

CAI 学習プログラム記述言語 TEACHTRAN の開発

TEACHTRAN: Hitachi CAI Language

平 野 睦 房*
Chikafusa Hirano

津 田 順 司*
Junji Tsuda

中 村 襄**
Noboru Nakamura

要 旨

CAI 言語は教育用ソフトウェア(学習プログラム)を記述するのに便利のように工夫された、一つの問題向き言語である。したがって、CAI で実施しうる教授機能に対応して言語のステートメントが定められているわけで、いわば CAI 教授機能の要約表現である。今回試作した CAI システムでは TEACHTRAN と名づけた CAI 言語を開発し適用した。TEACHTRAN 言語は既存の標準的 CAI システムに比べて融通性の高い学習プログラムを受け入れることができ、種々の教育手順を試用する教育の研究にも利用することが可能であると考えられる。本言語の仕様提案の背景には CAI における教授・学習活動の流れについての分析と提案ならびに小形計算機で時分割サービス可能な最大限の機能についての検討がある。

TEACHTRAN は国内の CAI 標準言語第 1 次案を受け入れうるように考慮されており、学習プログラムの共同利用と蓄積に貢献しうることを期待している。

1. 緒 言

今回開発された CAI (Computer Assisted Instruction) システムは、教育の研究用にも使用することを考慮しているので、小形計算機を中心としていながらも、別題で述べているように融通性の高い学習機能を持っている。つまり、教材を提示する手段、方法について、学習者の反応、応答のしかたの自由度について、また学習者の応答を得てどのように評価し、分岐するか論理について、既存の CAI システムで実施されている以上の自由度ある学習プログラムを受け入れられる。

この機能は一方で、学習プログラム作成者が上記の自由度を自在に使いこなしてこそ、その真価を発揮しうるものであるから、このような学習プログラムを容易に記述できる方法手段が用意されなければならないことになる。学習プログラム記述言語はこのような目的で開発されたもので、CAI システムがもっている教授機能がこの言語体系の中に要約されていると考えることもできる。

2. CAI による学習

CAI システムにおける学習者の学習は、次のような計算機と学習者のやりとりによって進行する。

(1) まず計算機が学習者に問題や注意を与える。具体的には質問の書いてあるスライドを投映したり、CRT ディスプレイ画面に注意を印字したりする。その後、計算機は学習者の応答を待ち続ける。

(2) 次に学習者が計算機に応答する。学習者は提示された問題や注意に対して解答や「わかった」という了解の応答をする。すなわち学習者の手もとにあるキーボードのタイプライタと応答キーを押して計算機に返答をする。

