順序型 複合機械 概念에 관하여

金 玄 在
(全南大 工大 敎授)

■ 차 례 ■

1. 서론
2. 複合機械의 定義
3. 複合機械의 動作方式
4. 應用例
5. 결 론
參考文獻

1 서 론

順序型 複合機械(sequential polymachine)모델은, 미이리型(Mealy machine)이나 무우어型 機械(Moore machine)처럼, 論理回路나 디지탈 시스템의 記述과 解析 및 設計 等을 위해서 쓰일 수 있는 順序回路의 基礎모델이며, 從來의 모델보다는 더 넓은 應用性을 지니고 있다[4].

즉, 이 複合機械 모델은 論理回路 水準의 設計뿐 아니라 시스템 水準의 設計 문제나 言語構造의 解析을 위해서도 쉽게 合理的으로 適用할 수 있는 基礎的 特徵을 지니고 있다고 생각된다.

이 複合機械의 概念은 이미 1977年에 電子工學會誌에 提示된 것이며, 그후에 약간의 理論的인 展開와 應用이 추가되었고, 앞으로도 더 補完할 計劃이지만, 여기서는 그 概念의 大要를 紹介하는데 그치기로 한다.[1)-3)]

2 複合機械의 定義

複合機械의 概念을 불럭線圖(block diagram)로 提示한다면 그림 1과 같다.[4] 그리고 다음과 같이 定義한다.

〈定義 1〉 複合機械(polymachine)는 다음과 같이 8重組(octuple)에 의해서 規定되는 順序機械(sequential machine)이다. 즉,

$M=\langle \Sigma, I, Q, Y, Z, \delta, \eta, \Omega \rangle$

여기서

Σ = 制御入力(control input)記號 $x_1, x_2, \cdots x_k$의 有限集合이며 空集合이 아니다.

I : 데이타入力(data input)記號 $i_1, i_2 \cdots\cdots i_J$의 有限集合이다.

Q = 記憶裝置(memory)의 內部狀態 $q_1, q_2 \cdots q_n$의 有限集合이며 空集合이 아니다.

Y = 內部의 指示記號(indexing symbols) $y_1, y_2, y_3, \cdots\cdots y_p$의 有限集合이며 記憶裝置의 出力$Q$에서 계산되는 값이다.

Z = 出力記號 $z_1, z_2, \cdots\cdots z_m$의 有限集合이

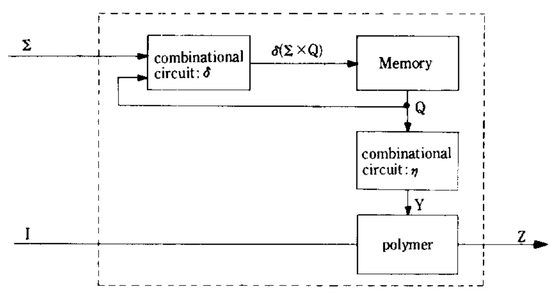


그림 1. 複合機械의 構造

며 空集合이 아니다.

δ =다음狀態函數(next-state function) 이며 이는 다음과 같이 寫像한다.

즉, $\Sigma \times Q \longrightarrow Q$

η : 指示函數(indexing function)이며 다음과 같이 寫像된다.

즉, $Q \longrightarrow Y$

Ω =指示記號 $y_1, y_2, \cdots\cdots y_p$에 依해서 指定되는 出力函數 $w_1, w_2, \cdots w_p$의 有限集合이며 다음과 같이 寫像한다

즉, $I \times Y \longrightarrow Z$

그림 1에서 記憶裝置와 그에 附屬되어있는 組合回路(combinational circuit)는 함께 어울려 하나의 順序回路를 이루므로, 巨視的으로 그것을 하나로 보면, 複合機械를 그림 2처럼 더 간단한 모양으로 表現할 수 있다. 다시 말하면, 複合機械는 巨視的으로 두가지의 모듈(module)에 의해서 구성되는 셈이다. 이때 그 두 모듈, 즉, 두 部分機械를 다음과 같이 定義한다.

