

# ネットモデルによる 離散系制御設計支援ツールの開発

北九州職業能力開発短期大学校 西見安則・木下七生

A Development of CASE Tool for Design on Discrete Event Systems Control by Net Model  
Yasunori NISHIMI, Nanao KINOSITA

**要約** パーソナルコンピュータを用いた離散事象システムの制御設計支援ツールを開発した。ペトリネットのサブクラスモデルを用いた設計支援ツールである本ツールを用いれば、制御対象の要求動作表現から連続してネットモデルによる制御内容をコンピュータ上に作成できる。作成された制御内容は、直ちにシミュレーションで試験され、結果が良ければプログラムをコントローラに転送して実試験できる。また、実試験の状況は、モニタによりリアルタイムに監視できるので、問題分析を容易にしている。また、従来のネットモデルでは、システムの動作を緊急に停止するための一定の方法が示されておらず、解決は個々のユーザの努力に任されていた。ここに提案する方法を用いれば、明確な図的表現で緊急停止を容易に記述できる。

## 1 はじめに

離散系の制御対象は、連続系（フィードバック系）の制御対象とは区別して離散事象システムとして捉えられるのが一般的である<sup>(1)</sup>。離散事象システムは、生産工場の自動化（FA）、自動機械の制御盤、計算機、通信ネットワーク等のシステムに見られるが、近年の自動化の動向の中で注目を集めている制御対象である。このシステムは、離散的な事象と離散的な状況とで構成され、事象は不規則な時間間隔で生起する特徴を持つ。その際システムは、複数の状況が並行的に保持され、それぞれがランダムに別々の事象を生起して次の状況に遷移する並行性と共に単一流れの中でも複数の事象が同時に生起して複数の状況の成立と状況遷移を引き起こす同時進行性の特性を持つ。したがって、コンピュータによる制御では、離散系も連続系と同様並列処理及びリアルタイム性が要求される。

従来この離散系の制御は、シーケンス制御で実現されることが多く、制御内容の表現形式はリレー回路などの電気回路が採用されてきた。しかし、回路表現ではシステムの動的状況が明示されないため、ダイナミック動特性を追跡・分析するには不具合があった。

更に近年のように大規模化・複雑化したシステムの制御を見通しよく設計するには、制御内容を構造化・階層化して表現することが要請されている<sup>(2)</sup>。しかしながら、回路表現でこの要請に応えるには困難があった。また、回路設計方法論は必ずしも確立しておらず、特に当該の非同期式順序論理回路の設計方法論の領域では、順序論理回路論や論理代数論を基礎とした手法が一部で用いられているに過ぎず、ほとんどの場合基本回路要素の組合せ手法や過去から伝承されてきた経験的手法に頼っているのが実状である<sup>(3)</sup>。

このような背景から、離散事象システムの制御を体系的に取り扱う要請が高まり、これに応える新しい手法としてペトリネットモデルが注目され、ペトリネットのサブクラスモデルを応用する試みが各方面で検討されている<sup>(4)</sup>。筆者の最終目標も、ネットモデルを用いた制御設計表現法の確立に置いているが、今回この方法論確立に向けての一過程として、パーソナルコンピュータによる設計表現支援ツールを開発したので報告する。

## II ペトリネットと制御設計表現モデル

### 1 モデリング

離散事象システムの中でも特にシーケンス制御システムの制御内容の設計表現においては、具体的詳細レベルではコントローラの影響を受ける。従来、シーケンス制御システムの表現において用いられてきた電磁リレー回路や論理回路等の回路表現では、コントローラである電磁リレーの電気回路接続やデジタルICの2値論理回路の具体的詳細レベルを表現している。また、コントローラにコンピュータを用いた場合の表現では、一般にフローチャートが用いられるが、この表現はコンピュータプログラムの処理手続きを示しているのみである。即ち、これら従来の表現では、How-to-do内容の表現が主体であって、制御システムの表現をコントローラ依存形式から独立させ、制御対象及びコントローラを含めた制御システム全体と制御対象に要求される動作(What-to-do)まで表現できてはいない。システムが全体として備えなければならない制御内容の構造やダイナミックな動作状況までも表現できるようにするには新しい表現法が必要であった。