(3) 学習者の応答を受け取った計算機は応答の内容を直ちに判断して学習者のために適切な問題や注意を選んで再び与える。

以下このようなやりとりを繰り返すことによって学習者個人のペースで学習が進む。

以上の過程における計算機の動きについて概説すると次のとおりである。

計算機の動きは図 1 のように三つあるいは四つのプロセスの繰り返しである。

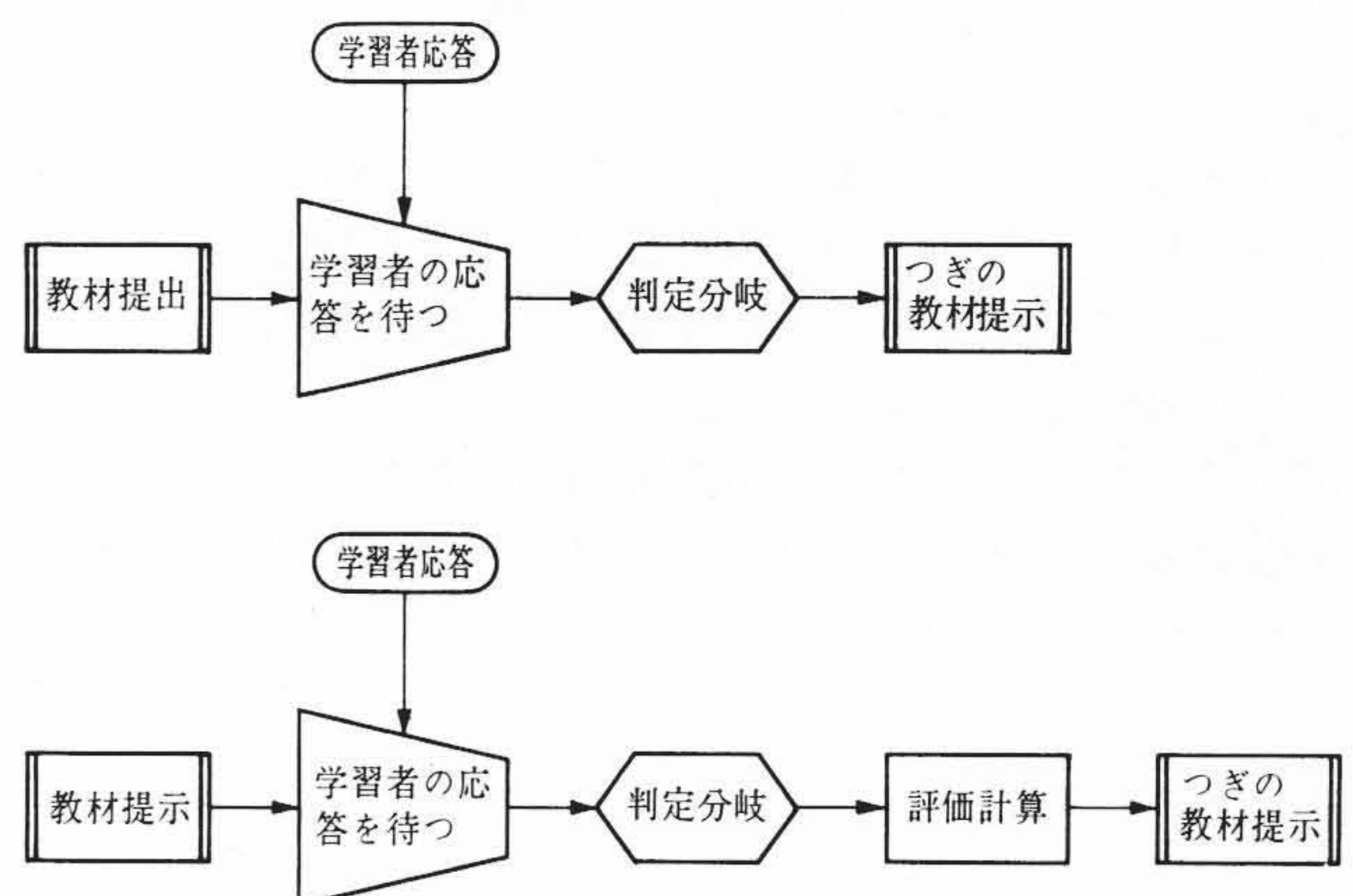


図1 CAI学習者と計算機の動き

ここで「教材提示」とは、学習者に問題や注意をスライドや印字や音声によって与えることであり、「判定分岐」とは学習者応答の内容や過去の学習過程を評価判断し、学習者のために適切な教材を選ぶことである。また「評価計算」とは学習者の得点や学習過程の評価を行なうことである。計算機は図1のように学習者に教材提示を行ない、学習者応答待ちの状態になる。そこへ学習者からの応答が受信されると、その内容を判定し(評価計算を行ない)、判定の結果に従って適当な教材提示を行なう。

さて計算機は上述のように動作するが、その手順つまりプログラムを作成したのは教授者である。プログラムは TEACHTRAN (学習プログラム記述言語) によって記述される。TEACHTRAN には幾つかのステートメント (Statement) と呼ばれる命令語があり、そのおもなものは教材提示を命ずるステートメント、学習者応答条件を宣言するステートメント、判定分岐を命ずるステートメント、評価計算を命ずるステートメントである。

このように学習の手順を表わすプログラムを学習プログラムというが、これは大別して三つの種類がある。まず中心となる学習プログラム、学習者にヒントを与えるために作られたプログラム、そして第3に参照資料や補助的な教材のプログラムである。本マニュアルではこれらをそれぞれ main sequence, hint sequence そして call sequence と呼ぶ。こうして教授者のプログラムに従って計算機は教材提示を行ない、学習者の応答の内容を分析して(評価計算して)適当な教材提示へ学習者を導く。

* 日立製作所中央研究所

** 日立電子株式会社

ここで本システムにおける学習者応答の自由度について述べておこう。

まず学習者の手もとにあるキーボードには、タイプライタと同様のキーボードのほか次の 8 個の応答キーが並んでいる。

- (1) ANS キーと NEXT キー： 問題が提示された場合は、解答メッセージをタイプしてそれが終わってから ANS キーを使用する。説明が提示された場合は次の提示へ進むことを要求するために NEXT キーを押す。
- (2) HINT キー： ヒント要求をするためのキー
- (3) CA キー： 正答要求をするためのキー
- (4) REV キー： 前回提示に戻るためのキー
- (5) CALL キー： 計算機を呼び出したり、参考教材を呼び出したりすることを要求するキー
- (6) OK キーと RETURN キー： hint sequence から元の main sequence へ戻るためには OK キーを使用する。また call sequence から元の main sequence へ戻るときには RETURN キーを使用する。