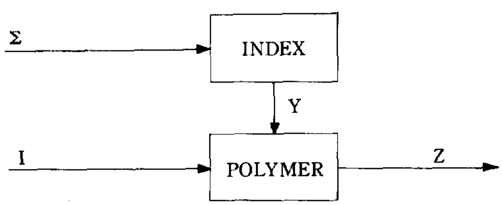


그림 2. 複合機械의 簡略表現

〈定義 2〉 指示器(index)는 制御入力 Σ를 받아들이고 指示記號 Y를 出力으로내는 順序機械이며 複合機械를 構成하는 한 部分機械이다.[7]

〈定義 3〉 複合體(polymer)는 데이타入力 I와 指示記號 Y의 두가지 入力을 그 入力으로 取하고, 複合機械의 外部出力 Z를 내는 部分機械이다.

이상의 定義 2와 3은 複合機械를 制御部(control part)와 被制御部(controlled part)로 小區分하여 부르기 위한 名稱의 定義이다.

그림 2에서 보듯이, 이 複合機械의 두드러진 特徵은 서로 다른 두가지 種類의 入力, 즉 制御入力과 데이타入力을 別途의 입구를 통해서 따로따로 받아들이도록 되어있는데 있다고도 생각된다.

앞의 定義 1에서는 複合體를 $w_1, w_2, \cdots w_p$라는 p個의 函數集合 "Ω"를 實現하는 部分으로 定義했으므로 그 個個의 函數를 實現하는 具體的인 작은 機械들이 필요하게 된다. 다음은 그 작은 機械들을 부르기 위한 名稱의 定義이다.

〈定義 4〉 元素機械(element machines)는 各各의 固有 入出力機能이 $w_1, w_2, \cdots w_P$인 P種의 函數를 個別的으로 實現하는 P個의 小機械들이다.

이 定義 4에 의하여 定義 3으로 주어진 複合體는 다시 다음과 같이 定義된다.

〈定義 5〉 複合體는 各種 元素機械의 有限集合이다.

[3] 複合機械의 動作方式

複合機械의 複合體는 여러個의 元素機械가 모여서 이루어진 것 이라고 定義된 바와 같이, 複合機械는 本質的으로 多機能性을 內包한 機械이며, 또 制御部인 指示器와 被制御部인 複合體로 區分하여 생각할 수 있듯이, 上下 二階의 階層構造(hierarchical structure)를 內藏한 機械이다.

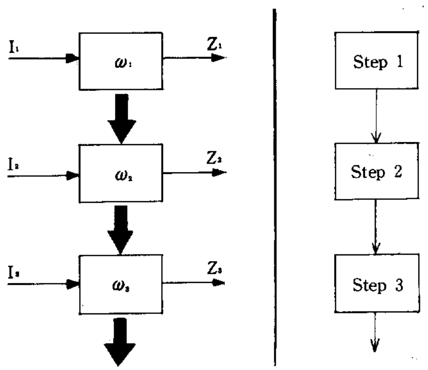
이 複合機械의 時間的인 順序動作(sequential behavior)을 설명한다면 다음과 같다.

1) 指示器의 內部狀態를 q_1에 固定시키고, 그 q_1에 對應하는 指示記號를 y_1, 入出力函數를 w_1, 또 이때 들어오는 데이타入力을 $I_1 \subseteq I$, 出力을 $Z_1 \subseteq Z$라 할때, 이 複合機械는 I_1을 받아서 y_1에 의해 指定된 函數 w_1으로 計算된 出力 Z_1을 내는 機械로 밖에 보이지 않기 때문에, 이 複合機械가 q_1狀態에 固定되어있는 동안은 複合機械의 對外的 機能은 오직 w_1이라는 固有機能밖에 行使하지 못하는 한가지의 元素機械처럼 보인다.

2) 指示器의 內部狀態가 q_2로 바뀌어 q_2에 固定되어있는 동안은 이에 對應하는 指示記號는 y_2에, 內部機能은 y_2로 指定되는 w_2에 固定되므로, 결국 複合機械의 外的機能이 內部의 元素機械機能 w_2로 나타나 보인다.

3) 위의 1), 2)에서처럼, 複合機械의 外部로 表現되어 나타나는 機能은 指示器에 의해서 選擇指定되며, 그 入出力機能이 時間的으로 交代되는 順序動作은 指示器의 順序動作에 따른다.