シーケンス制御システムは、離散事象システムとしてのいくつかの固有な特性を持つが、その中でも特に重要であるのは、上述した、制御内容の構造やダイナミックな動的特性をいかに表現できるかである。ペトリネット<sup>(5)</sup>は、この離散事象システムの特性を数学的な表現でモデル化するモデリングツールのひとつである。本モデルを用いれば、制御内容の構造と共にシステムの持つダイナミズムまでの表現が可能である。

ネットモデルを用いてシステムを表現する際の基本的考え方は、システムが互いに関連し合う個々の要素で構成されていると捉えていることである。個々の要素は、その要素が実行中であるという固有の状況<sup>(注1)</sup>を表すことになるが、状況という概念は、システムの現在および将来の動作を記述する上で欠くことのできない情報を抽象的に表現するものである。また状況は、その過去の履歴に依存して、空間時間経過と共に変化していく。関連し合う要素を状況で捉えることは、システムをモデル化する際に重要である。

### 2 ネットモデル

ペトリネットをシーケンス制御システムの表現と制御に応用するため、ペトリネットモデルに制限と拡張を施したサブクラスモデルがいくつか提案されている。ここでも、基本的にはその中の一つマーク流れ線

図を援用するが、シーケンス制御システムにおける実用上の要求から筆者は、このモデルを拡張してシステム緊急停止機能を付加した。本稿の第三章で述べる制御システムの設計支援ツール開発仕様は、この拡張モデルに基づいて作られたものである。以下準備として、使用モデルの要点を述べる。

#### 2.1 マーク流れ線図

ペトリネットの容量を1に制限し、ネット外からの2値信号受け入れゲートを付加したモデルであるマーク流れ線図(Mark Flow Graph 以下MFG)は、図1に示す要素で構成される<sup>(6)</sup>。ボックスとマークによってシステムの各部の動作状況を、トランジションによって動作状況の変わり目を、アークによってボックスとトランジションとの接続をそれぞれ表している。また、トランジションに対応する事象の発生を許可あるいは抑止するものとして、ゲート枝があり、許可ゲート枝と抑止ゲート枝とがある。ペトリネットモデルに対してゲート枝の中でネット外部からの信号枝がMFGでの新たな追加である。

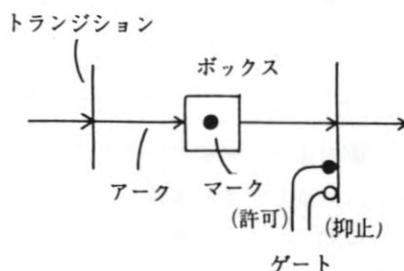


図1 MFGの基本要素

MFGは、マークの遷移によってシステムの動作状況を表現するが、マークの遷移は次の規則にしたがって行われる。

- ①ひとつのトランジションにおいて、入力側のすべてのボックスにマークが存在し、出力側にマークの存在するボックスがひとつもなく、かつ、許可ゲートが真(1)、抑止ゲートが偽(0)であるときトランジションは点弧する。
- ②点弧すると、トランジションの入力側のすべてのボックスのマークが消滅し、かつ、出力側ボックスのすべてにマークをひとつ生起する。

上記規則は、論理式で記述できる。図2に示すように任意のトランジション $t_k$ の入力側にU個のボックス、出力側にV個のボックス、M個の許可ゲート、N個の抑止ゲートが接続されているとする。入力側のボックス変数を $b_{ku}^i$  ( $u = 1 \sim U$ )、出力側のボックス変

数を  $b_{kv}^0$  ( $v = 1 \sim V$ )、許可ゲートを  $g_{Em}$  ( $m = 1 \sim M$ )、抑止ゲートを  $g_{In}$  ( $n = 1 \sim N$ ) とすると点弧条件式は次式となる。

