Modell zur Erklärung des „Stromdurchgangs" durch einen Kondensator.

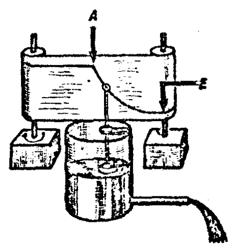


Abb. 21
Mechanisches Modell zur Erklärung der Entladung eines Kondensators.

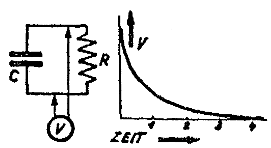
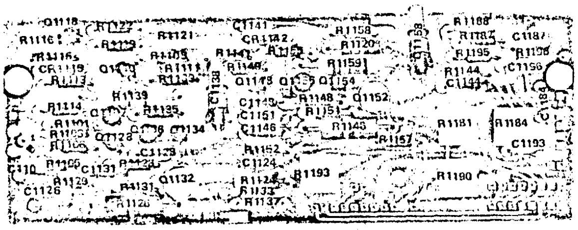


Abb. 22
Diagramm des zeitlichen Spannungsverlaufes während der Kondensatorentladung.

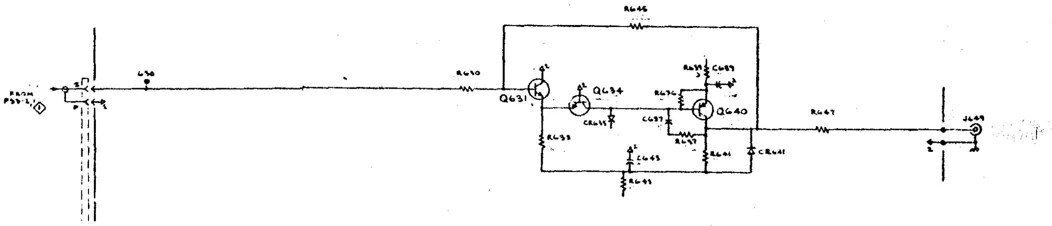
Relations expressed by analogy to familiar relations (or behaviour).

Models of transistor circuitry.



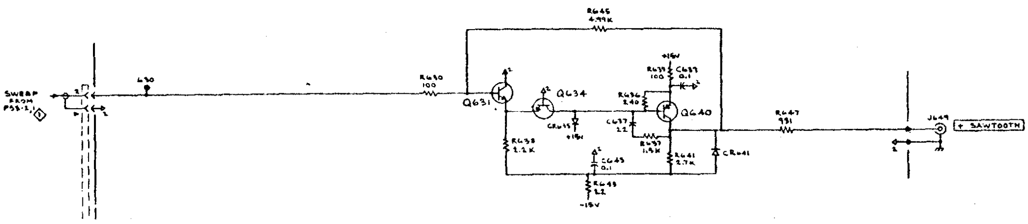
Pictorial physical model

Typically used to correlate schematic physical model to real world.



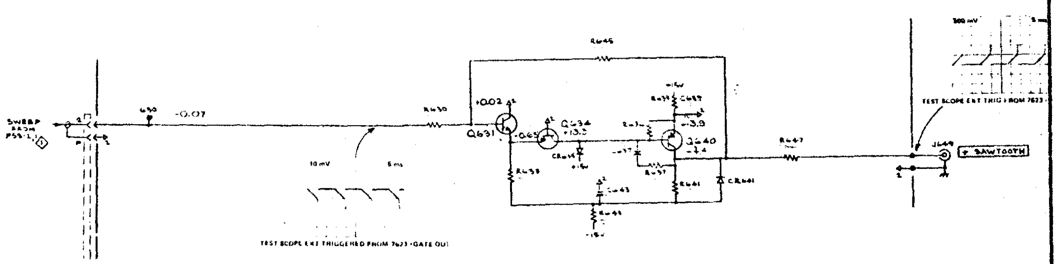
Schematic physical model.

Topographic map of components & connections.



Functional model =

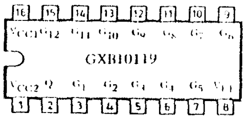
The map together with component properties & standard circuit conventions.



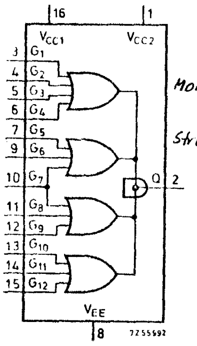
State model Topograpic map with normal state values (voltages & waveforms

Models of logic element

Monolithic physical model:



Functional model: symbolic & application oriented.



Monolithic – physically.
Structured – functionally

relations represented by graphic symbols

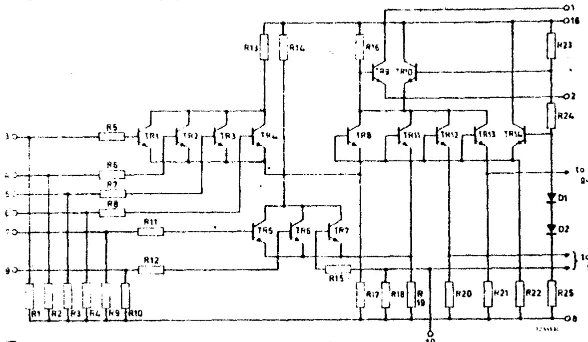
$$Q = (G_1 + G_2 + G_3 + G_4) \cdot (G_5 + G_6 + G_7) \cdot (G_7 + G_8 + G_9) \cdot (G_{10} + G_{11} + G_{12})$$

Positive logic: HIGH state = 1
LOW state = 0

and by math. equation

Related to designers goals & decissions. Can not be derived from system properties and causal laws

Structured, physical model:



Physical structure of system maintained.
Supports use of task independent rules for causal deductions.

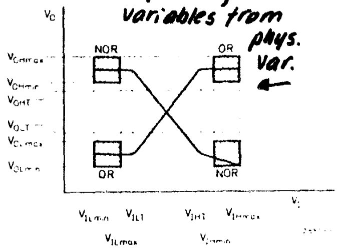
Data for state model:

CHARACTERISTICS (d.c.) at V_{CC} = ground; V_{EE} = −5, 2 V

Each GX circuit has been designed to meet the d.c. specifications shown in the test table below, after thermal equilibrium has been established. The circuit is in a test socket or mounted on a printed circuit board and transverse air flow > 2, 5 m/s is maintained. Outputs are terminated via a 50 Ω resistor to −2, 0 V. Test values for applied conditions are given in the table and defined in the figure.

Test table

T_{amb}	0	25	75	°C
V_{IHmax}	−0, 840	−0, 810	−0, 720	V
V_{IHT}	−1, 145	−1, 105	−1, 045	V
V_{ILT}	−1, 490	−1, 475	−1, 450	V
V_{ILmin}	−1, 870	−1, 850	−1, 830	V

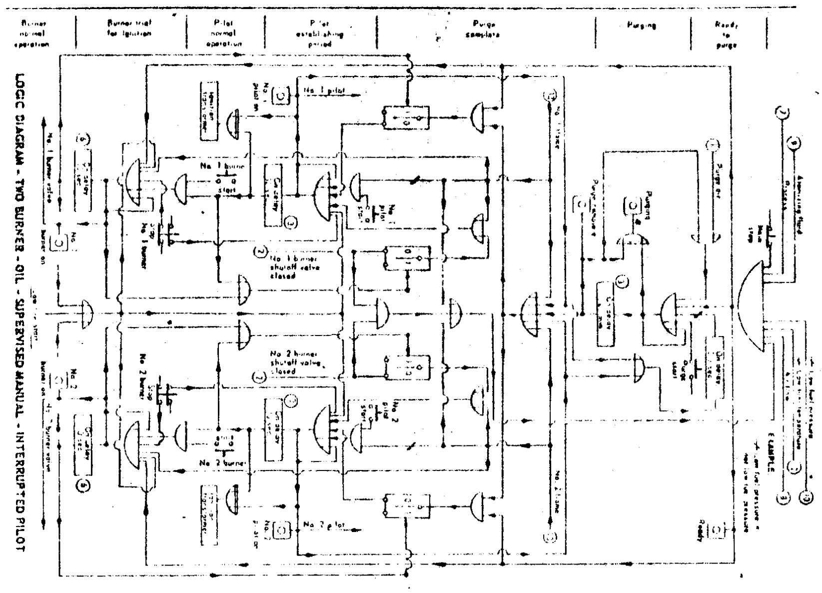


Defines symbolic variables from phys. var.

Related to system properties & variables

	Symbol		T_{amb} (°C) 0	25	75		Conditions
Output voltage HIGH	V_{OH}	min.	−1000	−960	−900	mV	Inputs at V_{IHmax}
		typ.	−	−880	−	mV	
		max.	−840	−810	−720	mV	
Output voltage LOW	V_{OL}	min.	−2, 000	−1, 990	−1, 970	V	inputs at V_{ILmin}
		typ.	−	−1, 720	−	V	
		max.	−1, 665	−1, 650	−1, 625	V	
Output threshold voltage HIGH	V_{OHT}	min.	−1020	−980	−920	mV	Inputs at V_{IHT}
Output threshold voltage LOW	V_{OLT}	max.	−1, 645	−1, 630	−1, 605	V	Inputs at V_{ILT}
Input current HIGH	I_{IH} pin 10	max.	−	355	−	µA	V_{IHmax} for input under test
	other inputs	max.	−	265	−	µA	
Input current LOW	I_{IL}	min.	−	10	−	µA	V_{ILmin} for input under test
Supply current	I_{EE}	typ.	−	20	−	mA	V_{ILmin} for all inputs
		max.	−	26	−	mA	
	$\frac{dV_{OL}}{dV_{EE}}$	typ.	−	0, 25	−		

Functional Model structured in functions symbolically, not refering to component structure.



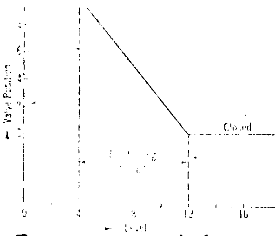
Input-output variables refering to states of components in connected plant system, i.e., to task objectives.

Supports formal deductions. Descriptive model, not one-to-one relation between physical components & elements of model, no simple relation between changes of system & corresponding modifications of model.