따라서, 이와 같은 複合機械의 時間的인 順序動作을 圖形으로 설명하자면 그림 3 (a)와 같다. 그림 3 (b)는 複合機械의 順序動作이 計算機의 소프트웨어 (software)와 어떻게 對應하는가를 비교하여 보여주기위한 一種의 흐름圖이다.



(a) 複合機械의 順序動作　(b) 소프트웨어의 구조

그림 3. 복합기계동작과 소프트웨어와의 비교

4 應用例

이번에는 한두가지의 應用例를 통해서 複合機械의 概念을 더 具體的으로 설명해 보기로 한다.

〈例 1〉 다음의 그림 4는 미이리型 機械의 狀態遷移圖 (state transition graph)이다. 이에 관한 順序回路를 複合機械 概念에 立脚하여 設計해보기로 한다.[5]

물론, 미이리型 機械는 그表現이 複合機械表現과 같지않기때문에 우선 그와 같은 미이리型 機械를 複合機械 表現으로 變換하기 위해서는 몇가지 妥協이 이루어져야 한다.

첫째로 그 두가지 機械의 入力을 생각해보면, 미이리型은 한가지 入力이고 複合機械는 두가지

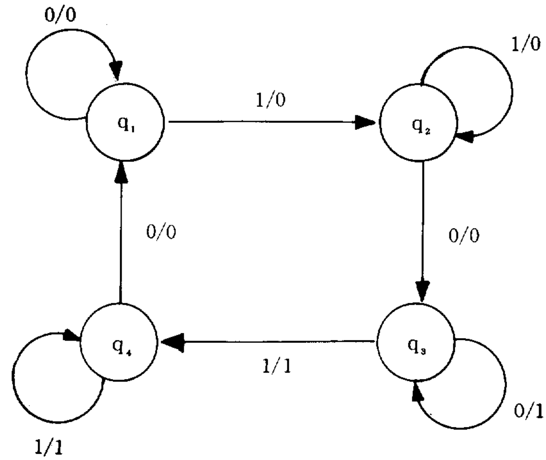


그림 4. 미이리型 機械의 狀態遷移圖

이다. 따라서 이 미이리型 機械의 複合機械 表現을 위해서는 두 入力이 共通인 것으로 생각하기로 한다.

둘째로 미이리型 機械와 複合機械가 크게 다른點은 그 入出力値의 그라프 要素에 대한 配當인데, 미이리型 機械의 경우는 그 入出力雙(I/O-pair)이 그라프의 定何枝(directed branch)에 매겨지지만 複合機械의 경우에는 頂點(vertex)의 狀態量에 歸屬되도록 되어있다.

그러므로, 이상과 같은 相異點의 妥協을 위해서는 다음과 같은 操作이 必要하다.

1) 그라프의 發散核別 分割

그림 4의 미이리型 그라프를 各 狀態點을 中心으로 分割하되, 각각의 狀態가 그 狀態에서

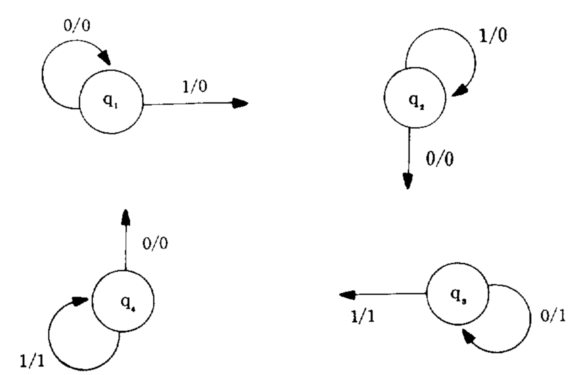


그림 5. 그라프의 發散核別 分割

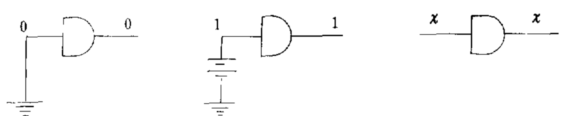
밖으로 향하는 外向枝(outgoing branch)만을 거느리도록 한다면 이는 모두 發散核(divergent kernel)이라고 말할 수 있으며, 그라프를 發散核別로 分割한 셈이다. 즉 그림 5는 그림 4를 發散核別로 分割한 결과이다.