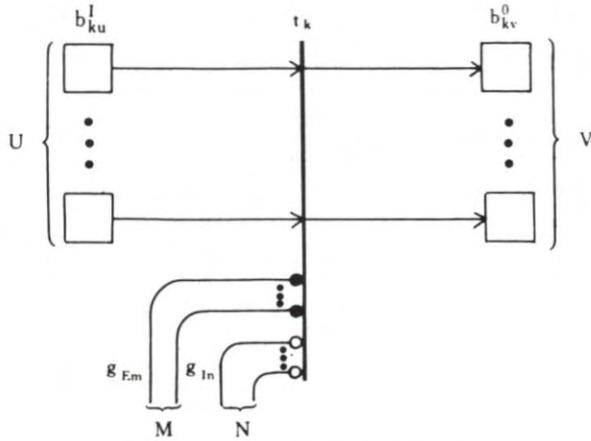


図2 トランジション  $t_k$  の点弧規則

$$t_k(s) = \prod_{u=1}^U b_{ku}^1(s) \cdot \prod_{v=1}^V \overline{b_{kv}^0}(s) \cdot \prod_{m=1}^M g_{Em}(s) \cdot \prod_{n=1}^N \overline{g_{In}}(s) \quad \dots(1)$$

ただし、 $s$  はタイムシーケンス番号 ( $s = 1, 2, 3 \dots$ )。

また、トランジション  $t_k$  の点弧によるボックスのマーク消滅および生起は、次式で表される

$$b_{ku}(s+1) = b_{ku}(s) \cdot \overline{t_k}(u = 1 \sim U) \quad \dots(2)$$

$$b_{kv}(s+1) = b_{kv}(s) \cdot t_k(v = 1 \sim V) \quad \dots(3)$$

## 2.2 シーケンスネット

シーケンスネット (Sequential net 以下 S-net) は、システム動作の安全性の確保の必要から、MFGで表現された工程などの作業実行状況に対してこれを緊急に停止する表現方法を提案している<sup>(7)</sup>。

MFGでは緊急停止表現に問題があった。問題点は次のとおりである。一連の動作からなる作業は、図3で示すモデル構造となる。各動作は各ボックスで表され、ボックス内にマークが入ることにより起動され、マークが出て行くことにより停止する。工程が正常に進行し一連の動作を停止させる必要がない場合はこれでよいが、ある動作を実行中に緊急停止を行う場合に問題が生ずる。すなわち、実行中の動作を停止させるためには、そのボックスからマークを消滅させる必要があるが、マークの行き先は次工程しかないため、実行中の動作は停止するものの次の動作が新たに起動されてしまう。図3を例にとると、トランジション  $t_2$  に付くゲート  $g_2$  から  $b_2$  のマークを消滅できるが、マーク

は  $b_3$  に遷移し、これが動作することになり全体として動作停止に至らない。

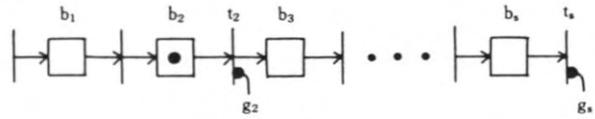
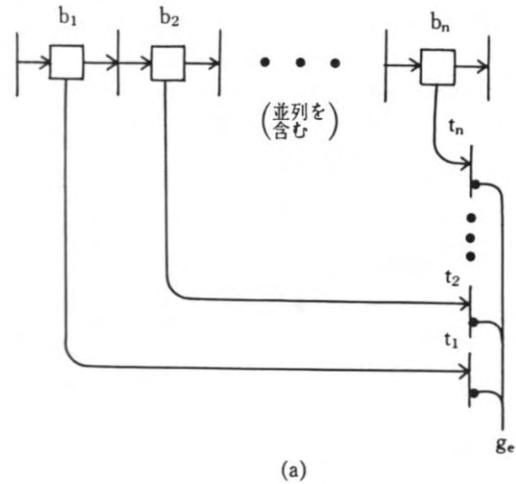


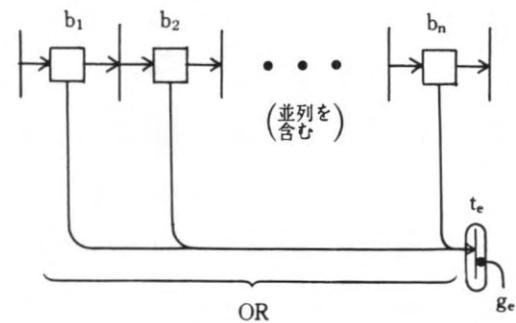
図3 緊急停止問題

### (1) 緊急停止トランジション

一般に、図4(a)に示すような任意のモデルで、停止用の押しボタンスイッチが接続されているゲート枝  $g_e$  により、動作を緊急停止(いずれかのボックスにあるマークを消滅)させるには、マークを消滅させるべき各ボックスから、 $g_e$  が付くそれぞれのトランジション ( $t_1 \sim t_n$ ) へアークを出す構造にすれば可能である。これと等価なものとして(b)のように、ひとつのトランジション  $t_e$  に、各ボックスが論理OR条件でアークを出すようにすると、トランジションが1個になり、(a)に比べて表現が簡単かつ明瞭になる。このとき、 $t_e$  の点弧



