

半導体システムのカラーペトリネット記述の簡素化

Simplification of Color Petri net modeling for the Semiconductor Manufacturing System

○岩尾浩晃*, 斎藤和之*

○Hiroaki Iwao*, Kazuyuki Saito*

*会津大学大学院

*The University of Aizu

キーワード： 生産システム(Manufacturing system), ペトリネット (Petri net)

連絡先： 〒965-0053 福島県会津若松市一箕町鶴賀会津大学大学院 コンピュータ理工学研究所 コンピュータシステム学専攻 コンピュータ素子学講座

岩尾浩晃, E-mail: m5071211@u-aizu.ac.jp

1. はじめに

消費者の多彩なニーズに対応するためにメーカーでは単一品種生産から多品種生産に生産方式が移行してきている。半導体生産においては、製品ごとの加工工程に大きな違いは少なく、そのため装置は複数製品を処理することが可能である。しかしながら複数製品を処理するためには装置が製品ごと、もしくは同一製品の場合でも処理段階の違いごとによって動作使用の変更が必要な場合があり、動作仕様の変更にとまなう時間が必要となる。そのために汎用的な装置は絶対的な稼働時間を落とすことにつながり、複数製品を扱うことで各製品の生産量、生産速度、各装置の稼働時間などの生産管理が重要になる。このことを改善するために有馬は納期を守りかつ装置稼働時間をあげるというディスパッチ法を考案した¹⁾。大規模な生産システムの解析をシミュレーションをもとに行っていることが多いことから、私達は半導体

生産システムをペトリネット理論を用いてモデル化し既存のディスパッチ法と新しく考案されたディスパッチ法の性能をシミュレーションによって解析することにした。ペトリネットを採用した理由はグラフィカルであり、並行性・非同期性・分散的・並列的・非決定的・確率的な動作を含むシステムを記述するのに優れていることがあげられ、またシステムを動的にシミュレートすることができる。ペトリネットはその記述の自由度の高さから様々なシステムを表現することができる。生産量、納期管理といった直接消費者に影響するようなデータを予測するときには、予測データの信頼度が重要になる。シミュレーションにおいてデータの信頼度をあげるには現実のシステムを出来るだけ忠実に再現しなければならない。しかし、装置ひとつをとっても複数品種を処理し、また同一品種においても処理段階の違いによって動作使用が多く存在するようなシステムをモデル化しようとする

とき、ペトリネットの構成要素が非常に単純であることから、詳細モデル化したネットはグラフィカルであることの利点の視覚的な見易さを減少させ、同一データを保持するオブジェクトを多く持つことになる。その結果、データ量が増大し、複数製品の処理順序の優先順位割り当てが複雑化する。そのため大規模なシステムにペトリネットを適用する場合はシステムに合わせた修正や、制限を加える必要があることが指摘されている²⁾。私の研究はペトリネット理論をベースにしたシミュレーターの設計であり、本論文では大規模な半導体生産システムをペトリネット理論をもちいて再現するときに、ネットモデルのデータ保持の整理、製品処理の優先順位割り当ての容易化を図るため、ペトリネットの構成要素のデータ保持規則、発火規則の制限について検討する。

2. 半導体生産システムとペトリネット

2.1 半導体業界と半導体システム

半導体製品は技術革新の速さからライフサイクルが非常に短い。一般に、新製品のICが開発されてから4年、市場に出荷されてから2年も経過すれば価格は暴落を始めるとされている。そのため、半導体メーカーは他社に先駆けて次世代ICを市場に出荷し、短期間のうちに利益を確保する（いわゆる市場シェアを確保する）ことが最大の課題とされている。半導体生産工程は複雑で多岐にわたる工程から形成され、量産工場立ち上げるには多額の設備投資が必要となる。半導体製品のコストはそのほとんどが設備投資償却費にあたる。また高い歩留まりを維持するには加工装置の性能に因るところが多く人的要因が大きく影響することはあまりないのも特徴としてあげられる。以上のことから市場シェアを拡大し利益をあげるためには、次世代ICの開発と共に、工場の設備運用が課題と