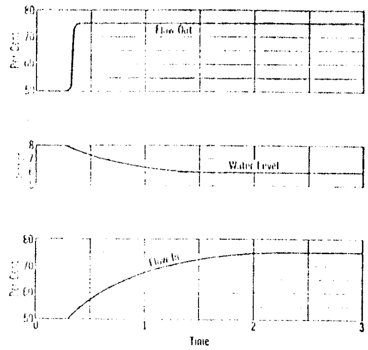
Typically use for complex, task-variable system structures, e.g., electronic information processing systems. (abstract "flow medium")

Models of Dynamic Systems

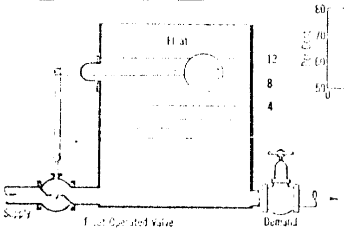
Descriptive models



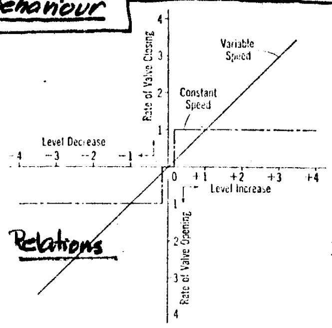
Functional relation



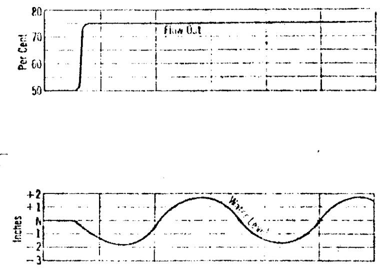
Behaviour



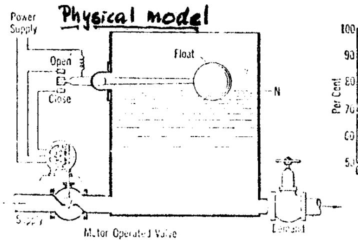
Physical model

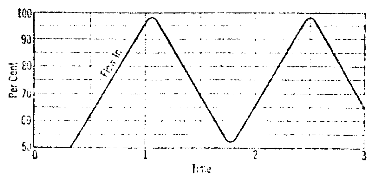


Relations



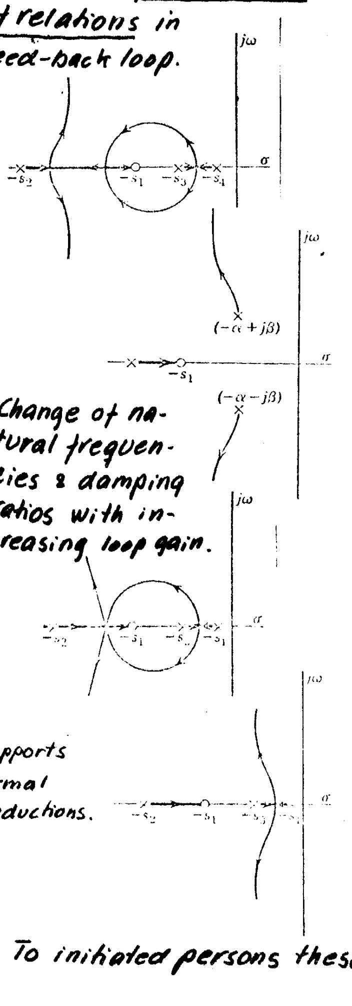
Physical model





Behaviour

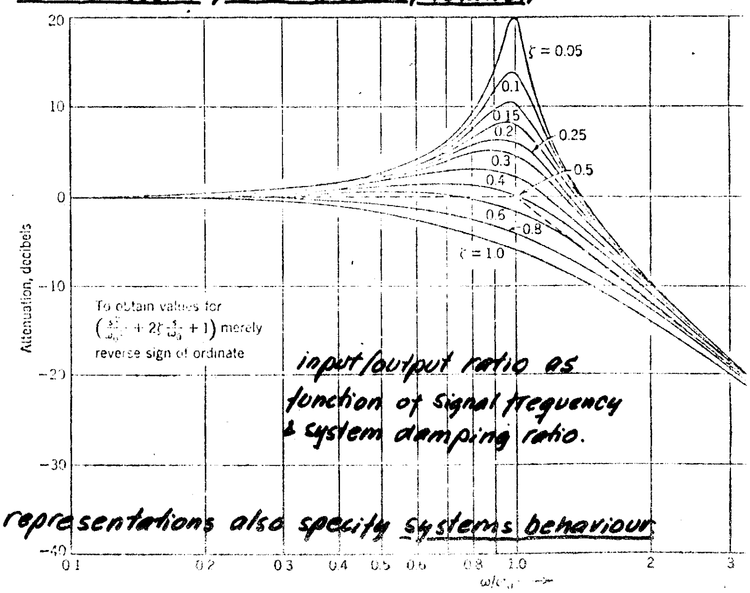
Symbolic representation of relations in feed-back loop.



Change of natural frequencies & damping ratios with increasing loop gain.

upports
ormal
leductions.

Generalized representation of relation



input/output ratio as function of signal frequency & system damping ratio.

To initiated persons these representations also specify systems behaviour.

Process Plant Models

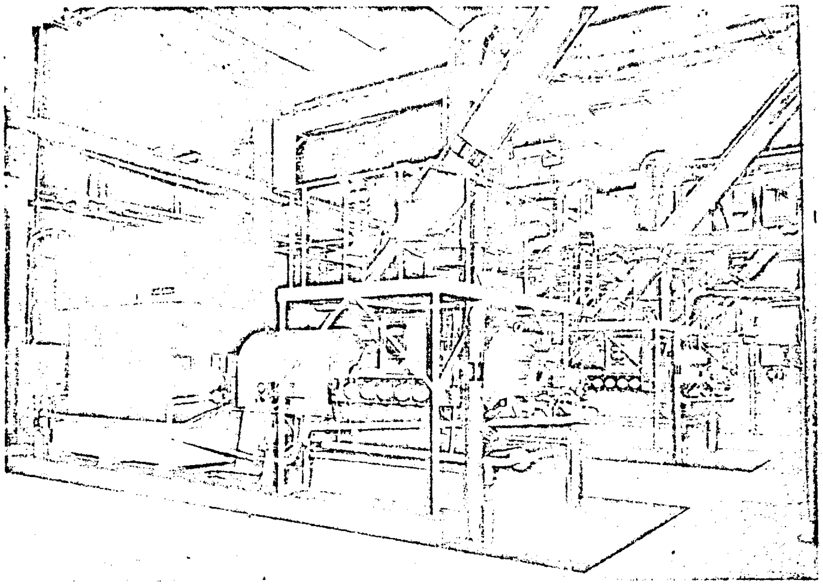


Fig. 12. Fødepumper.

Physical Model

Pictorial model, typically used to find where to look and to 'kick'.

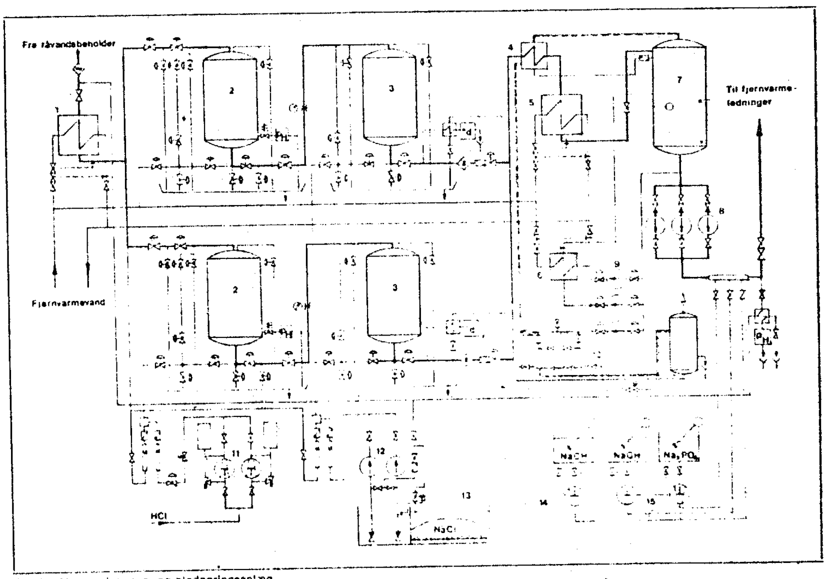
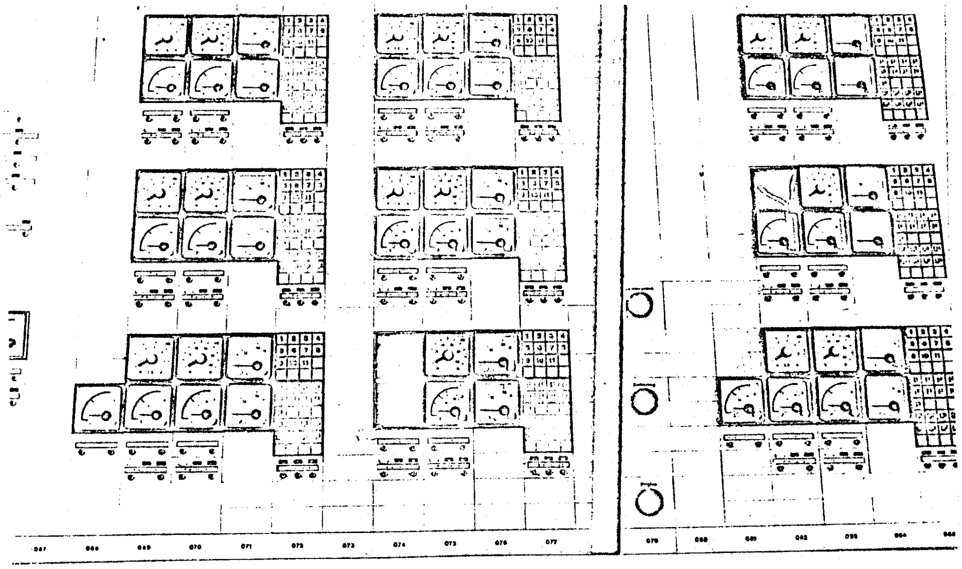


Fig. 20. Alkarboniserings- og blødgøringsanlæg.