(a)



(b)

図4 緊急停止表現

条件式は、

$$t_e = (b_1 + b_2 + b_3 + \dots + b_n) \cdot g_e \dots (4)$$

で与えられるので、 $t_e$ の点弧でいずれかのボックスにあるマークを消滅させることができる。ボックスのマークの消滅は、次式で表される。

$$b_n(s+1) = b_n(s) \cdot \bar{t}_e \quad (n = 1, 2, 3 \dots)$$

このトランジション  $t_e$  を緊急停止トランジションとする。緊急停止トランジションの点弧でボックスのマークを消滅させ、システムの緊急停止が実現できる。

上記の一般的な状況表現することの他に実際のシーケンス制御では、タイマ機能およびカウンタ機能の表現が必要である。MFGでは、状況を表すボックス要素の拡張として時間および容量を表現できる要素を導入している<sup>(8)</sup>。しかし、これら時間および容量要素は、システムが緊急に停止したことに対応して、それぞれの設定値到達前に経過した時間および変化した容量をリセットすることはできなかった。そこで、時間および容量要素に対しても緊急停止に対応できるよう要素機能を拡張した。

まず、時間要素を拡張して次のとおりタイマモジュールとして定義する。

(2)タイマモジュール (図5参照)

経過すべき時間をボックス (b部) の出力側寄りのP部にタイマ機能として割り当てる。ボックス内にマークが入ると、設定された時間が経過してはじめて、出力側に接続されているトランジションの点弧条件に寄与する。もし、ボックスに入ったマークが、設定時間を経過しない間に消滅したら、それまでの経過時間はリセットされるものとする。

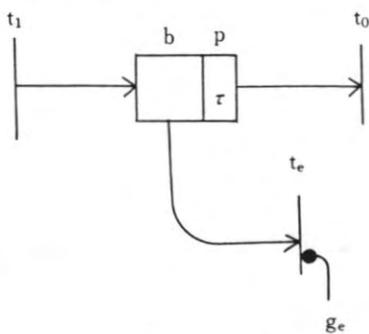


図5 タイマ機能モジュール

つぎに、容量要素を拡張して次のとおりカウンタモジュールとして定義する。

(3)カウンタモジュール (図6参照)

白丸 (○、c部) 内のカウント現在値が設定値 (N) 未満のときのみ、入力側トランジションの点弧を許し、点弧すると○内のカウント値が1つ増える。カウント

値が設定値に達すると、ボックス (□、bc部) 内にマークを1つ生起する。□内にマークがあれば、出力側トランジションの点弧を許し、点弧すると□内のマークは消滅する。カウント現在値のリセットは、○の出力側トランジション  $t_e$  の点弧によって行う。

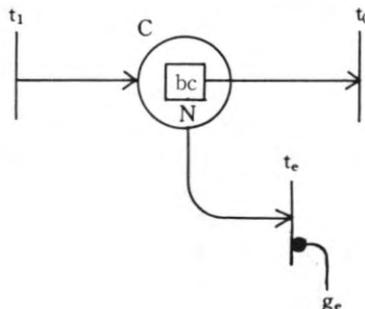


図6 カウンタ機能モジュール

### III 設計支援ツール開発仕様

制御設計支援ツールのCAD部およびシミュレータ部は、既にS-netとして開発済みである。今回の開発は、S-net制御ルールに基づきCAD上に作成された制御内容を実行コントローラである汎用プログラマブルコントローラ (以下PC) の命令コードに変換するトランスレータおよび制御実行状況をコンピュータ側から監視するモニタの開発である。開発作業は、開発計画側が開発仕様を作成し、この仕様を委託側に提示して、外部委託によりソフトウェア作成を行う<sup>(注2)</sup>方法を採用した。以下に開発仕様の要点を示す。