なっている。半導体の製造工程は、大きくウェハ上にICを形成する前工程、完成したチップを切断し、リードフレーム上に装着、パッケージ化して、検査出荷までの後工程に分かれる。半導体製品はウェハの上に化学物質や金属を層状に重ねて出来上がっていく。Fig.1は前工程の基本的工程図である。半導体製品製造工程は300工程前後からなるが、Fig.1の様に同等の工程を繰り返し回っていきながら半導体製品が出来上がっていく。また半

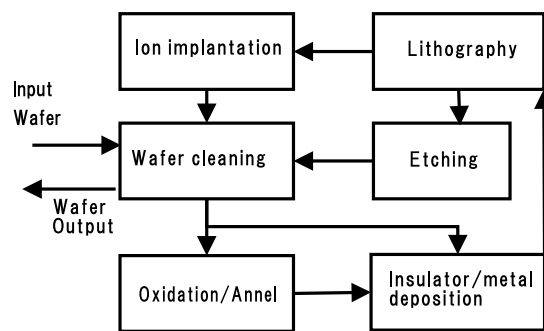


Fig. 1 半導体製造工程（前工程）

半導体生産システムにおいては、製品ごとに固有の生産ラインを持つのではなく、ラインを工程ごとに扱っている。これは加工装置は製品別に専用の装置を使うのではなく、他製品にも使える汎用な装置ということに起因する。各工程には複数の処理装置が存在し、処理されるウェハは製品の種類そして、処理段階によって処理される内容が異なる。Fig.2は前工程内の一工程の構成例である。半

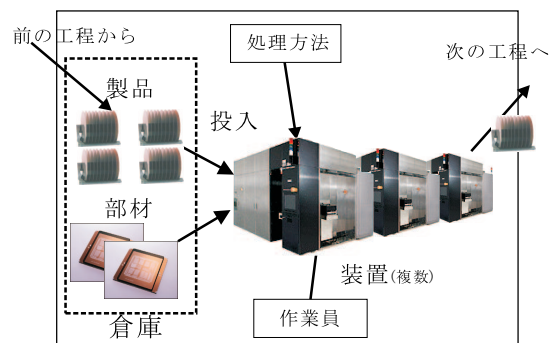


Fig. 2 工程の構成

半導体生産工場での物の流れはウェハによって作ら

れており、製品の生産状況はウェハの動きを監視することで見てくる。先にのべたように半導体製品のコストの多くは設備償却費であることから、装置の稼働時間を多くし、より多くの製品を短期間に作り出していくことが必要になる。装置の稼働時間を低下させる要因に必然的な要因と、偶発的な要因がある。前者は装置の定期的なメンテナンスに必要な時間、加工する品種を変えるときに生じる段取り時間がある。後者には装置の故障、予定外の処理時間の遅れ、人為的な装置の操作ミスなどがあげられる。製品ごとの加工工程に大きな違いは少なく加工装置の流用が効くことから多品種生産で工場を運営するときに各装置を無駄無く稼働させることで、運用の効率を解析するにあたって、シミュレーションからえられるデータを元に実データとの比較によって解析することがおおく、生産システムシミュレータが多くリリースされていることからもうかがえる。

2.2 ペトリネット

ペトリネットはお互いに関連しあう同時進行的な要素からなるシステムをモデリングすることを目的に発展してきた。ペトリネットの要素として、プレース(Place)、トランジション(Transition)、アーク(arc)の静的な3要素とトークン(Token)と呼ばれる動的な要素がありFig.3の様に表示される。ネットの状態を表すのに使用するのがトークンであり、トークンがマークされているプレースによってネットの状態の変化を表す。ネットの動作を司るのがトランジションであり、入力プレースに入力アークが要求するだけのトークンが存在するときトランジションが動作し、この動作を発火(fire)すると言う。トランジションが発火することでネット上のトークンがプレースから取り出されたり、プレースに新たにマーキングされ、あたかもトークンがネットを移動しているように動作する。ペト