まず提示された教材が問題である場合には、通常学習者はそれに対して解答を行なう。タイプライタから解答メッセージをタイプし、それが終わって ANS キーを押すと、これが初めて解答として計算機に読み取られる。ANS キーを押す以前ならば、学習者は納得するまで任意に解答を修正することができる。学習者はまた要求があった場合、ライトペンを使って CRT ディスプレイ面画面上の点をプロットすることによって解答する。計算機は解答を受け取ると学習プログラムに従って学習者の応答内容を判定して、適当な教材提示を行なう。

提示された教材が説明や注意書きであるならば、問題の場合と異なり学習者は解答する必要はない。学習者は説明内容をよく理解したら NEXT キーを押すと学習プログラムで次に定められた提示へ進む。

また学習者が解答を見いだせない場合に備えて hint sequence を用意しておくことができる。もし学習者が HINT キーを押してそれを要求したならば、学習プログラムに従って hint sequence に学習者を導くことができる。

同様に学習者が CA キーを押してこれを要求したときに、学習プログラムによって正答提示を行なわせることができる。この場合は、学習プログラムの中で、正答が用意されていなければならない。hint sequence が終了したとき、あるいは学習者が途中で了解に達し、OK キーを押したときは、CAI システムが hint sequence にはいる以前の main sequence の提示に戻る。そこで学習者は解答を行ない、以後 main Sequence に沿って学習が進む。

HINT キー、CA キーは学習者に乱用されると教育上悪い効果を生むおそれもあるので、教授者は学習プログラムでその要求を許すかどうかを指定することができる。これが許されていないとき、学習者が要求した場合には、システムによって自動的にその旨を学習者に告げて、学習者が自分自身で考えることを要求する。

また学習プログラムでは学習者の応答時間を制限することができる。この場合、学習者が制限時間内に答えられなかったときは、学習プログラムに従って次の提示へ進む。

学習者が学習中に参考教材や補助教材を要求した場合には、学習者はその教材名をキーボードでタイプしてから、CALL キーを押すとその教材が提示され、そこから学習者の学習は call sequence を進むことになる。学習者は call sequence を進んでいる最中でもほかの call sequence を呼ぶことができる。call sequence から元の

表 1 TEACHTRAN ステートメントの種類

Statement	説 明
Presentation Statements	学習者に提示する教材の内容を指定するステートメント
	学習者の応答条件を規定するステートメント
Assignment Statements	学習者の学習経過を記憶したり、評価したりするための算術演算や論理演算を行なうためのステートメント
Branching Statements	プログラムの流れを任意に変えるためのステートメント
	条件判定を行ない、プログラムの流れを分岐するためのステートメント
INSERT Statement	同一のプログラムを異なったところで何度も繰り返し使うような場合、これの一つ独立したプログラムとして扱い必要に応じてそう入して使う方法がある。このようなプログラムをそう入するために用いるステートメント
Miscellaneous Statements	プログラムの制御およびその他の雑目的を行なうためのステートメント

提示に戻るには RETURN キーを押せばよい。

REV キーは一つ前の提示に戻ることを要求するためのキーで学習者が前の提示で出された説明などをもう一度見たいと思うときに使うことができる。

以上の説明を要約すると次のとおりである。

- (1) CAI システムにおける学習は計算機と学習者とのやりとりによって進行する。つまり計算機が学習者に問題や注意を与える。すると学習者はそれに対して応答する。その応答を受け取って計算機は内容を判定し、必要なら評価計算を行なって学習者のために適切な教材提示を行なう。以後これを繰り返す。
- (2) 上記のようにシステムを動かすには学習プログラムと呼ばれる文章を書く必要がある。
- (3) プログラムには main sequence, hint sequence そして call sequence の 3 種類があり、一つのプログラムは何種類かのステートメントの組合せから成っている。
- (4) 学習者の応答はキーボードのタイプライタと 8 種類の応答キーを押すことによって行なわれる。

3. TEACHTRAN ステートメント

3.1 概 要

上記のように、学習プログラムは TEACHTRAN のステートメントによって記述されるが、特に学習の流れに関係の深いものは次の 3 種である。

- ① Presentation Statements
- ② Assignment Statements
- ③ Branching Statements

①のステートメントには教材を指定し指示を行なわせるステートメントと、判断分岐の際判定の規準になる学習者応答条件を宣言するステートメントとがある。②のステートメントは評価計算を行なうステートメント、③のステートメントは判定分岐を行なわしめるステートメントである。このほかにプログラムを組みやすくするために用意された INSERT Statement, 計算機の制御その他を行なう Miscellaneous Statements などがある。これらのステートメントを表にまとめると表 1 のようになる。