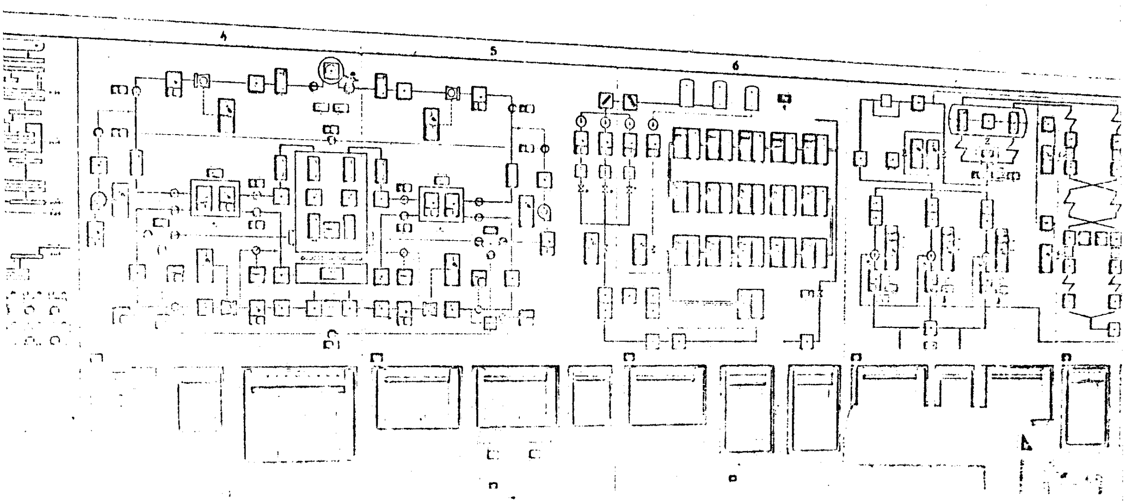
1 Råvandsforvarmer.
2 Alkarboniseringsfilter.
3 Blødgøringsfilter.
4 Emkøler.
5 Blødvandsforvarmer.
6 Afluftervarmeveksler.
7 Vacuumaflufter.
8 Aflufterpumper.
9 Recirkulationspumper.
10 Vacuumpumper.
11 HCl-doseringspumper.
12 NaCl-doseringspumper.
13 Saltsaturator.
14 Doseringspumpe for NaOH til vandkonditionering.
15 Doseringspumpe for NaOH og Na_3PO_4 til vandkonditionering.

Physical Model, schematic map. To professional operator symbols refer to familiar relations - functional model.

Process plant models



Pictorial physical model. Surface model - can be used only for task & system specific procedures (cook book recipes). (if taken as state model - the state is 'power off'. (Indicators on zero)).



Physical model
- with dual level topography.

Process Plant Models

Physical Model

topographic, pictorial map.

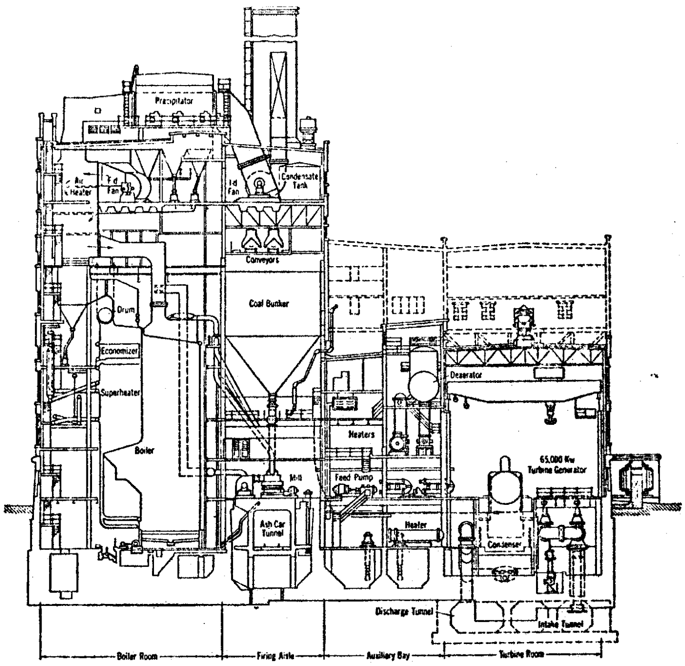


Fig. 9–1. Section through Springdale Unit No. 7, West Penn Power Company.

State Model

Structured schematic map of component and flow-paths with state values.

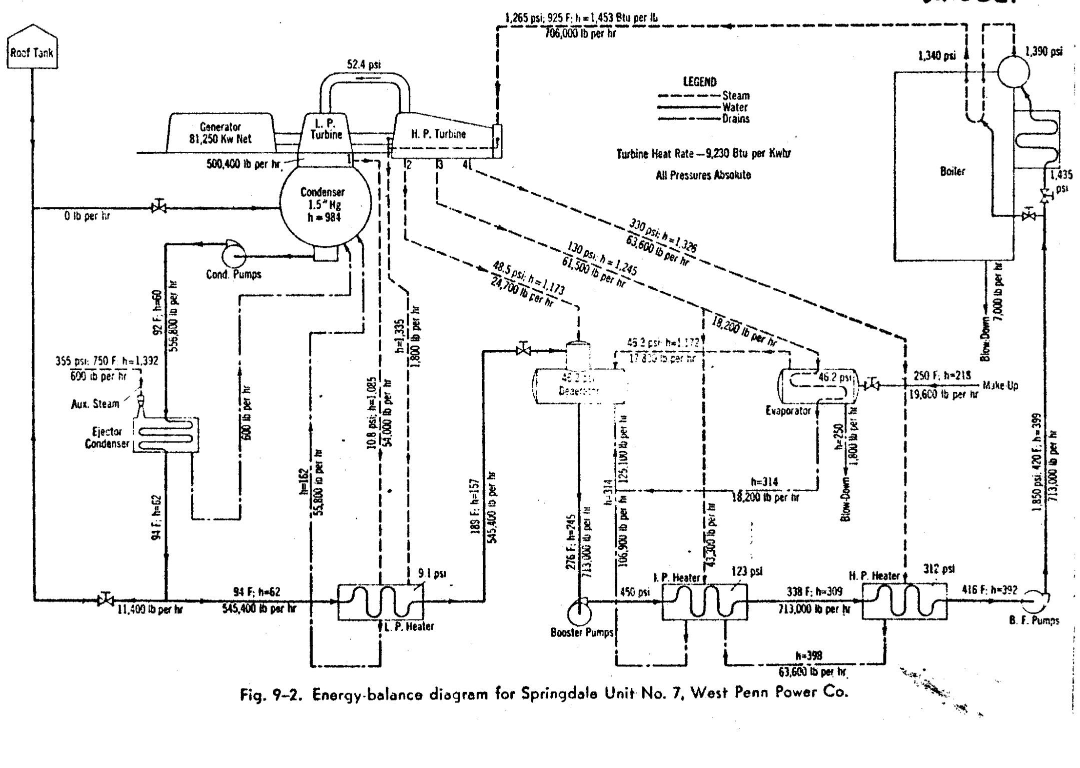
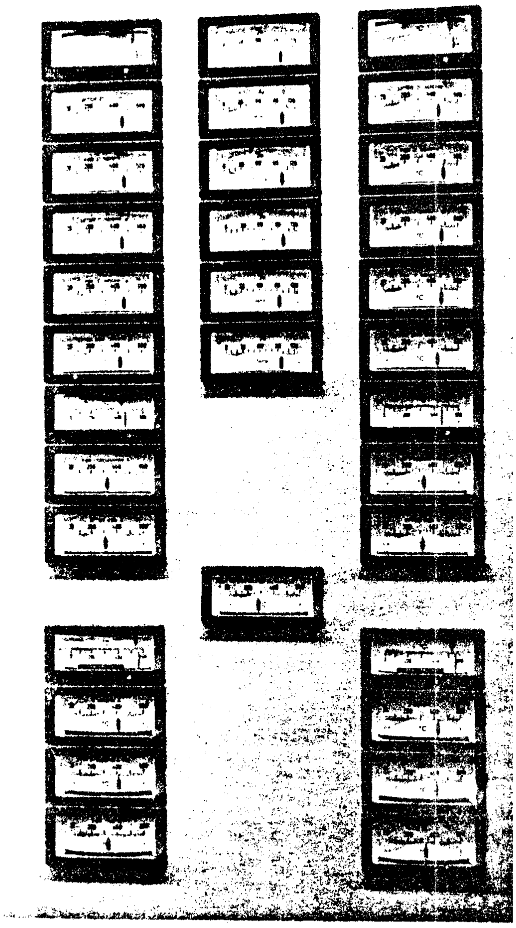


Fig. 9–2. Energy-balance diagram for Springdale Unit No. 7, West Penn Power Co.

State Models



Monolithic (unstructured) state model; - temperature profile in "once-through-boiler" is normal.-

The display has built-in reference state model - data for two independent, equal sections.

Monolithic state model.

Geographic map with data pattern.

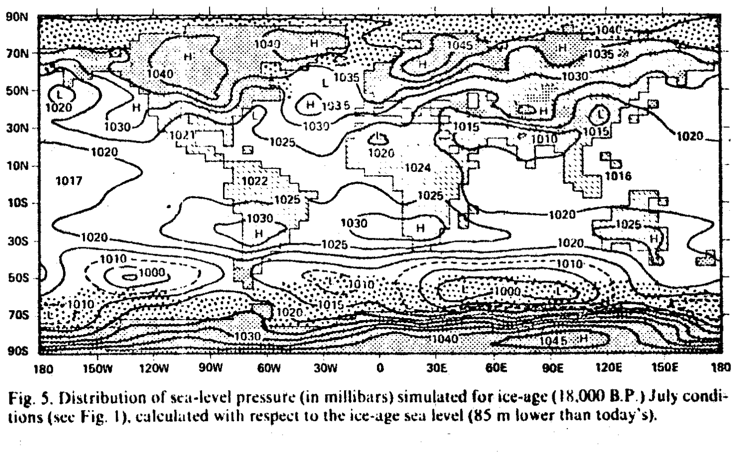


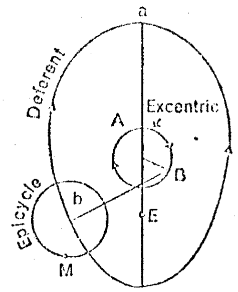
Fig. 5. Distribution of sea-level pressure (in millibars) simulated for ice-age (18,000 B.P.) July conditions (see Fig. 1), calculated with respect to the ice-age sea level (85 m lower than today's).