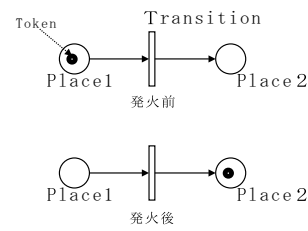


Fig. 3 基本的ペトリネットの要素

リネットの拡張の一部として、カラーペトリネット (Color Petri Net)、時間ペトリネット (Timed Petri Net)、確率ペトリネット (Stochastic Petri Net)が存在する。

カラーペトリネット: トークンを複数色で表し、トークンの色ごとに属性の差別化を図ることができる。このようなトークンをcolored tokenと呼ぶ。

時間ペトリネット: トランジションに遅延時間 $T(T > 0)$ を持たせ、遅延時間を持つトランジションをTimed Transitionとよぶ、発火してから T 時間経過すると出力プレースにトークンを置くことで、時間の概念を持つことができる。

確率ペトリネット: 時間ペトリネットの遅延時間 T を連続な確率分布をもつ確率変数で定義し、時間の揺らぎをもつことができる。

生産システムをペトリネットで記述するために両者の対応関係を考える。多品種生産システムでは2種類以上の製品が存在し、かつ、処理段階の違いで加工処理のされ方が違う。この違いを色つきのトークンで表す。装置をtimed transitionで表すことで各装置は処理時間 t を持つことができる。製品の製造レシピをアークで指示することでプレースからの処理される製品、処理する装置を表せる。この対応関係を用いてFig.2の単一工程をペトリネットでの記述したものをFig.4で表す。トランジション1~3の入力アークは発火組み合わせ $a-b:c$ で表し、入力プレースAのトークン a と入力プレースBのトークン b が数量1で発火し、トークン c を出力するものとしてある。ここでは単純に対応表に

基づいて工程例をペトリネットで表現したが、このままでは半導体生産システムを記述し、シミュレーションを行うことができない。私達は以前に

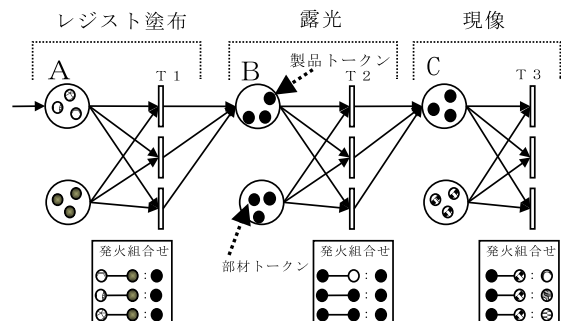


Fig. 4 工程のペトリネット表現

カラーペトリネットシミュレーターを作成した。半導体生産システムを従来のカラーペトリネットの要素のみでは表現できない部分が存在する。まず、半導体製造工程は数百におよぶ工程を必要とするが、Fig.1からも分かるように実際には各工程をループしながら製造されていく。このことは同じ装置郡を使用することでネットのモデルサイズは小さくなる。その一方で製品の属性が製品種別と製品ごとの工程種別が存在し、工程段階が異なる同製品が多数同じ工程に存在する。このことがトランジションがもつ製品、処理段階ごとの処理時間、アークで表されるトークンの発火条件と発火後のトークンの変化のデータを重複して保持することになり、ネット内部の構造を複雑になってくる。またループも単純な環状ではなくウェハ洗浄工程、露光工程の後では分岐をしており、トークンの流れに条件分岐の構造を含むことが一層ネットモデルを複雑にさせる。また製品の流れや、処理条件をアークとトランジションが別々に保持することで製品を変更する度にすべてのアーク、トランジションのデータを変更しなければならない。そのために従来のカラーペトリネットでは半導体生産システムを表現するのではなく、半導体生産システム用にカラーペトリネットを機能拡張、