これらのステートメントを組み合わせて学習プログラムを作成するわけであるが、作成する前にプログラムの内容をフローチャートと呼ばれる流れ図に書いておくでプログラムの作成を容易にするばかりでなく、学習の流れをつかみやすい。たとえば「スライド 20 番を提示し、解答が 25 ならば正答とみなし Yes に分岐し、そうでないときは No に分岐してもう一度スライド 20 番を提示する」という学習の内容をフローチャートに直すと図 2 のようになる。

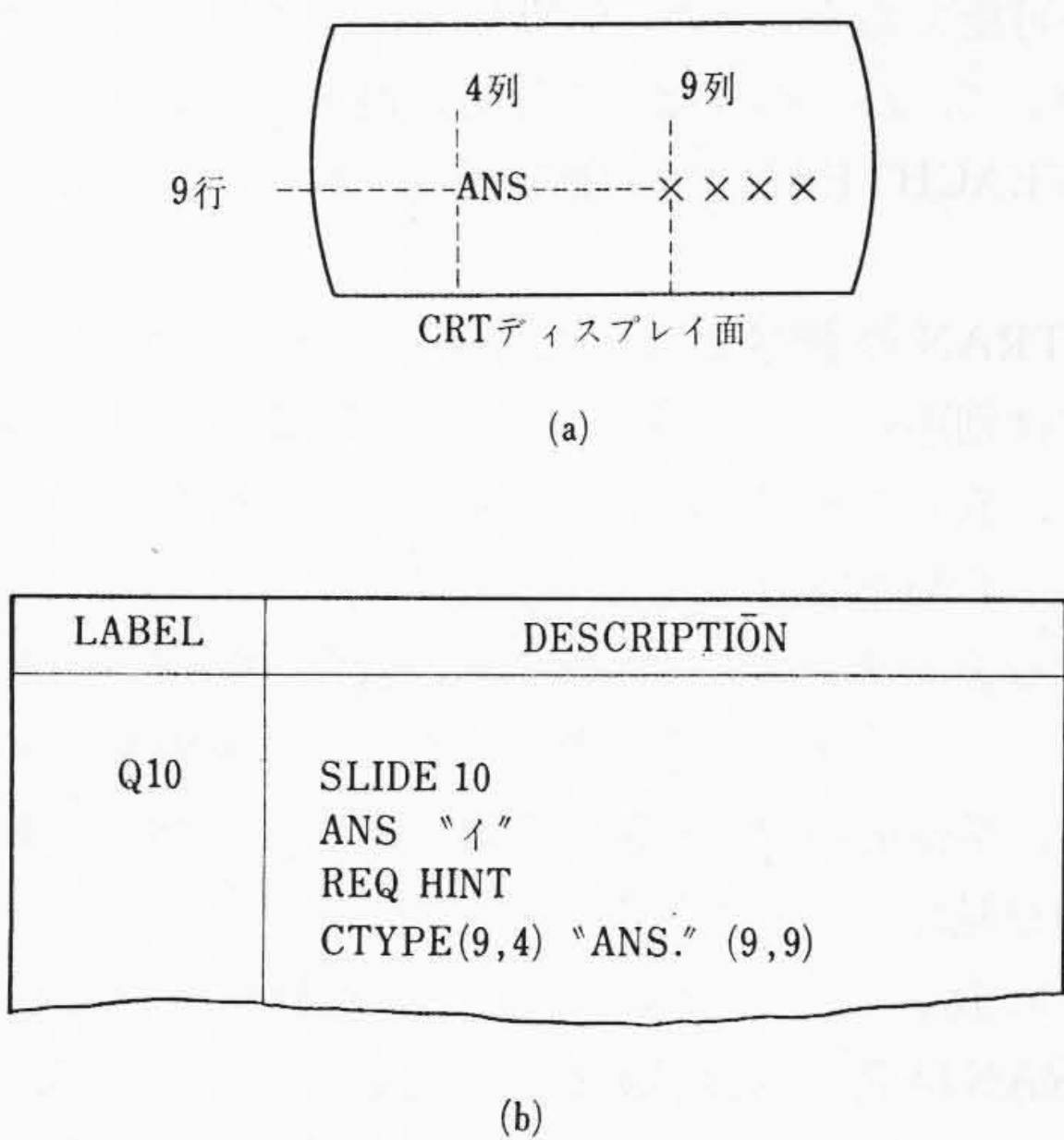


図5 Presentation の例