Satelite system models

Describing functional models

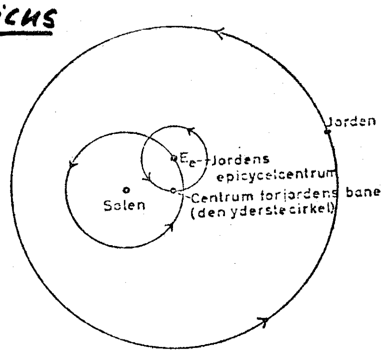
Relations expressed by a complex structure of simple uniform and circular movements to fit the observations. Structure and constants in representation of relations are system-specific.

Ptolemy

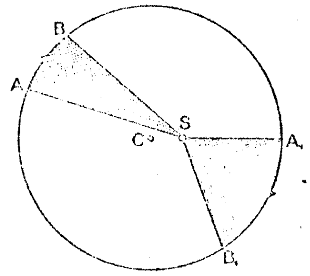


Egg-shaped Orbit of Mercury, according to Ptolemy: E = Earth; M = Mercury.

Copernicus



Kepler

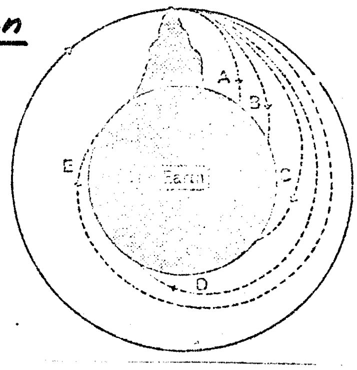


Area $\overline{ABS}$ = Area $\overline{A_1B_1S}$

Describing functional model.

Relation expressed by simple rule controlling circular non-uniform movement. Constants system specific, rule specific for type of system

Newton



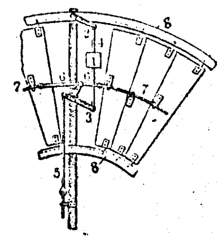
Causal functional model

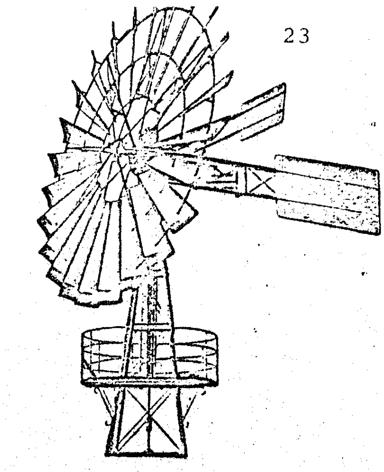
Non circular, non uniform movements. Relations expressed by universal physical laws. Only constants are system specific.

Different types of models can give satisfactory predictions. But the complexity of changes needed to fit changes in the system – e.g. due to component failures – depend strongly upon the model.

Selectivity of Models

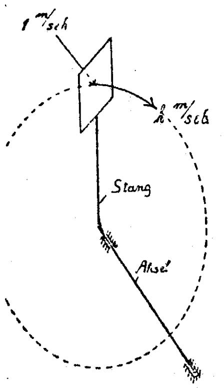
Physical Model

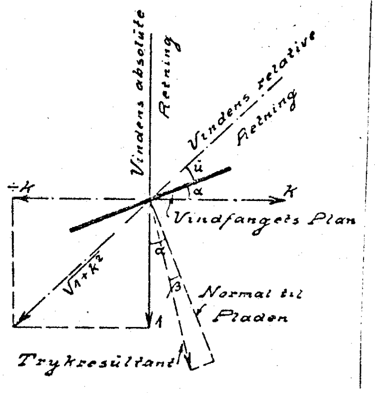




Functional Model

(individual representation of measurable variables)



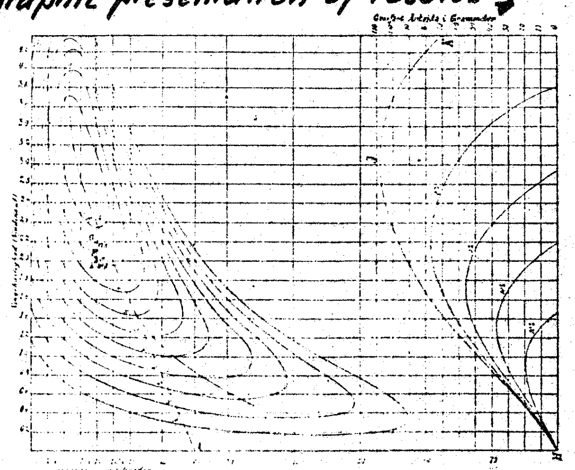


Relation expressed by procedure to calcu-late magnitudes

$$A = \frac{t(1 + k^2)\,k \cdot \sin(\alpha - \beta)}{\cos \beta}$$

relating power out-put to system variables:

graphic presentation of results



Functional models typically only include relations ser-ving a specific purpose.

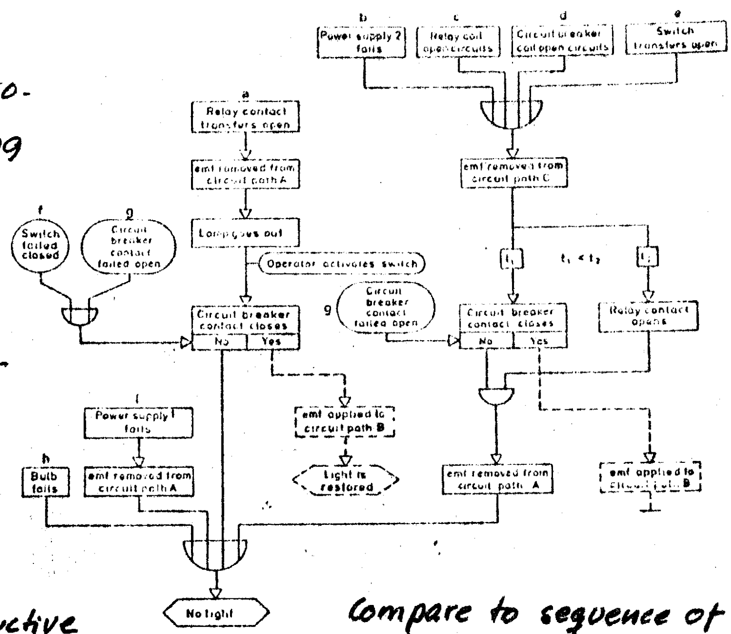
Infrequent tasks may meet inappropriate models, e.g., in-adequately structured models.

Functional plant model

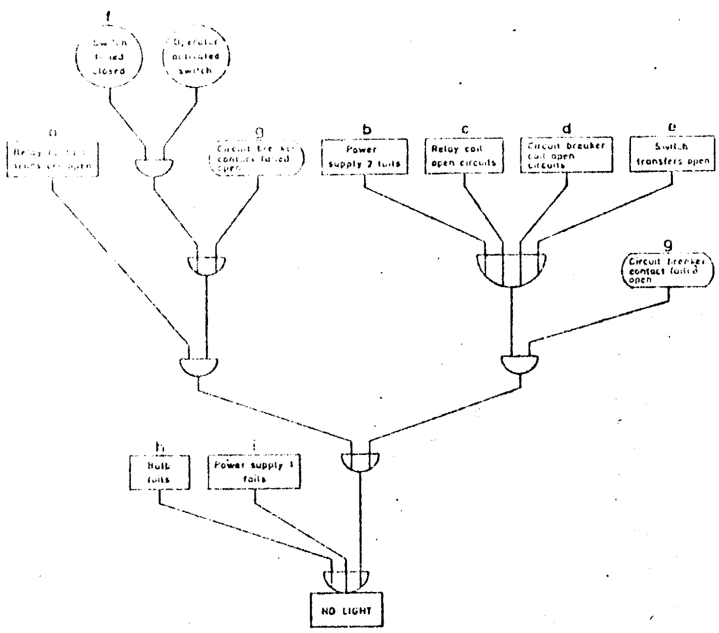
Structured in functional relations connecting component states (collective representation of physical variables)

Model includes only relations & component needed to deal with one variable's output state.

- Record of set of deductive sequences connecting <u>potential</u> object states (collective variables) and actions (properties).



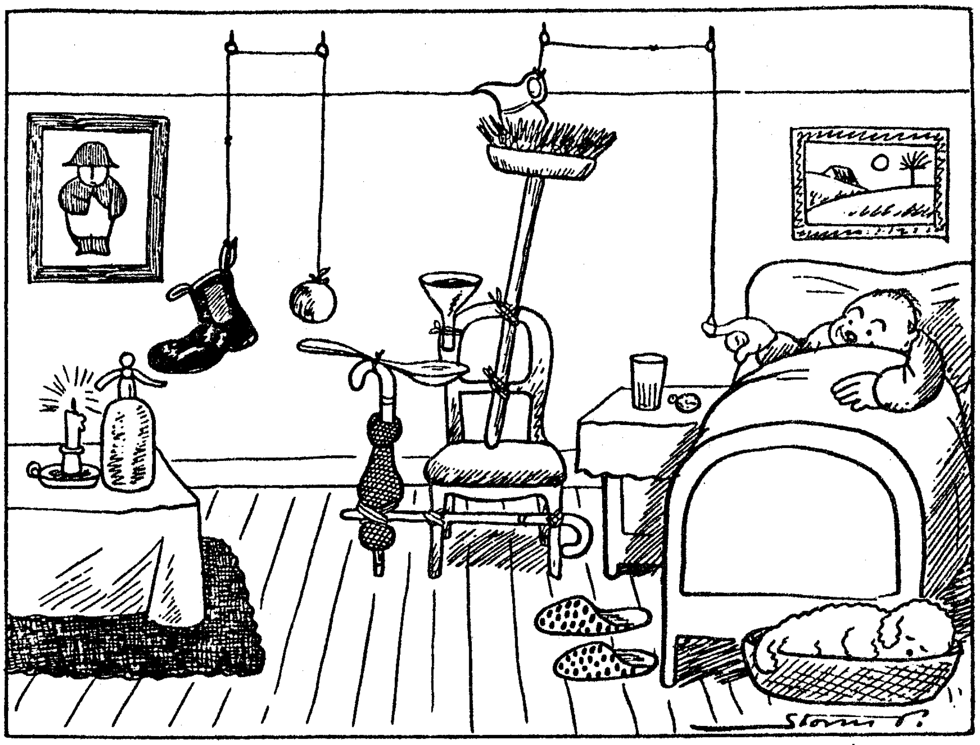
Compare to sequence of statements used to analyze the drawing next page.