規則の変更をしてやることでネットモデリングを簡素化を図る。

3. 半導体生産システムをペトリネットで記述する

3.1 製品トークンと装置トランジションの関係

まず工場を流れる半導体製品について考える。トークンの種別を製品を製品トークン、部材を部材トークンと分ける。製品の違いをトークンのカラーの違いで表し、属性の種類分けをする。先にも述べたように同一工程内に複数製品、同製品でも異なる工程段階の処理待ち製品が存在する。このことからカラーの分け方は製品名、工程段階という2段階の色付けが必要になる。また半導体製品は製品ごとにレシピとよばれる製造工程表が存在する。半導体製品製造において、装置の製品処理時間というものは装置固有の処理時間を持つのではなく、同じ工程を通過するときでも製品、製品の処理段階によって異なる。時間ペトリネットはトランジションが発火完了に遅延時間を持つことで時間的な扱いをするのだが、半導体生産においては各製品がレシピから処理時間を装置に渡す形をとる。このために装置トランジションは処理時間を固有に持つのではなくトークンによって処理時間を渡される形をとらなければ装置トランジションがすべての処理時間を持たなければならなくなり、同じ情報を複数のオブジェクトが保持することになる。工場を流れる製品をかえる度に装置の処理時間を再設定しなければならない。複数の同種類の装置が存在するネットにおいてその作業は膨大になってしまう。これを回避するためにも製品トークンが各製品ごとにレシピ（工程表）情報を持ちレシピを参照して処理時間を装置に与えるようにする。これによってデータ保持を簡潔にすることができる。table1は製品トークン

が保持するレシピの属性の種類を表している。レシピは工程番号、処理工程名、処理装置、処理時間、納期、製造数のデータが格納される。カラーという色の種別をこのように表現することで半導体製造工程をペトリネットで表現することが可能になっていく。そのためにネットは可達ネットで書かれる必要がある。このレシピ表をもとにカラー

Table 1 製品トークンが持つレシピの例

製品名	工程番号	処理工程名	処理装置	処理時間	要求部材
製品A	1	ウェハ洗浄	洗浄装置	1	
	2	酸化	酸化拡散装置	3	
	3	レジスト塗布	コーダ	3	レジスト
	4	露光	ステツバ	2	マスク1層目
	⋮	⋮	⋮	⋮	⋮

ペトリネット上での製品トークンの変化はFig.5で表せる。簡単のため製品を1つにし、工程ごとの装置トランジションは同一の装置である例を考える。製品の処理段階は工程の装置トランジション

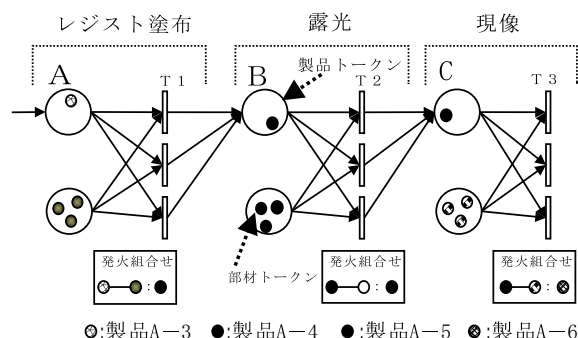


Fig. 5 製品トークンをレシピに対応させる

を通過することで、レシピ上の工程番号が1つ進むことになる。処理を終えた製品は次の工程の属性に変化し、処理されるべき装置、処理時間が変化する。この図では各工程に1層目処理段階の製品しか存在してないが、実際には各工程を一通り回ってきた、2層目以降の製品段階を処理する製品トークンが存在してくる。

3.2 プレースについて（優先度の属性を付加）

製品処理を考えたときに、各工程に貯まっている処理待ちの製品の中から、どの製品から処理するか、どの装置に割り当てるかを決定しなければならない。この時に、まず処理する製品の優先度の高いものをチェックする（製品処理の優先度については後述する）、処理される製品用部材をチェックする、そして装置に割り当てる。ペトリネットで記述するとき製品が格納されるプレースと部材が格納されるプレースをわけることで、チェックするトークンの数を少なくしてやることできる（あらかじめ処理される製品トークンを優先度順に並べておく必要がある）。製品を格納するプレースをメインプレースとし、部材を格納するプレースをサブプレースとする。そして発火する製品トークンの順序を決定するとき、順序の決定の第一要因の製品トークンを保持するメインプレースから発火順序を決め、処理される製品トークンが部材を必要とするときにサブプレースに必要な部材が存在するかをチェックする。いま露光工程の一連の処理工程のFig.6を考える。プレースA、