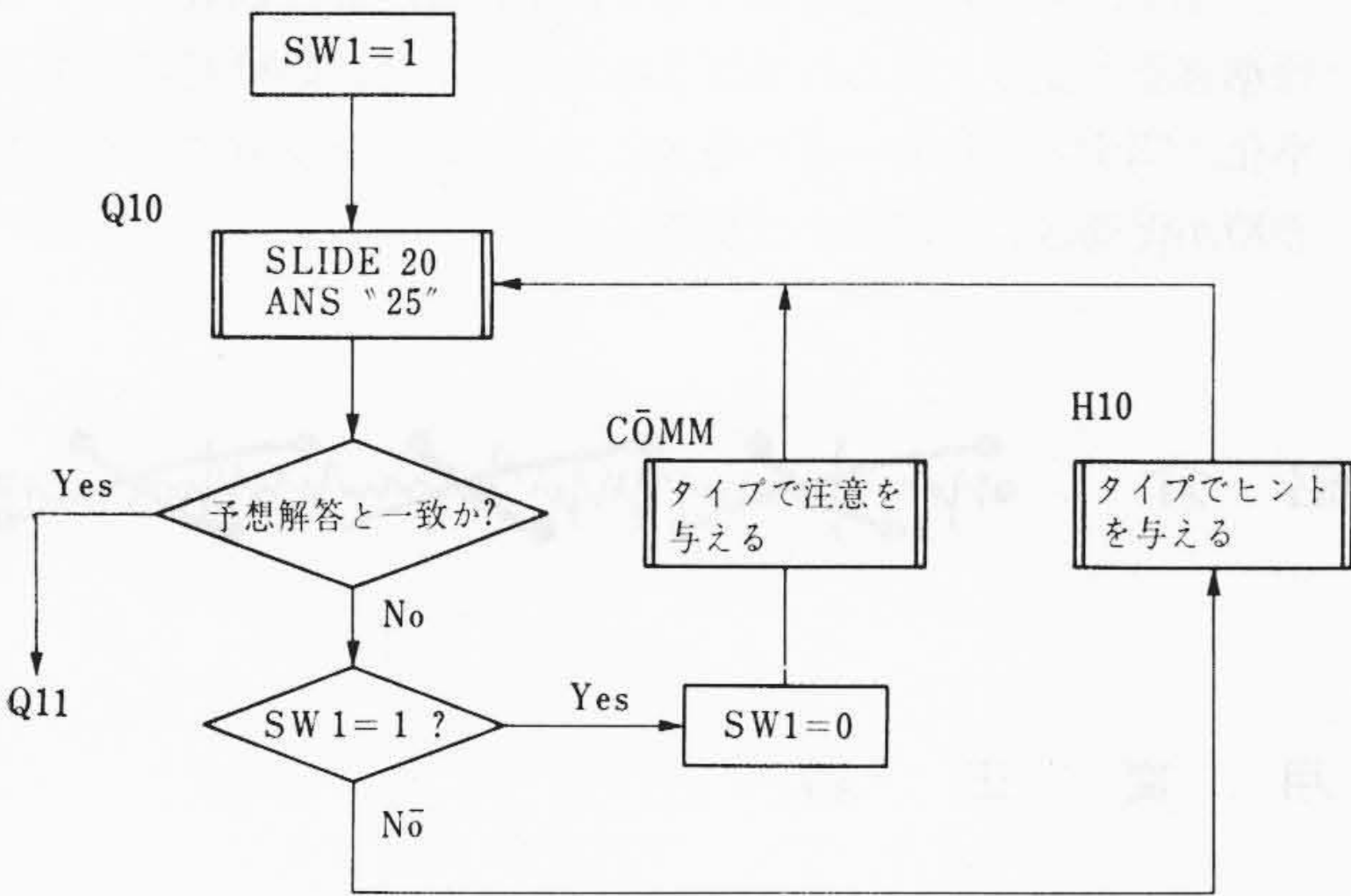


図6 学習プログラムの例

LABEL	DESCRIPTION
Q10	SW1=1 SLIDE 20 ANS "25" IF ANS1 THEN Gō Tō Q11
H10	IF SW1 THEN SW1=0, Gō Tō C0MM TYPE "5×3=15, 5×4=20 デスネ。ヨクカンガエテコタエナサイ" TIME 1 SEC Gō Tō Q10
C0MM	TYPE "チガイマス。モウイチドコタエナサイ。" TIME 1 SEC Gō Tō Q10
Q11	