Plant state model

representing the set of component states which are labelled "No light."

Used for pattern recognition (- pattern matching -)



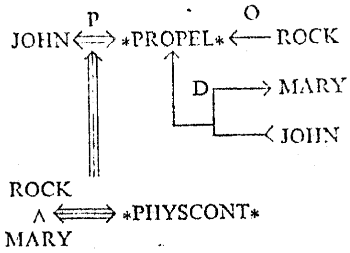
<u>Physical model</u> of a mechanical system - i.e. a system displaying a visible process.

The obvious value properties of the system & familiar properties & attributes of the system components immediately activates the observer's internal <u>behavioural model</u>.

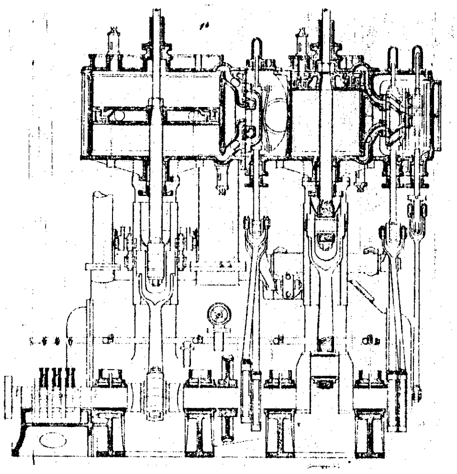
Or - the picture supports an internal <u>functional model</u> based upon familiar properties of system components; a sequence of cause & effect relations between components is traced. A collective representation of measurable variables, i.e., component states & actions, is used. And only qualitative statements are made.

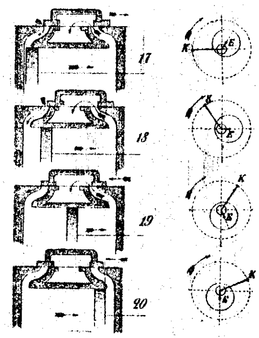
Natural Language Reasoning.

Natural language refer to models structured in objects, their properties and actions. Statements initiate and control the performance of behavioural models.

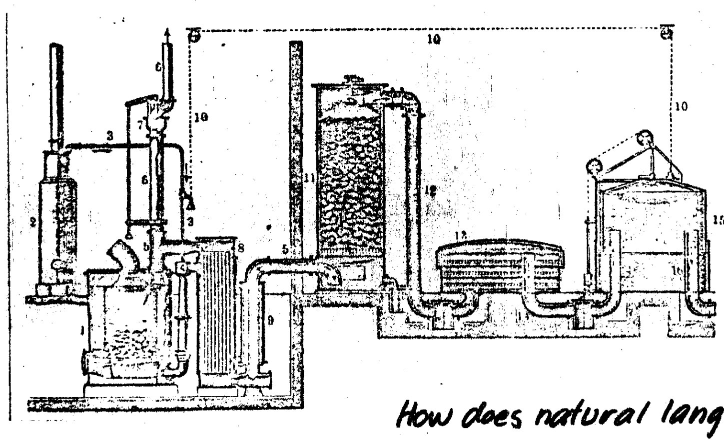


("John throws the rock at Mary")





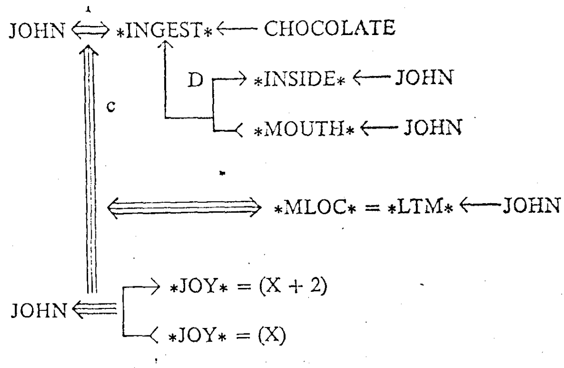
This is the case for space-time systems, e.g. mechanical systems, when reasoning is supported by pictorial physical models. The components or objects act.



Also, if a physical model specifies the scenery for the behaviour of invisible objects of amoebean nature, e.g. liquid, gas, heat.

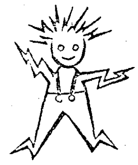
How does natural language manage abstractions?

Abstract, formal reasoning.



Formal reasoning is based on direct representations of the individual system variables and relations. The model is structured in functions; the components are dissociated into sets of relational rules – they become transformations connecting variables – which are the 'objects' of data processing.

fig. 79



Natural language reasoning controls object manipulation. Are abstract signs & sympols treated like artificial physical objects?

Uden at det skal tages alt for bogstaveligt, kan man forestille sig, at elektricitet består af nogle små fyre, der hedder elektroner (bestanddele af atomet), og de farer – billedligt talt – af sted i ledningen, når vi bruger strøm. Når vi ingen strøm bruger, må vi tænke os, at de har lagt sig til at sove. Vi kan til illustration forestille os et elektron således (*fig. 79*).

Når der løber strøm gennem ledningen, ser det således ud:

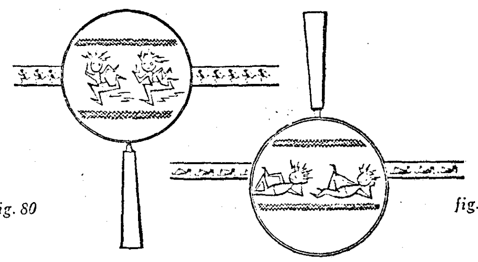


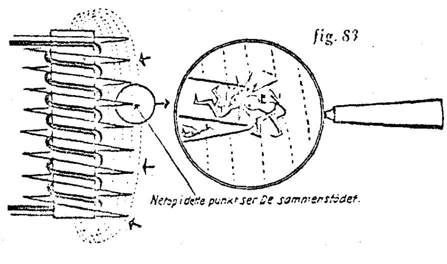
fig. 80 fig. 81

Når der ikke bruges strøm, sover de (*fig. 81*).

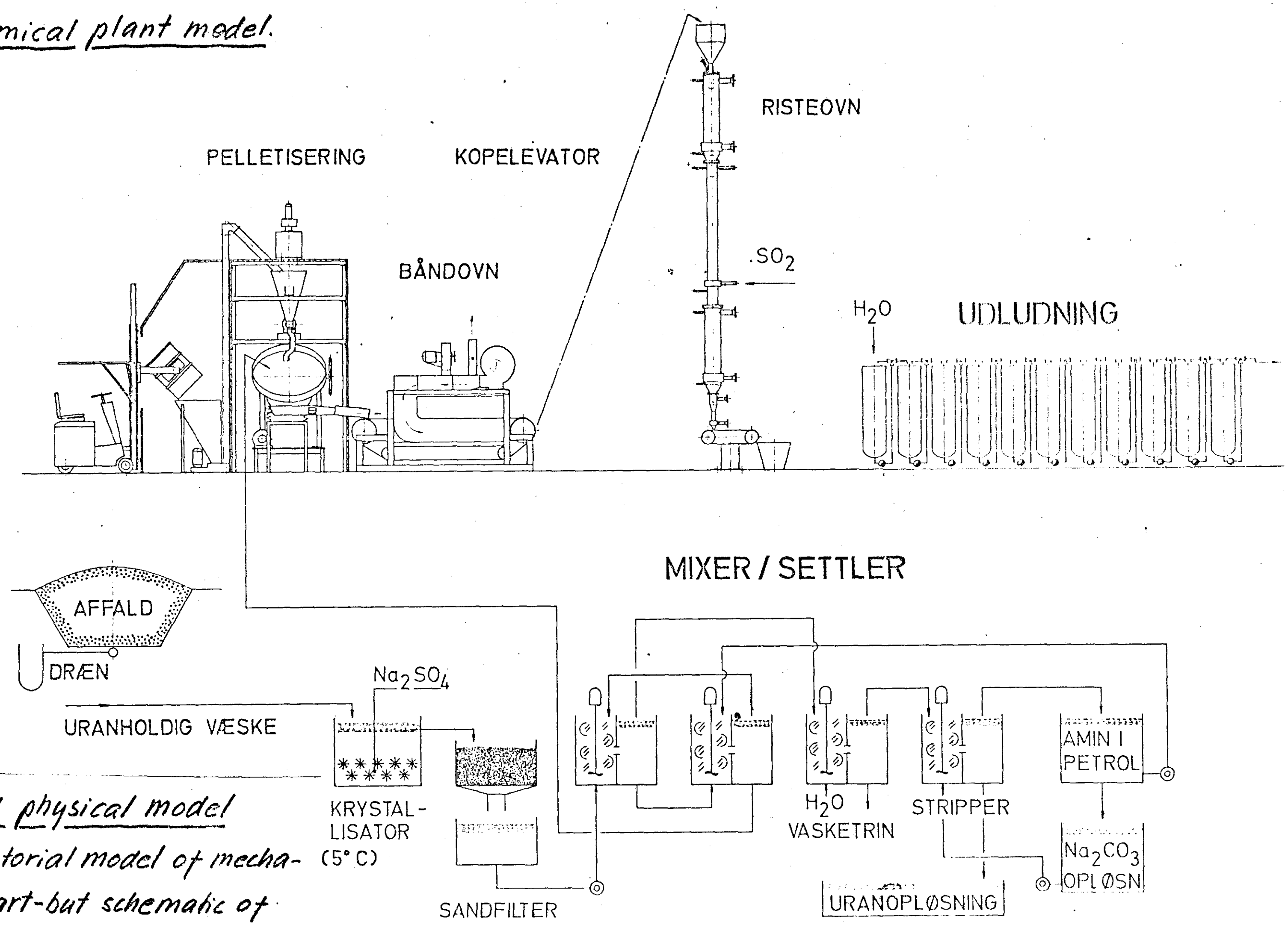
Are the artificial objects having complex states and properties and are they capable of acting?