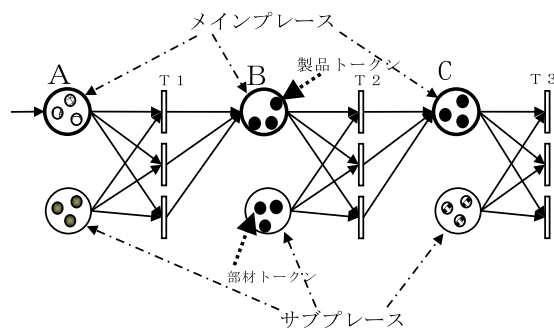


Fig. 6 プレースに優先度を付けたネットモデル

B、Cがメインプレース、それ以外のプレースがサブプレースにあたる。発火可能トークンを探すためにプレース内に格納されているトークンを調べる必要がある。製品トークンが格納されているメインプレースを先に参照し、その次に部材トーク

ンが格納されているプレースBを参照する。トークンの発火可能を調べるときにおいて、プレース参照の優先度をメインプレースをサブプレースより高くすることで検索するプレースの順序を確定的なものにすることができる。またこのときすべてのトランジションの入力アークがメインプレースにつながれている必要がある。また製品トークンが別の工程から入ってくる時、処理を終えた製品トークンが次の工程に出ていくときには必ずメインプレースに格納される必要がある。このことで半導体工場内のウェハの動きはペトリネットシステムでは常にメインプレースを通過していくことになる。

3.3 製品処理について

製造工程において、製品処理をトランジションの発火で表している。トランジションが発火する条件はトランジションに接続しているすべての入力プレースが入力アークが表示するトークンを保持しているとき発火が可能になり、発火する。半導体製品を製造するときに装置が複数種の材料（製品と部材）を要求することがあるが、このとき部材の性質は2種類に分類される。製品の種類ごとに異なり他製品に使用できない部材、製品の種類に関係せず他製品にも使用される部材の2種類である。このことからペトリネットの発火規則を同色発火、無色発火の2つの限定的な発火規則に作り変えることができる。

3.3.1 同色発火

製品の種類によって部材が異なる場合、言い換えると製品との対応を管理すべき部材について考える。製品の種類そして、同じ製品であっても処理段階によってその処理方法は異なり、要求される部材も異なる時、製品（製品の処理段階）に対応した部材のリストが必要になる。ペトリネット

ト上でこのことをモデル化すると発火の可能性を調べるために、対応表を参照し、製品に合う部材をリストから検索し、サブプレースから部材トークンの有無を調べなくてはならない。露光工程を

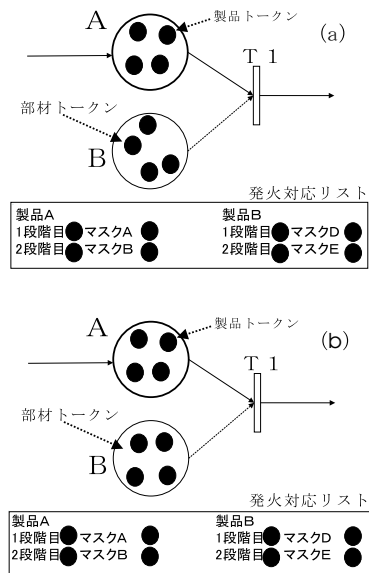


Fig. 7 従来の発火(a)と同色発火モデル(b)