図7 図6に対応するプログラム

点から成る第1象限を用いる。プログラムで使用可能な点は格子点で、その座標 (x, y) はシステムのハードウェアで定められる範囲の整数である。座標を $()$ でくくって列記すれば、それらの点をその順序で結ぶ折れ線を CRT ディスプレイ画面上に表示する。
(4) TIME 1 SEC と記述すると提示に対する学習者の応答時間を 1 秒に制限する。ただし 1 は 0 および正整数である。このステートメントが記述されない場合は、学習者からのなんらかの応答がなければプログラムは先に進まない。学習者応答時間の起点

表4 Assignment ステートメント一覧表

Assignment Statements	説明
$SW_n = 1$ (または 0)	スイッチ n をオン(またはオフ)にする(スイッチは状態を記憶するために使用)。
$SW = 1$ (または 0)	すべてのスイッチをオン(またはオフ)にする。
$ACC_n = +B$	右辺の計算結果を左辺へ入れる (ACC は回数、得点などをカウントするために使用)。
$ACC_n = -B$	
$ACC_n = \pm B + C$	

注: B, C は整数または ACC_n'

表5 Branching ステートメント一覧表

Branching Statement	説明
Go To label	無条件ジャンプ
IF TIME UP THEN.....	応答が制限時間内にこなかったとき
IF HINT THEN.....	ヒント要求があったとき
IF CANS THEN.....	正答要求があったとき
IF ANS_n THEN.....	学習者解答が予想解答の n 番目と一致したとき
IF ANSO THEN.....	学習者解答が予想解答外のものであるとき
IF SW_n THEN.....	スイッチ n がオンのとき
IF $ACC_n \{ \geq \} ACC_n' (または N)$	ACC と ACC_n' (または整数 N) が左の関係にあるとき

は、おおむね教材提示終了時点であるが、システムのハードウェアで任意に定めることができる。

(5) ANS ステートメントは指示された問題に対し、プログラマが予想する解答を宣言するステートメントである。このステートメントによって計算機は学習者の解答と、予想解答との文字パターンが一致しているかどうかを調べて判定分岐を行なうものである。予想解答は “ ” でくくって列記するものとし、おのおのの予想解答の間は、(カンマ)でくぎって表わす。

問題の性質によってはパターン一致方式による判定分岐が行えない場合がある。このような場合には AiDn で特別な処理ルーチン呼び出して処理を行なう。“ ” でくくった文字定数は AiD プログラムのパラメータのような働きをするわけである。

(6) 例題として下記のような場合を考える。

スライド 10 番の問題を提示する。予想解答はイである。応答時間の制限はないが、ヒント要求を許す。正答要求は許さない。学習者の解答は CRT ディスプレイ画面に ANS とタイプしたものに印字するものとする (図 5 参照)。

これをコーディングシートに書くと図 5 (b) のようになる。

3.4 Assignment ステートメント

学習者の過去の学習経過を記録するため、スイッチ (SW) と累算器 (ACC) を用意し、これを利用する。その記録方法をプログラムで任意に指定したり、その内容を判定してプログラムを分岐させたりできる。Assignment ステートメントはこれらのスイッチや累算器の内容を変更するためのステートメントである。

Assignment ステートメントには表 4 のようなものが用意されている。

3.5 Branching ステートメント

プログラムを分岐させるステートメントである。このステートメントには 2 種類あり、一つは学習者の応答結果や過去の学習経過を判定して分岐するいわば条件付き分岐 (IF Statements) である。ほかの一つはプログラマが任意に内岐をするために用いるいわば無条件の分岐 (Go To Statement) である。表 5 は Branching ステートメントの一覧表である。

IF ~ THEN..... の Statements において、~ の部分に書けるのは一覧表に示された 7 種の記述に限られる。プログラミングの際は

この中から判定分岐の内容に合ったものを選び使用する。また……の部分には Assignment Statements, INSERT Statement, Go To Statement を組み合わせていろいろな処理を書くことができる。

3.6 プログラミングの例

「1回目が誤答であるとき無条件にもう一度答えさせ、2回目以上が誤答であるときはヒントを与えて正答するまで繰り返し答えさせる」というプログラムを作ってみる。このフローチャートは図6のようになる。これをプログラムに記述すると、図7のようになる。

3.7 そのほかのステートメント

以上述べたもののほかに、サブルーチンと呼び出すための INSERT ステートメント、プログラムの終了、記録を行なう命令など雑目的のステートメントが幾つか用意されている。