Fonctional model of ignition coil supposed to support natural language understanding.

fig. 83

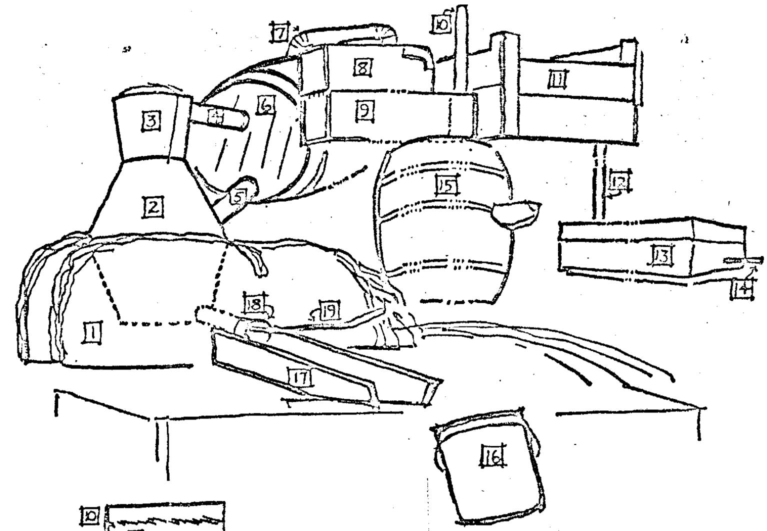


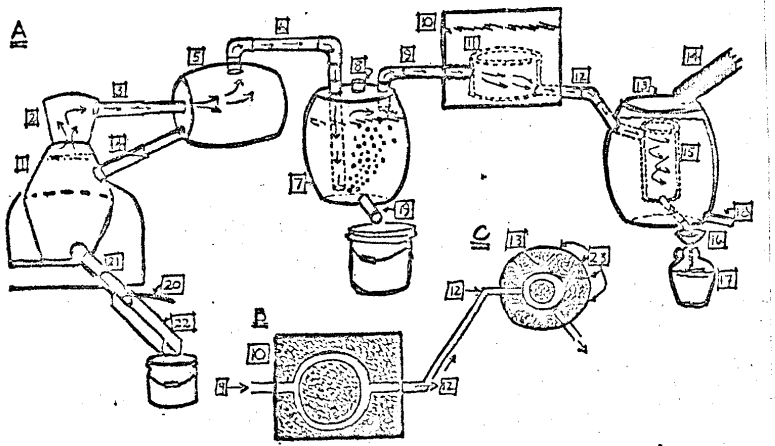
Chemical plant model.

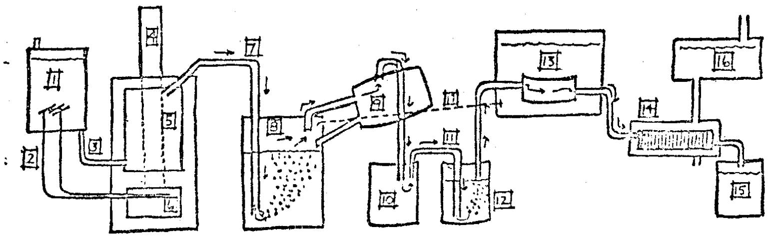


Hybrid physical model

Why pictorial model of mechanical part-but schematic of chemical part?







Physical models of moonshine still.

Different degree of schematization.

Any influence on natural data process?

Procedure

$42_{10} \rightarrow 101010_2$; $47_{10} \rightarrow 101111_2$.

Set of symbol manipulation rules for specific task, -data representation conversion-.

CONVERSION OF DECIMAL TO BINARY

There are two commonly used methods for converting decimal numbers to binary equivalents. The reader may choose whichever method he finds easier to use.

1. *Subtraction of Powers Method*—To convert any decimal number to its binary equivalent by the subtraction of powers method, proceed as follows.

Subtract the highest possible power of two from the decimal number, and place a "1" in the appropriate weighting position of the partially completed binary number. Continue this procedure until the decimal number is reduced to 0. If, after the first subtraction, the next lower power of 2 cannot be subtracted, place a 0 in the appropriate weighting position. Example:

42_{10} = ? binary

$$\begin{array}{r} 42 \\ -32 \\ \hline 10 \end{array} \qquad \begin{array}{r} 10 \\ -\ 8 \\ \hline 2 \end{array} \qquad \begin{array}{r} 2 \\ -2 \\ \hline 0 \end{array}$$

2^5	2^4	2^3	2^2	2^1	2^0	Power
32	16	8	4	2	1	Value
1	0	1	0	1	0	Binary

Therefore, $42_{10} = 101010_2$.

2. *Division Method*—To convert a decimal number to binary by the division method, proceed as follows.

Divide the decimal number by 2. If there is a remainder, put a 1 in the LSD of the partially formed binary number; if there is no remainder, put a 0 in the LSD of the binary number. Divide the quotient from the first division by 2, and repeat the process. If there is a remainder, record a 1; if there is no remainder, record a 0. Continue until the quotient has been reduced to 0. Example:

47_{10} = ? Binary

		Quotient	Remainder
$2\overline{)47}$	=	23	1
$2\overline{)23}$	=	11	1
$2\overline{)11}$	=	5	1
$2\overline{)5}$	=	2	1
$2\overline{)2}$	=	1	0
$2\overline{)1}$	=	0	1

1 0 1 1 1 1

Therefore, $47_{10} = 101111_2$.

Procedure

Physical Model representing location of input-output devices (sources of information)

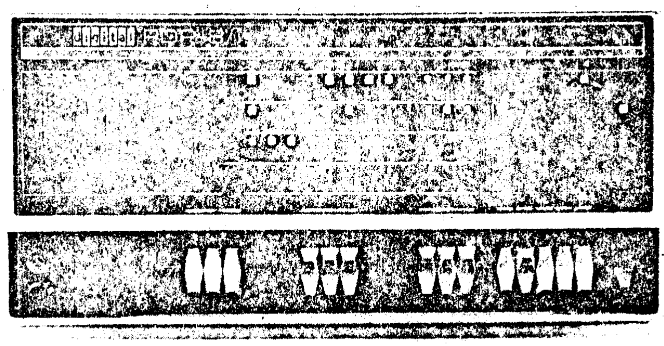
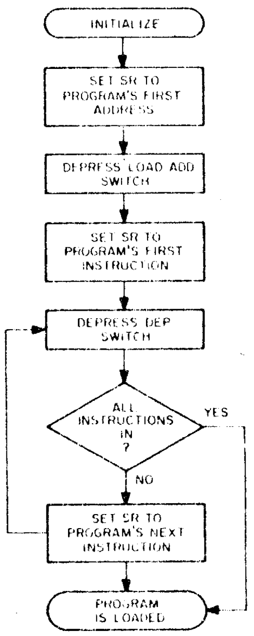


Figure 4-1. PDP-8/I Computer Console



CONSOLE SWITCH POSITIONING

PDP-8/I When the top of a switch is out, it represents a binary 1 and is considered set; conversely, when the bottom of the switch is out it represents a binary 0 and is not set.

INITIALIZING THE CONSOLE

1. Computer POWER is on.
2. PANEL LOCK is off.
3. SWITCH REGISTER equals zero.
4. Both SING STEP and SING INST are not set.
5. All peripheral devices turned off.

Figure 4-3. Manually Loading a Program

Procedure - sequence of actions referring to the physical model for a specific task.

Procedure & Task sequence

Procedure:

Sequence of actions expressed with reference to a physical system.

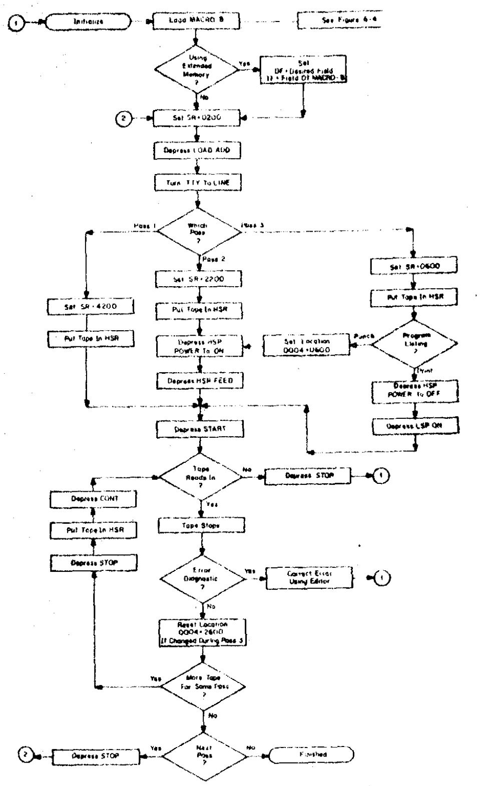
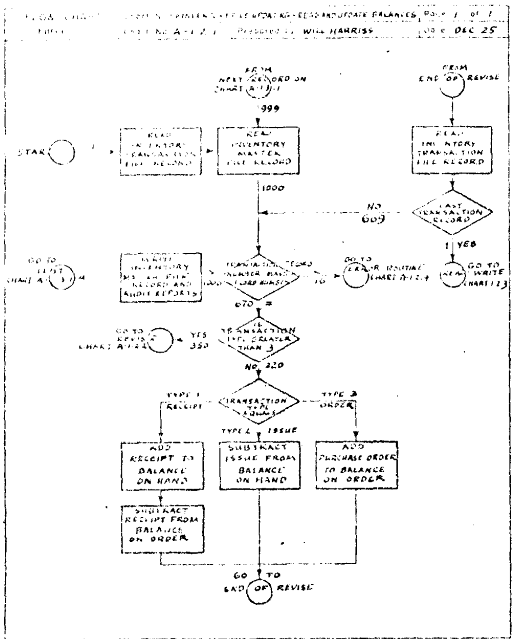


Figure 6-10. Assembling a MACRO-8 Symbolic Program Using the High-Speed Reader/Punch

6-37



'. Procedure for file reading and balance updating.

Task-sequence

Break-down of task into subtasks or transactions'. Sequence expressed with reference to objectives', not to physical system.

Model transformation

Physical model

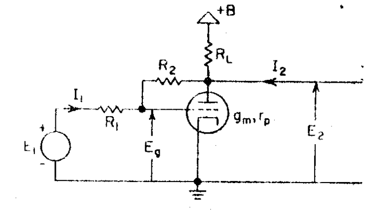


FIG. 29-11. A triode amplifier with plate-to-grid feedback.

Schematic, functional model

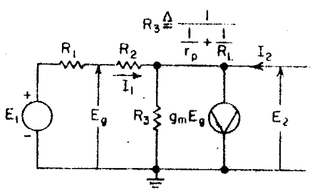


FIG. 29-12. Linear incremental model of the triode amplifier.