例としてあげる。ここでは簡単のために装置を1台で考えている。今2種類の2製品があり、各製品は2回露光工程を通る。製品の処理段階によって1段階目と2段階目に区別され、1段階目と2段階目は違うマスクを部材として使用する。Fig.7の(a)は従来の発火モデルであり、製品と部材のそれぞれの種類ごとにトークンの色区別している。発火には対応する製品と部材の関係をアークが持つことが必要になる。しかしマスクの様な部材トークンは製品の処理段階ごとに異なるマスクを使用し、他の製品トークンに使われることはない。露光工程において製品トークンと部材トークン（マスク）の関係は1:1であり、製品の対応は唯一に定まる。このことから部材に個々の属性を持たせる必要性が無くなる。また製品は必ずメインプレースに保持することを前節で述べていることから、部材と製品の対応を同色で表し、発火条件を満たすためにはサブプレースに製品トークンと同じ色属性を

もった部材トークンを探せばよくなる。よって対応リストがFig.7の(b)の様に製品と部材とを同一になる。これで実際には対応表を参照せずに、直接部材を格納してあるサブプレースに製品トークンと同じ色を持つトークンを参照することになる。このように製品ごとに特定の部材の必要になる時には同色発火をすることで検索するデータを削減することが出来る。

3.3.2 無色発火

しかし露光工程以外の工程、製品の種類に関係せず他製品にも使用される部材を扱う場合、はどう扱うか？もちろん同色発火では再現できない。いわゆる対応を管理するのではなく、工程に存在する絶対量を管理すべき部材と考えられる。これは部材トークンに色に関する特別な属性を与えることで表現できる。このよう場合部材トークンはすべての製品で扱えればいいので、トークンに無色という属性を与える。この無色属性をどの製品にも対応するという性質のものにする。ただし、このような強力な性質を持ったトークンを扱うためには使用条件を限定的にする必要がある。

- 1) メインプレースに無色トークンを置かない。
- 2) 複数種類のトークンが存在するプレースに無色トークンを置かない

メインプレースはネットのクリティカルパスにあたり、メインプレースに無色の製品トークンを置いてしまうと、解析データが無意味なものになってしまう。無色トークンはすべての製品に対応するので、複数種の部材トークンが存在するプレースに置いてしまうと何の部材トークンとして存在するかが不明になるからである。そのため複数の無色の部材トークンを扱う時には、部材トークンを保持するサブプレースを分けてやることで解決できる。

3.4 分岐を表現する

製品トークンが各装置トランジションを通過して工程が進んでいくが、Fig.1からもわかるようにウェハ洗浄、露光現像工程の後に処理ルートが分岐する。table2は同製品においてのレシピによるルートの違いである。各種製品は処理される段階によって分岐しながらループを回っていき、ルートはこのようにレシピによって書かれている。このレシピでは工程5と工程11の後に分岐をおり、工程5の後ではイオン注入、工程11の後ではエッチングに進んでいる。この分岐をペトリネットで表現するとFig.8の様になる。

Table 2 工程によるルートの違い

製品名	工程番号	処理工程名	処理装置	処理時間
製品A	:	:	:	:
	4	露光	ステツバ	2
	5	現像	デベロッパ	2
	6	イオン注入	ドーピング装置	3
	:	:	:	:
	10	露光	ステツバ	2
	11	現像	デベロッパ	2
	12	エッチング	エッチング装置	2
	:	:	:	:

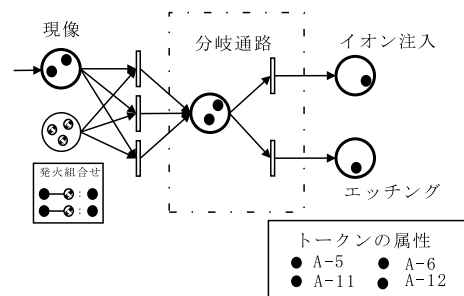


Fig. 8 ペトリネットによる分岐構造

ここまではトランジションは装置としての役割のみを行ってきた。そして各製品トークンはトランジションを通るごとに工程段階を1つ進めて属性の変化をしてきた。しかし、この分岐では製品トークンがトランジションを通るときに工程段階が進むわけではなく、この分岐の構造によって分配されるだけの処理をすることになる。もちろん装置とは異なるので、トランジションが処理時間を

持たなくなる。そのかわり分岐用のトランジションはアウトプットにつながっている工程名のデータを持ちを製品トークンを参照して処理工程名が一致する製品トークンを発火させ、分配する。