4. 結 言

いうまでもなく、学習プログラム記述言語は一つの問題向き言語である。本来問題向き言語を設定するには、記述の対象となる問題の全容が詳細に把握(はあく)されているべきものであるが、本文で対象としている CAI の学習プログラムでは、CAI の試用の歴史がきわめて浅いこともあって、記述すべき対象が実際は明らかではない。つまり、学習プログラムではどのような手順が記述されなければならないのかを教育側から仕様として定義してもらうことは現状では困難である。この事実は、TEACHTRAN の提案の背景に、CAI 学

習では何が可能であるべきか、CAI 学習に要求される機能はなんであるかについて、われわれなりの分析と提案とがあり、その結果をふまえて TEACHTRAN の仕様が提案されていることを意味している。

TEACHTRAN の背景となった CAI システムのあるべき機能仕様については別題の「小形計算機による CAI システム」で述べているように、教育の研究用に利用されることも想定しているので、既存のほかの CAI 言語よりも高い機能のものになっている。特徴的な点をあげると、プリンタ、スライド、音声、CRT を用いた提示を直接ステートメントで指定できること、時間制限がプログラムでできること、学習者の要求を種々許容できることなどはほかの CAI 言語には例を見ないものである。

このように教育のニーズについての分析を含んでいる点にも TEACHTRAN 開発の意義があると考えられるものであるが、同時に、今後の試用によって明らかにされる問題点があれば、その対策を取り入れて言語体系の洗練を図るべきであると考えている。

先般、電子工業振興協会が幹事機関となって、国内の CAI 標準言語第一次案が判定された。TEACHTRAN はこの CAI 標準言語をじゅうぶん受け入れられるものであり、日立製作所の CAI システムでは標準言語を受け入れられるようになっている。CAI 言語の開発と標準化に関するこれら一連の努力によって、学習プログラムの開発と蓄積が促進されれば幸いである。



新 案 の 紹 介



登録実用新案 第848527号

小 沢 実

揚 水 発 電 所 用 変 圧 器

近年、ピーク電力需要および余剰電力の平均化のため、大容量の揚水式発電所が各所に建設されてきている。この揚水発電所に設置される変圧器は、低圧側の巻線を Δ 結線し、揚水時に発電機を電動機として駆動し起動時に約 $1/2$ の電圧を印加するため低圧側巻線に内接 Δ 結線の端子を有するように成される。これら変圧器の低圧側引出端子は JEC-120 の国内規格や類似の外国規格のため、相回転順に配置せねばならない。しかしながら、単に低圧側の半電圧および全電圧端子を相順に配置しそれぞれ設けられる二つの切換器を用いて切換を行なうとする場合、半電圧端子と全電圧端子との間には、相順で内部電気角が 300 度の差があるので、切換時に発電電動機に衝撃が生ずるから、内部電気角が変化するまで待つ必要がある。これを防ぐためには、図2のように低圧巻線の u_1 端子には w_2 端子を、 v_1 端子には u_2 端子を、 w_1 端子には v_2 端子をそれぞれ対向させて内部電気角の差を 60 度にして切換を容易にするが、このようにすると発電電動機に接続する途中で相順の入変えを必要とするので、各相の密閉母線箱を入替えねばならず、このために大きなスペースを要しきわめて不便かつ不経済であるばかりか電氣的にも望ましいことではなかった。

この発明では、全電圧および半電圧端子間の内部電気角の差を、 60 度として発電電動機の運転を円滑にしても、上述したような問題を解決したものであり、図1に示すように Δ 結線された低圧巻線より全電圧端子 u_1, v_1, w_1 と内接 Δ 結線の半電圧端子 u_2, v_2, w_2 とを変圧器油槽 10 外に引出し、これら両端子側をそれぞれ切換器 SW_1, SW_2 を介して発電電動機 GM へ接続する際、油槽 10 外に引き出す低圧巻線の各全電圧および半電圧端子の配列を、全電圧端子 u_1 には半電圧端子 w_2 を、全電圧端子 v_1 には半電圧端子 u_2 を、全電圧端子 w_1 には半電圧端子 v_2 をそれぞれ対向させて配置することによって揚水発電所用変圧器を構成するものである。

このようにすることにより、全電圧および半電圧端子の引出線を切換器 SW_1, SW_2 に接続する途中で、引出線を収納した各密閉母線箱を交さすることなく容易に行なえる。(白土)