#### 1 トランスレータ

入力したCADデータからPC命令コードを生成する。変換ターゲットPCは、今回は一つの型式の汎用機<sup>(注3)</sup>とする。

##### 1.1 変換要素

変換するS-net要素は次のとおりである。

[トランジション (|)、ボックス (□)、アーク (→)、時間 (□ タイマ)、容量 (⊕ カウンタ)、ゲート (許可/抑止、内部/外部、緊急停止)、緊急停止トランジション (| T00)、緊急停止トランジションへのアーク (\*)]

ただし、次の要素は変換しない。

[CAD図に初期値として含まれるマーク、アービタ]

##### 1.2変換方法

各要素のPC命令コードへの変換は次の方法によって行う。

(1)緊急停止トランジション：II2.2式(4)による。

(2) トランジション：II2.1 トランジション点弧条件式(1)による。

(3) ボックスのマーク消滅および生起：II2.1式(2)および式(3)による。

なお、上(2) トランジションおよび(3) ボックスのPC命令コード変換例を参考として付録に示す。

(4) 時間 (タイマ)：II2.2 時間要素定義による。同様に変換例を付録に示す。

(5) 容量 (カウンタ)：II2.2 容量要素定義による。同様に変換例を付録に示す。

なお、PC命令コードへの変換に際しては上要素を、この順序でそれぞれ1群のPC命令コードに展開する。これ以外の要素(アーク、ゲート、緊急停止トランジションへのアーク)については、上要素の展開の際に第II節の要素定義に従い単独のPC命令またはアドレスとして組み込む。

### 1.3 アドレス割付

トランスレータ実行に際しては、エディタを準備しユーザが次のアドレス割付を行う。また、エディタは不正な割付を排除するチェック機能を持つ。

(1) ボックス、トランジション、タイマ、カウンタ、緊急停止トランジションそれぞれの変換で生成されるOUT命令の出力アドレス。

(2) 緊急停止ゲートを含む外部ゲートのアドレス。

(3) タイマ機能およびカウンタ機能のアドレス。

### 1.4 変換結果出力

次の情報を、フロッピーディスクに出力する。また、要求すれば、同一内容をプリンタへも出力する。

(1) 変換後のPC命令コード

(2) アドレス割付結果

## 2 モニタ

PCでのS-net制御実行状況をコンピュータディスプレイで監視するモニタである。

### 2.1 PC命令転送

変換後のPC命令コード(アセンブラ形式)をPC機械語コードに変換し、通信回線(RS-232C)を通してPCに転送する。また、PCから機械語コードを読み出してコードが正しく転送されたことを確認するペリファイ機能を持つ。

### 2.2 CAD図データ・変換結果の読み込み

PCでの実行状況をモニタする準備として、次のデータをフロッピーディスクから読み込む。制御実行状況表示画面は、S-netのシミュレーション画面と同一とする。

(1) 変換前のCAD図データ

(2) 変換後のPC命令コード

(3) アドレス割付結果

### 2.3 制御実行状況監視モニタ

通信回線(RS-232C)を通して、PCでのS-net制御実行状況を監視してコンピュータディスプレイ上に表示する。動作を監視するのは次の要素である。

(1) ボックスのマークの有無

(2) タイマ機能モジュールの動作状況、設定時間、経過時間

(3) カウンタ機能モジュールの動作状況、設定カウント値、カウント現在値

(4) 外部ゲートのON/OFF

### 2.4 通信環境設定

次の情報に対する通信環境を設定できるようにする。

(1) PC号機番号

(2) 伝送速度(bps)

(3) パリティ(偶/奇)

(4) コード(ASCII7ビット/JIS8ビット)

## IV 支援ツールの特徴

(1) ネットモデルは、並行的な状況がランダムな事象の生起で進行するシステムの動作を表現する図的記述言語であるから、便利なグラフィックエディタを準備した。CADモードにおいて、アイコンに示してあるボックス、トランジション等の要素をマウスで選択して目標の制御内容を作成することができる。変更・修正は作図と同様にマウスで容易に行える。