2) 狀態別 元素機械 機能의 決定

그림 5처럼 미이리型機械의 順序圖를 發散核別로 分割했을때는 各 狀態點에 對應하는 元素機械의 機能이 그 外向枝에 매겨진 入出力雙으로 決定된다.

우선 狀態 q_1에 對應하는 元素機械의 函數機能 w_1을 생각해보면, q_1의 두 外向枝에 매겨진 入出力雙이 0/0와 1/0이므로 w_1으로는 零字發生器(0-generator)가 適當하다. 왜냐하면 入力値 如何에 不拘하고 出力이 恒常 零으로 되기 때문이다. 狀態 q_2에 관해서도 마찬가지이다.

그런데 q_3에 대한 w_3로는 壹字發生器(1-generator)를 쓸 수 있으며, q_4에 대한 w_4로는 正의 傳達要素(positive transfer element)가 適當하다. 즉, w_4로는 入力이 그대로 出力側에 傳達되는 것이라야 하기 때문이다. 이 元素機械들을 圖形으로 그리면 그림 6과 같다.



(a) 0－generator (b) 1－generator

(c) positive tranfer element

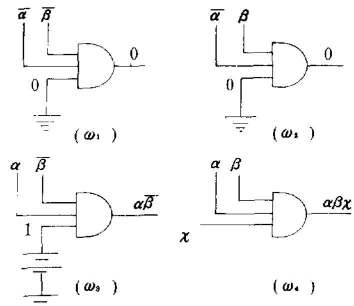
그림 6. 元素 機械

3) 各 元素機械에 대한 命名(tagging)

그림 6에서 보는 바와 같이, 여기서 必要로하는 元素機械들은 모두 다 만들어진 셈인데, 이들을 한데 모아 複合體를 만들려면 먼저 各元素機械에 呼出名(tag-name)을 붙여주어야 한다. 즉 指示記號 y를 割當하는 일이다.

여기서 必要한 네가지 指示記號 즉 呼出名을 y_1, y_2, y_3, y_4라하고, 이 指示記號를 符號化하기 爲해 必要로 하는 指示變數를 α, β라하여, 各指示記號에 指示變數값을 매겨줌과 함께, 呼

Y	Tag variable α	β
y_1	0	0
y_2	0	1
y_3	1	0
y_4	1	1



(a) Tagging list (b) Tagged elements

그림 7. 命名表와 命名된 元素機械

出名이 더 붙은 元素機械들을 그림으로 나타내 보이자면 그림 7과 같다.

4) 複合體(polymer)의 構成

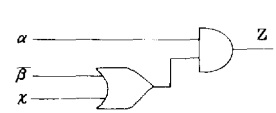
위에서 各 元素機械에 그 選擇條件에 該當하는 呼出名(tag-name)을 붙였으므로, 그들을 한데 모아 合成하면 複合體가 되는데, 그 元素機械들이 모두 論理回路 素子로 되어 있기 때문에 合成回路를 簡單化 시킬 수 있다.

그림 8은 複合體의 入出力特性을 나타내는 眞理値表(truth table)와 그 回路이다.

α	β	χ	Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

χ \ $\alpha\beta$	00	01	11	10
0	0	0	0	1
1	0	0	1	1

$Z=\alpha\,(\bar{\beta}+\chi)$



(a) 眞理値表 (b) 複合體回路

그림 8. 複合體의 眞理値表와 그 回路

5) 指示函數 η을 實現하는 指示函數 回路에 관하여

일반적으로 狀態集合 Q와 指示記號集合 Y가 서로 같지 않을때는 $Q \rightarrow Y$를 計算하는 η回路를 必要로 하게 되는데, Q와 Y를 같게 취하면 指示函數回路 η을 省略할 수 있다.

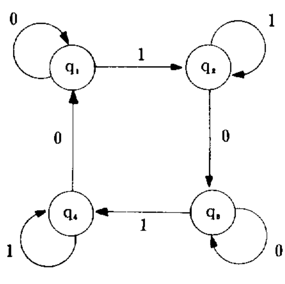
여기서는 $Q = Y$ 되게하여 η回路를 省略하기로 한다. 따라서 $q_1 = y_1$, $q_2 = y_2$ $q_3 = y_3$, $q_4 = y_4$되게 잡는다.