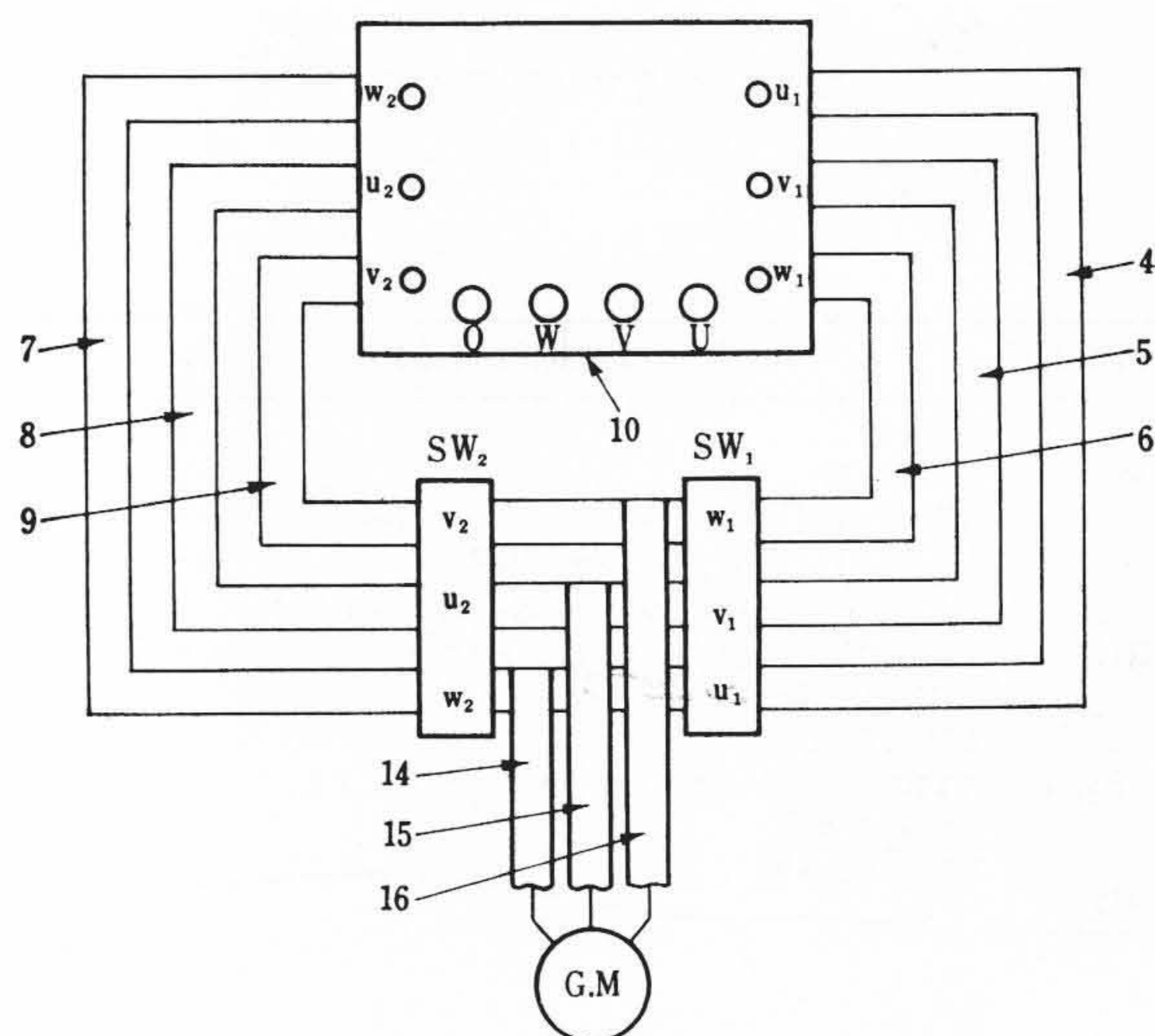


図 1

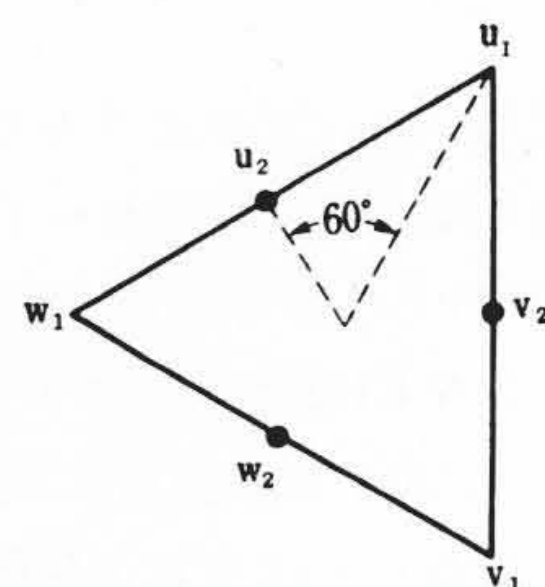


図 2