(2) CADによる制御内容の作成が完了したら、制御内容が設計仕様に適合しているか否かを検証するために、コントローラで実行する前にオフラインでのシミュレーションが可能になっている。初期マークの配置から点弧可能なトランジションを逐次点弧(トランジションを示す棒|が発色)させていきながら、システムの動作をマークの移動によってシミュレートできるようになっている。

(3) 設計した制御内容は、その構造如何によってはシステムの動作が先にも後にも進めない続行不能のデッドロックに陥ることがあるが、離散事象システムの解析ではデッドロックの検出が重要である。シミュレーションには、デッドロックの検出機能を備えていて、デッド時点で画面と音声による表示を行い、システムの解析に貢献できる。

(4) CADで作成した制御内容の図的表現は、コント

ローラの物理 I/O アドレスの割付をした後、トランスレータによってプログラム命令コードに自動変換され、コントローラに転送できるので、労力を伴うプログラム作成作業を効率化できる。

- (5) 従来のネットモデルでは、システムの動作を緊急に停止するための一定の方法が示されておらず、解決は個々のユーザの努力に任されていた。ここに提案している緊急停止トランジションを用いれば、統一した明確な図的表現で緊急停止を容易に記述できる。
- (6) また、緊急停止表現は、タイマ機能およびカウンタ機能に対しても適用できるようにしたので、タイマ、カウンタを多用する実用システムの表現に有効である。
- (7) コントローラが制御を実行している間のシステムの動作状況は、通信回線を通してコンピュータのディスプレイ上にマークの動きとしてモニタされる。したがって、制御対象を接続した実運転状況も、コントローラから送られて来る情報を基にして、CAD で作成した図の上のマークの移動として、リアルタイムに表示できる。これは、システム動作の全体の把握や故障診断に有効である。
- (8) システムの動作上で発生する競合などの制御問題は、システムの動作時間のためにシミュレーションだけでは発見できない場合がある。このときは、コントローラを運転してオンラインで動作状況をモニタしながら解決しなければならない。モニタ機能は、この問題分析作業を効果的に支援できる。

## V むすび

パーソナルコンピュータによる離散事象システムの制御設計支援ツールを開発した。ペトリネットモデルを応用した設計表現法検討のための支援ツールである。ツールの CAD 部及びシミュレータ部は既に開発済みで、今回は汎用プログラマブルコントローラへのトランスレータとコントローラでの制御実行状況のモニタである。

ネットモデルを汎用コントローラで実現した例はこれまでにないので、本ツールを用いれば、現有のコントローラでネットモデルを実行でき設計表現法の検討ができる。

また、従来のネットモデルでは、システム動作の緊急停止表現に難点があったが、提案した方法によれば、明確な図的表現で緊急停止を容易に記述できるようになった。

開発した設計支援ツールは、数回の保守の結果、問題点をかなり改善できた。今後の課題としては、本ツールを活用した制御設計方法論の検討がある。

付記：本研究は雇用促進事業団の指定研究費による。

### (注)

- (注1) ここでは、状態を2値の「1」または「0」を表すことにしており、状況は2値「1」・「0」の双方を表現した要素として状態と区別をしている。
- (注2) 株式会社 S C C 東京都中野区中野5-62-1
- (注3) OMRON SYSMAC C200H

### (参考文献)

- (1) W. Reisig/長谷川健介・高橋宏治訳：ペトリネット理論入門、シュプリンガーフェアラーク東京、1987
- (2) Yourdon・Constantin/原田実・久保未沙訳：ソフトウェア構造化設計法、日本コンピュータ協会、1986
- (3) シーケンス制御工学体系化委員会：シーケンス制御の工学体系化に関する調査研究、電気学会技術報告(II部)第138号、1982
- (4) 市川惇信：ペトリネット研究の動向と展望、計測と制御、Vol. 28, No. 9, 1989
- (5) J. L. Peterson/市川惇信・小林重信訳：ペトリネット入門、共立出版、1992
- (6) 長谷川・高橋・増田・大野：非連続システム制御のためのマーク流れ線図の提案、計測自動制御学会論文集、Vol. 20, No. 2, 1984
- (7) 西見安則：ネットモデルによって表現されたシーケンス制御システムにおける緊急停止法、計測自動制御学会第6回離散事象システム研究会講演論文集、1990
- (8) 高橋宏治：マーク流れ線図による非連続生産システムの制御とシミュレーションに関する研究、東京工業大学博士論文、1984