3.5 階層化による全体像

半導体システムをペトリネットで表現するときの対応づけを整理するとtable3の様になる。各製品の工程をレシピとして持つことで、ペトリネットのアークの要素が可達なルートを表示することになり、装置トランジションの処理時間をレシピから参照することになる。また、製品トークンは完全に有色で表現され、製品名、工程段階の2段階の色識別を必要とする。部材トークンは工程に依存して有色、無色の場合に分かれることになる。各

Table 3 生産システムとペトリネットの対応づけ

半導体生産システム	ペトリネット
製造レシピ(処理方法)	製品tokenが保持
製品(原材料)	製品token(完全有色)
リソース(部品,材料)	部材token
製品部材の種類	トークンカラー色:(製品名)-(工程段階)
機械装置(処理時間つき)	timed transition 処理時間はレシピを参照
製品置場	main place
部材置場	sub place

工程には必ずメインプレースが存在し、製品トークンはメインプレースから装置トランジションを選択され処理されていく(分岐構造に置いてはプレースが1つしか存在しない)。このことからメインプレースを基準に各工程を階層化することが容易にできる。階層化を施すことで全体像を見ることがたやすくなる。Fig.9はFig.1をメインプレースを基準に階層化を施し表現したたペトリネットモデルである。点線の四角で囲まれているのが階層化された工程を表している。それ以外の部分は分岐構造である。このネットにレシピを参照する製品トークンを流すことで半導体生産システムをシュ

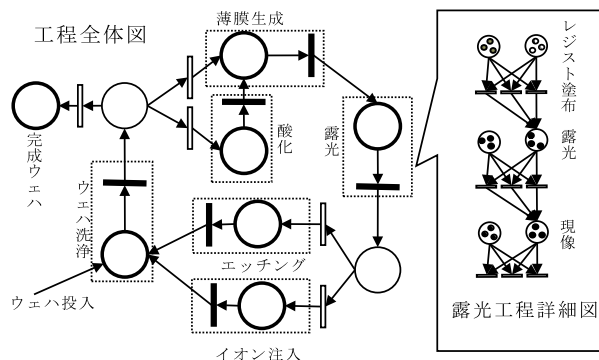


Fig. 9 半導体製造工程のペトリネット表現

ミレートするときのペトリネットが出来上がる。

4. まとめ

本研究は半導体システムのシミュレーターをペトリネットを使用して設計することを目的として、今回は主に半導体生産システムをペトリネットで記述することについて考察した。データ保持の集約とプレースの保持するトークンの分類分け、発火規則の変更、そして製品トークン分配を行う特別なプレースとトランジションからなる分岐を持つことで複雑な半導体生産システムを記述することができる。今後は半導体生産システムシミュレーター設計するにあたり、製品処理順序のソート要因の決定やディスバッチ部分に必要なデータの保持に関する事など考慮すべき点はまだ幾分か残されており、よりよいシミュレーターにするために今後考察していかなければならない。

参考文献

- 1) S.Arima and K.Saito: A Simulation Study on Periodical Priority Dispatching of WIP for Product-mix Fabrication, Proc.IEEE/SEMI Advanced Semicon. Manufact. Conference, **13**, 33/37 (2002)
- 2) 村田 忠夫: ペトリネットの解析と応用, 1/3, 近代科学社(1992)
- 3) 椎塚 久雄: 実例ペトリネット-その基礎からコンピュータツールまで-,112/153, コロナ社(1992)
- 4) Jean-Marie Proth and Xiaolan Xie: Petri Nets A

Tool for Design and Management of Manufacturing Systems, JOHN WILEY & SONS(1996)

- 5) 深海 登世司: 半導体工学-基礎からデバイスまで-, 275/322, 東京電機大学出版局(1987)
- 6) 中小企業金融公庫調査部: 半導体・液晶産業の業界動向と中小企業のビジネスチャンス, 1/12, http://www.jasme.go.jp/jpn/result/c2_0002.pdf (2001)