Generalized, functional model

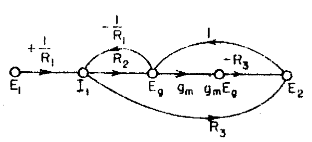


FIG. 29-13. Flow-graph interpretation of the model of Fig. 29-12.

Procedure, rules for model transformation:

related to model convention, not task specific

Step-by-step Reduction. Before presenting the reduction formula for the evaluation of a transmittance of a graph by inspection, it is useful to consider some reduction techniques which can be applied step by step. Algebraic reductions (such as the elimination of dependent variables, for example) have their flow-graph counterparts, and several elementary simplifications suggest themselves as a result of the properties of the nodes and the assumption that the graphs are linear.

In general, it is useful to think of graph reduction or simplification as involving the absorption of nodes or the elimination of loops or both. We shall first consider the reduction of single loops and then present a comprehensive step-by-step procedure for the absorption of any desired group of dependent nodes, with the end result, if desired, the reduction of the graph to a single branch. The transmittance of this final branch is, of course, the graph transmittance or transfer function being sought.

An elementary single-loop graph may be reduced to a single branch by dividing the forward path transmittance by one minus the loop transmittance. This simple rule, which is illustrated in Fig. 29-15, may easily be derived by reconstructing the set of equations which the graph represents and solving these for T. Thus, for $x_1 = 1$,

$$T_{12} = \frac{x_2}{1} = P\frac{1}{1-L} \qquad (29\text{-}24)$$

This simple rule for a single loop is applied to any paths which share nodes in common (i.e., touch) the loop.

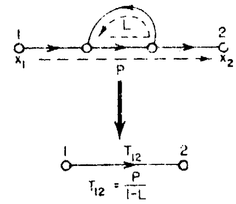
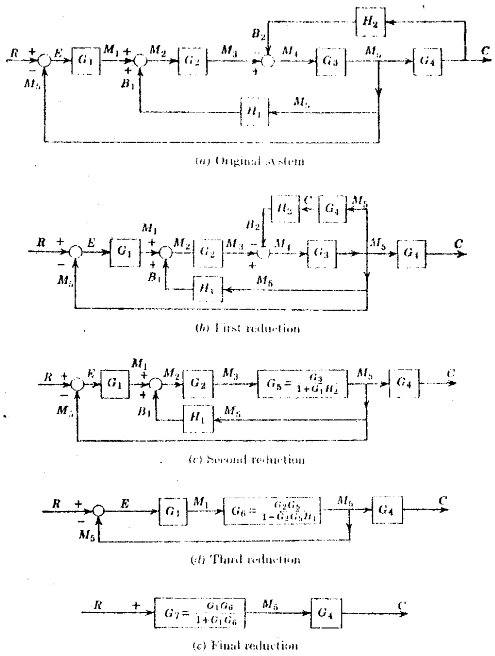


FIG. 29-15. Reduction of a single loop.

Model transformation

Functional model.
schematic, structured in functions, no reference to components.



Algebra of Block Diagrams

A complex block diagram can be rearranged or reduced by combining blocks algebraically. When all the loops are concentric, the indicated manipulations can be carried out directly by successively applying the relation $C/R = G/(1+GH)$ to the innermost loops. When the inner loops are not concentric or even intertwining, the block diagrams can usually be reduced to concentric loops by the following *rules* and by reference to Table 17.

1. Data takeoff channels can be moved forward (in the direction of arrows) or backward in the system at will except that the takeoff point cannot pass a summing point. Whenever a data takeoff branch is moved forward past a function G, the function $1/G$, must be added in series with the branch. Whenever a data takeoff branch is moved backward past a function G, the function G, must be added in series with the branch.

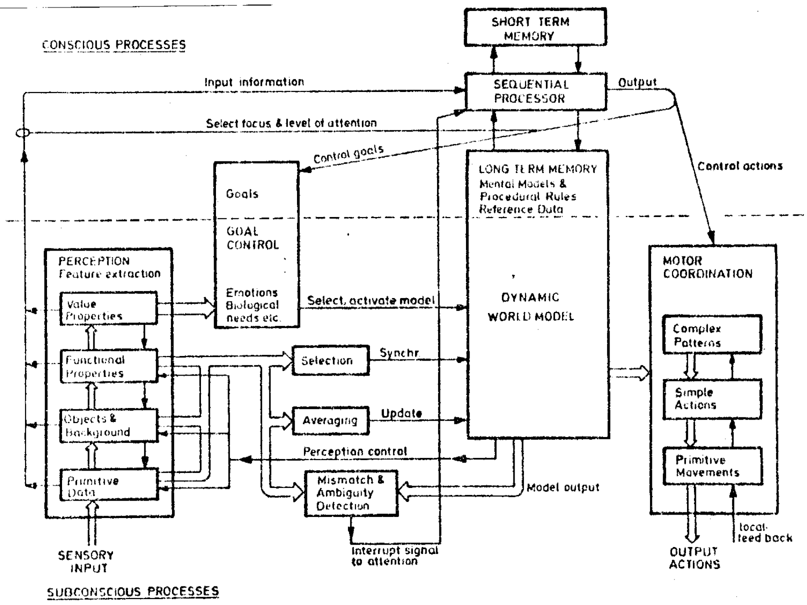
2. A channel feeding into a summing point can be moved forward and backward in the system at will except that it cannot pass a data takeoff point. As this feed channel is moved forward in the system past a function G, the function G must be added in series with the channel. As it is moved backward past a function G, the function $1/G$ must be inserted in the channel.

3. In some cases, it will be found necessary to move a takeoff point past a summing point or a summing point past a data takeoff point in order to reduce the system block diagram to simple concentric loops or parallel paths, which can be handled by methods (1) and (2). This can be done by removing a troublesome feedback point or data takeoff by closing an internal loop, thus replacing a loop by a closed loop transfer function, which has no takeoff or feedback points.

<u>Procedure</u> for transformation of model related to model conventions.

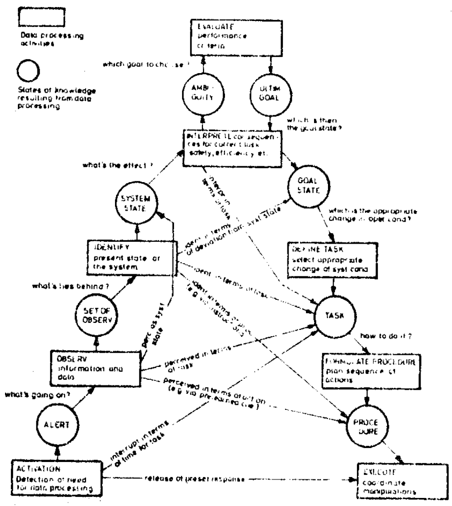
Schematic functional model

structured in functions, no reference to physical components.



State transition model

Specify transactions linking states of data in "segential data processor"



ANALYSIS OF MENTAL ACTIVITY

Due to the subjective formulation of performance criteria and goals and the large repertoire of different subroutines for a specific task available to a skilled person, the sequence of data processes performed in a specific situation will have a detailed structure and content controlled **by** very person- and situation-dependent features.

Even when it is possible to obtain detailed verbal protocols from a single task sequence, it will be very difficult to ident**ify** general features of the data processes due to frequent spontaneous shifts between different basic process strategies initiated **by** minute details in the situation. The trees obscure the sight of the forest.

A simultaneous analysis of several process descriptions (e.g. from verbal protocols) at the level of types of data, models and procedural rules support a more efficient identification of consistent strategies. This leads to a description in terms of information transformations between different states of knowledge, in the form of an information flow map rather than a sequence description.

When this analysis has led to the fundamental structure of different strategies used in a mental subtask it may be possible logically and rationally to complete the description of a strat**egy** and to generate a full set of possible strategies. In this way a description, what the operator can do **is** obtained.

The strategy maps enable the formulation of precise questions to be used for an analysis of the sequence in a specific protocol in terms of the information extracted from observations, the type of models and operations used. Thus a detailed analysis can lead to identification of leaps between formal strategies.

A description of what the operator will do in a specific situation can only be obtained in terms of a description of cues in a situation which may lead to leaps between operations belonging to different strategies. Hopefully, these can be connected with some indication of the transition probability.

If data processing is based upon simulation by an active behavioural model, the data process can not be described by a linear sequence, but only in terms of the structure of the model and the properties of the element. Semantic linguistic analysis of the expressions used in verbal descriptions can possibly serve this purpose.

AN ILLUSTRATIVE EXAMPLE

To test this analytical approach the task of trouble shooting in an electronic maintenance shop has been chosen. The reasons were primarily due to the availability of detailed verbal protocols and to the well specified task.

In a diagnostic task aiming only at repair of a system, the task is primarily to identify the location of a faulty component in the system. This task does not necessarily include considerations of the functional consequences of the fault which, fx, typically will be the case in process plant diagnosis where protection of the system is of primary importance.

Thus the task is to derive in some way a topographic reference to the location of the faulty component. This can be done in several ways leading to different formal search strategies. Basically, the task is an inductive one but, as the system is known to have been operating satisfactorily, the task is solved by a search for a change from normal conditions.

Three different classes of strategies can be used. The information extracted from observations giving a reference to the location of the faully component is different in these strategies, as is the use of different types of models. This is illustrated by the schematic information flow maps.