〈付録1 S-net各要素のPC命令コード変換例〉

(1) トランジション

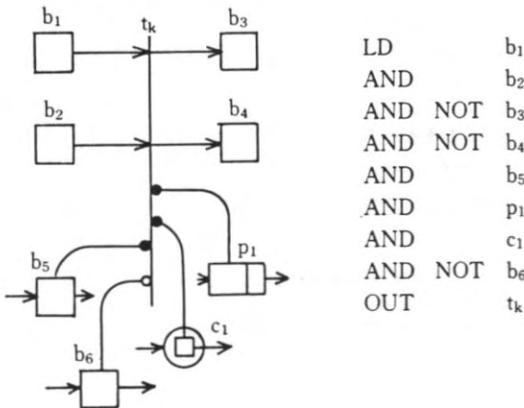
(a)入出力側に対する要件

- 入力側アーク：0以上、複数存在可能
- 出力側アーク：0以上、複数存在可能
- 入力側アークまたは出力側アークが少なくとも一つ接続されていること。
- 許可ゲート：0以上、複数存在可能
- 抑止ゲート：0以上、複数存在可能
- 抑止ゲート：0以上、複数存在可能
- 状態：点弧「1」、非点弧「0」

(b)点弧の決定要件

- (ア)入力側アークに接続されているボックス、タイマ、カウンタが全てON「1」でありかつ出力側アークにつながっているボックス、タイマ、カウンタが全てOFF「0」であって、かつ、
- (イ)許可ゲートが全てon「1」でありかつ抑止ゲートが全てoff「0」であること。

(c)変換例 (付図1参照)



付図1 トランジション変換例

(2) ボックス

(a)入出力側に対する要件

- 入力側アーク:0以上、複数存在可能
- 出力側アーク(緊急停止アークを含む):0以上、複数存在可能
- 入力側アークまたは出力側アークが少なくとも一つ接続されていること。
- 状態:マークの有「1」、無「0」

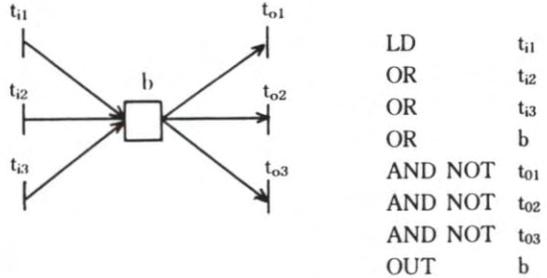
(b)状態の決定要件

- (ア)マークがありかつ出力側アークに接続されているトランジションが全て非点弧「0」であるときマ

ク有り「1」となる。

- (イ)マークがあるかあるいは入力側アークに接続されているトランジションの内少なくとも一つが点弧「1」しているときマーク有り「1」となる。
- (ウ)上(ア)、(イ)以外、マーク無し「0」となる。

(c)変換例 (付図2参照)



付図2 ボックス変換例

(3) 時間 (タイマ)

(a)入出力側に対する要件

- 入力側アーク数：0以上、複数存在可能
- 出力側アーク数 (トランジションへ)：1
- (緊急停止トランジションへ)：0または1
- タイマボックス(b)からタイマ機能 (P) への出力有り「1」：タイマ起動
- 出力無し「0」：タイマリセット
- タイマボックス(b)へ入力不可：マーク有り「1」
- 入力可：マーク無し「0」
- タイマ機能(p)がタイムアップしている
- 出力する「1」
- タイムアップしていない
- 出力しない「0」

(b)状態の決定要件

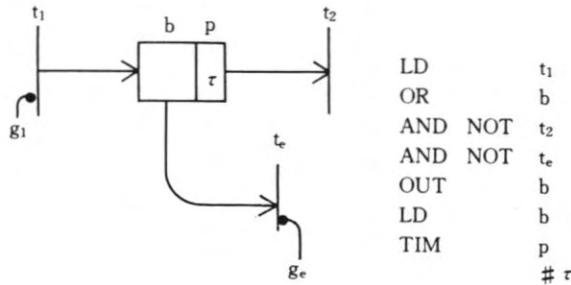
- タイマボックス (b)
- (ア)bにマークがあるかあるいは入力側アークに接続されているトランジションが点弧しており、かつ出力側アークに接続されているトランジションがいずれも非点弧のとき、「1」(マーク有り)となる。
- (イ)上(ア)以外、「0」(マーク無し)となる。
- タイマ機能 (P)
- (ア)タイムアップ：「1」
- (イ)タイムアップでない：「0」

(c)変換例 (付図3参照)

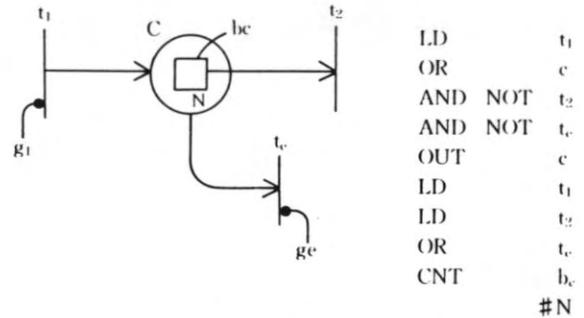
(4) 容量 (カウンタ)

(a)入出力側に対する要件

- 入力側アーク数：1



付図3 タイマ変換例



付図4 カウンタ変換例

- 出力側アーク数 (トランジションへ) : 1  
(緊急停止トランジションへ) : 0 または 1
- トランジションからカウンタ (c) への  
出力有り : カウント  
出力無し : カウントしない
- カウンタ (c) からカウンタ機能 (bc) への  
出力有り : セット bc  
無し : リセット bc
- カウンタ機能 (bc) からの入力 (トランジションへ)  
有り : カウントアップ  
無し : カウントアップでない
- カウンタ (c) の出力が「1」 : カウント中  
「0」 : カウント中でない
- カウンタ (bc) の出力が「1」 : カウントアップ  
「0」 : カウントアップでない

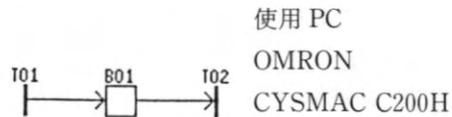
(b)状態の決定要件

- カウンタ (c)  
c 内にマーク投入中であるか、または入力側アークに接続されているトランジションが点弧しており、かつ出力アークに接続されているトランジションが全て非点弧であれば、カウント中とする。
- カウンタ機能 (bc)  
カウンタ (c) からの入力があればカウンタ機能 (bc) をカウントアップ状態「1」とする。
- トランジションからカウンタ (c) への状態  
(ア)入力側アークに接続されているトランジションが点弧しているとき、「1」(カウントする)とする。  
(イ)上(ア)以外、「0」(カウントしない)とする。
- カウンタ機能 (bc) からトランジションへの状態およびカウンタ (c) から緊急停止トランジションへの状態  
(ア)トランジションかまたは緊急停止トランジションのいずれかが点弧したとき、bc および c を「1」(リセットする)とする。  
(イ)上(ア)以外、「0」(リセットしない)とする。

(c)変換例 (付図4 参照)

《付録2 支援ツールによって実試験されたシステムのプログラム例》

[リセット優先自己保持]



CAD 画面コピー

```

***** ゲート状況リスト
*****  MAIN *****
TRANSITION NO : BOX NO . TIMER NO . COUNTER NO
T01           E01-E02
T02           E02

***** アドレス割付け表
トランジションの状態          外部ゲートの状態
MAIN      : T01=    00305
MAIN      : T02=    00306
                                E01=    00001
                                E02=    00002

ボックスの状態
MAIN      : B01=    00300

NAME: T-TM(1)                                PAGE 1
-----TRANSITION MAIN -T01
00000 LD NOT      00300 (MAIN :B01)
00001 AND          00001 (E01)
00002 AND NOT     00002 (E02)
00003 OUT         00305 (MAIN :T01)
-----TRANSITION MAIN -T02
00004 LD          00300 (MAIN :B01)
00005 AND         00002 (E02)
00006 OUT         00306 (MAIN :T02)
-----BOX MAIN -B01
00007 LD          00305 (MAIN :T01)
00008 OR          00300 (MAIN :B01)
00009 AND NOT    00306 (MAIN :T02)
00010 OUT        00300 (MAIN :B01)
00011 END
    
```