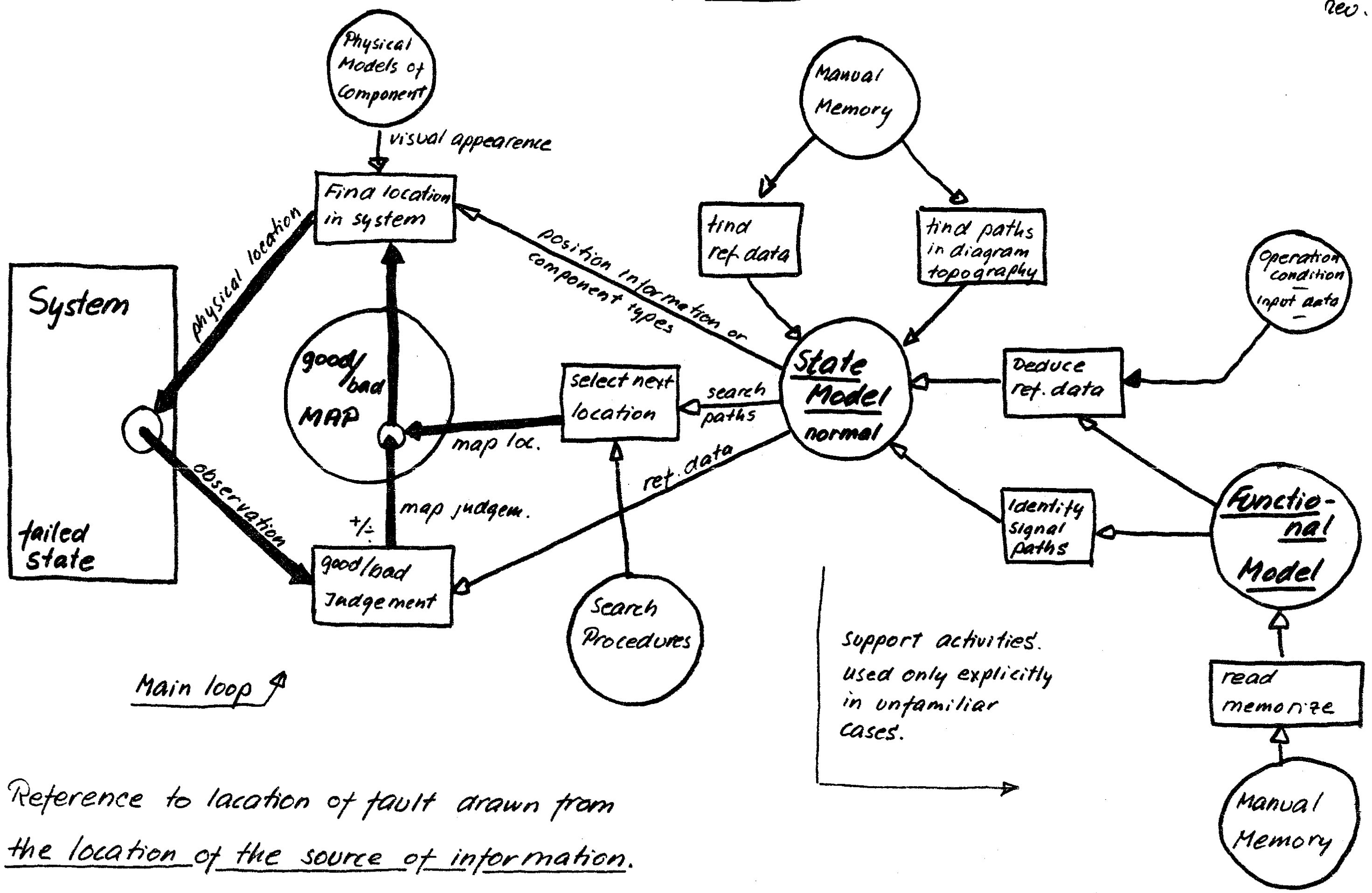
In a specific diagnostic task, frequent leaps between the different strategies will take place. All different strategies are latently present and available, and difficulties during a current activity and/Qr indications that another strategy may be more promising will initiate a leap, which is controlled by very person- and situation-dependent features.

Furthermore, the sequence in which the different subactivities of a strategy are performed is very dependent on details in the specific case.

Rather than a linear sequential description of the sequence of activities during specific cases, the analysis should aim at a description of the activities and their role in a set of formal strategies, together with a listing of typical cues and heuristic rules controlling the leaps between strategies.

18/376 R
rev. 5/4.

Search by topographic good/bad mapping.



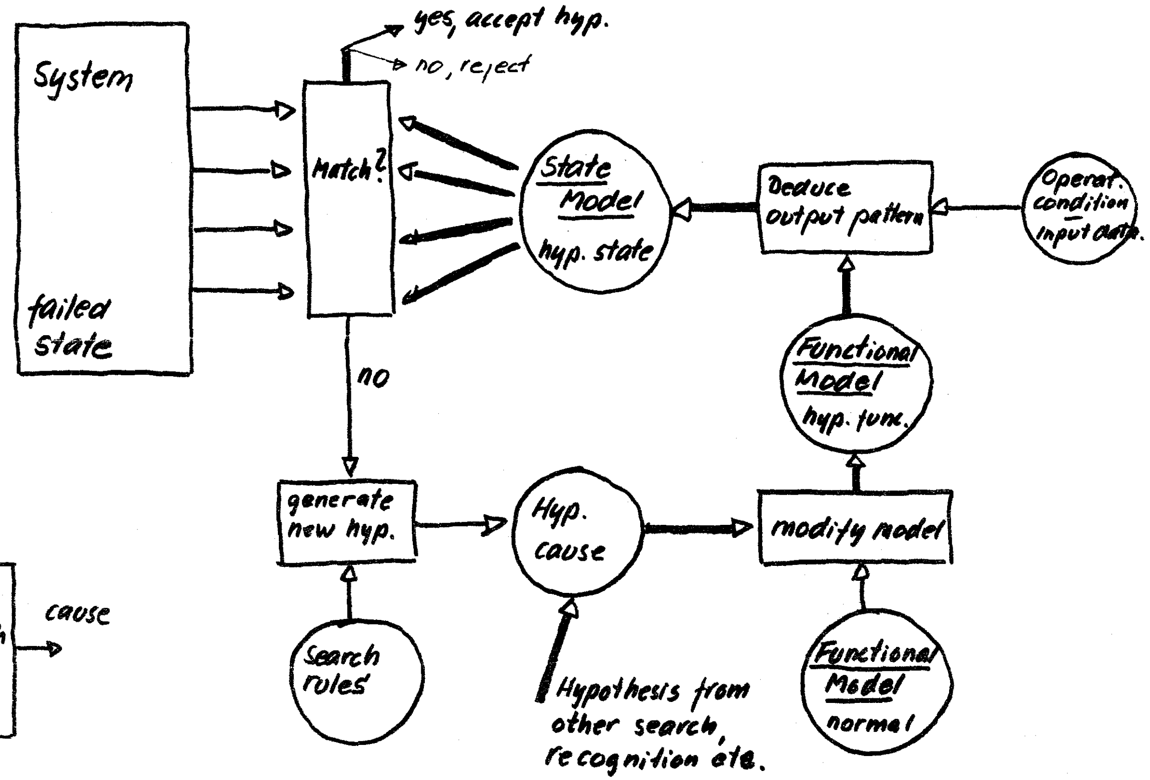
Reference to location of fault drawn from the location of the source of information.

Observations only used for good/bad judgements.

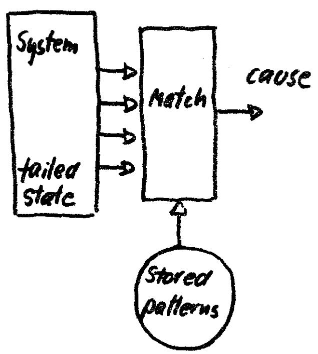
18/3/76 R
Rev. 5/4.

Search by Data pattern Matching

Hypothesis and conceptual test



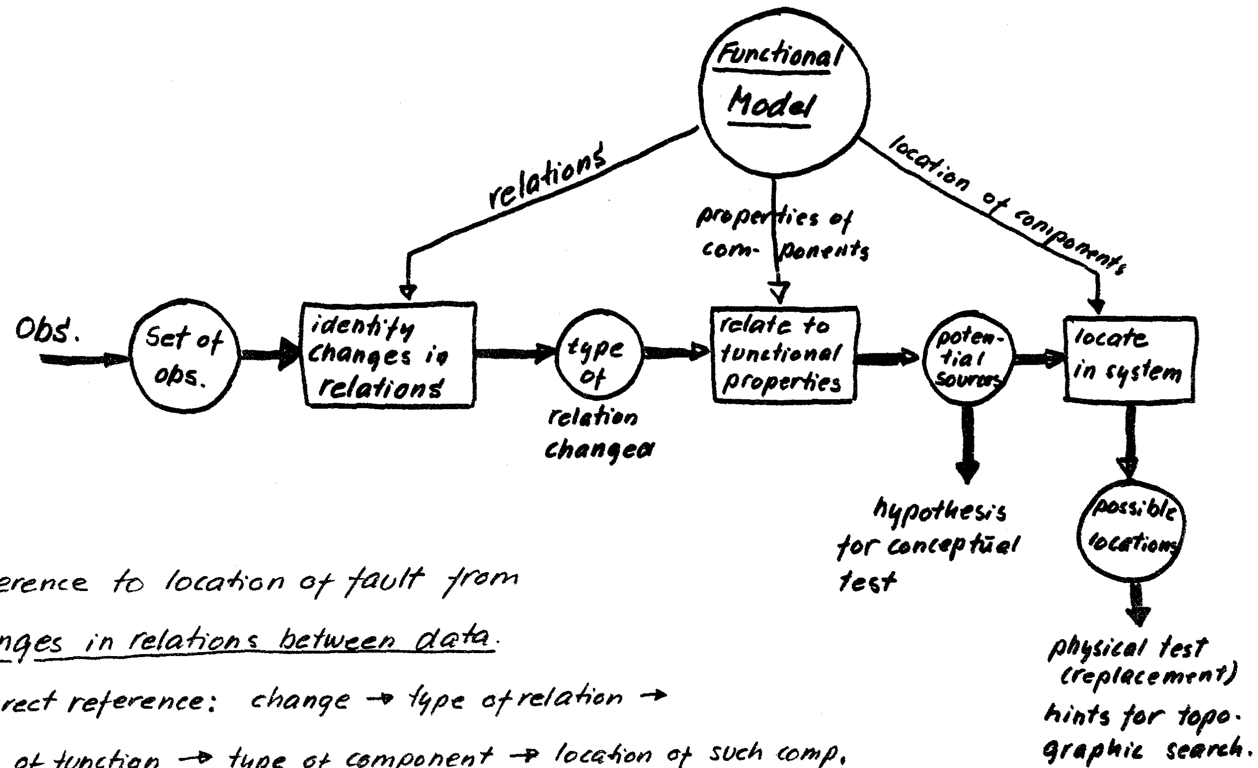
Recognition



Reference to location from data value patterns labelled in causes.
Search by hyp & test depend on suitable rules for generating hyp. Generally the strategy is used for once-through test of a single hyp.

18/3 76
Rev. 5/4

Search by evaluation of functional relations



Reference to location of fault from

changes in relations between data.

Indirect reference: change → type of relation →

type of function → type of component → location of such comp.

No search loop! Typically used to support other search strategies.