6) 指示器의 狀態遷移圖

이 設計過程에서는 주어진 미이리型 機械의 單一入力을, 그대로 複合機械의 制御入力과 데이타入力에 兩用하기로 했으므로, 그림 4의 定向枝 上에 나타난 出力値 만을 모두 지우면 그림 9와 같은 指示器의 狀態遷移圖가 얻어진다. 이때, 定向枝 위에 남은 入力値는 바로 制御入力値가 되는 셈이다.

7) 플립플롭(flip-flops)의 選定

그림 9의 狀態遷移圖를 갖는 指示器를 作成하기 위해서는 먼저 거기에 사용될 플립플롭을 選定할 必要가 있다.



Q \ x	0	1
q_1	q_1	q_2
q_2	q_3	q_3
q_3	q_3	q_4
q_4	q_1	q_4

(a) 狀態遷移圖 (b) 遷移表

그림 9. 指示器의 狀態遷移圖와 遷移表

여기서는 J-K플립플롭을 쓰기로 한다. 指示器의 狀態數가 넷 이므로 두개의 플립플롭을 써야하고, 두 플립플롭의 이름과 그에 따른 狀態變數를 指示變數와 一致시켜 α, β라 부르기로 한다면, 그림 10(a)와 같은 狀態割當表가 얻어진다. 그림 10(b)는 J-K플립플롭의 勵起表(excitation table)이다.

Q	α	β
q_1	0	0
q_2	0	1
q_3	1	0
q_4	1	1

$q(t)$	$q(t+1)$	J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

d : don't care

(a) 狀態割當 (b) $J-K$플립플롭의 勵起表

그림 10. $J-K$플립플롭의 狀態割當과 勵起表

8) 플립플롭 入力의 計算

두개의 플립플롭 α, β의 네가지 入力인 $J\alpha$, $K\alpha$, J_β, K_β는 外部入力 x와 α, β의 現在出力에서 計算된다.

그런데, 이들 플립플롭의 入力計算을 위해서는 그림 9의 狀態遷移表와 그림10의 狀態割當表 및 $J-K$플립플롭의 勵起表도 必要하므로, 그와 같은 것들을 綜合하여 플립플롭의 入力計算을 爲한 眞理値表를 만들면 그림11과 같다.

present states		input	Next States		Required Excitation			
$\alpha(t)$	$\beta(t)$	x	$\alpha(t+1)$	$\beta(t+1)$	$J\alpha$	$K\alpha$	J_β	K_β
0	0	0	0	0	0	d	0	d
0	0	1	0	1	0	d	1	d
0	1	0	1	0	1	d	d	1
0	1	1	0	1	0	d	d	0
1	0	0	1	0	d	0	0	d
1	0	1	1	1	d	0	1	d
1	1	0	0	0	d	1	d	1
1	1	1	1	1	d	0	d	0

그림11. 플립플롭의 入力計算을 위한 眞理値表

다음으로 그림12는 그림11의 表에서 플립플롭의 入力方程式을 計算해놓은 것이다.

이들 入力方程式은 바로 指示器의 다음狀態函數 δ를 實現하는 組合回路를 나타내는 것이기도 하다.

9) 指示器의 作成과 複合體의 結合

α와 β인 두 플립플롭의 入力回路를 앞의 그림12에 있는 入力方程式 대로 꾸미면 指示器回路가 完成되고, 또 이 指示器를 미리 만들어

x \ αβ	00	01	11	10
0	0	1	d	d
1	0	0	d	d

$J\alpha=\beta\bar{x}$

x \ αβ	00	01	11	10
0	d	d	1	0
1	d	d	0	0

$K\alpha=\beta\bar{x}$

x \ αβ	00	01	11	10
0	0	d	d	0
1	1	d	d	1

$J\beta=x$

x \ αβ	00	01	11	10
0	d	1	1	d
1	d	0	0	d

$K\beta=\bar{x}$

그림 12. $J-K$플립플롭의 人力方程式

놓은 複合體와 結合시킬 때 複合機械의 論理回路가 完成되는 셈이다.

바로 그림13은 구하는 順序型 複合機械 回路이다.

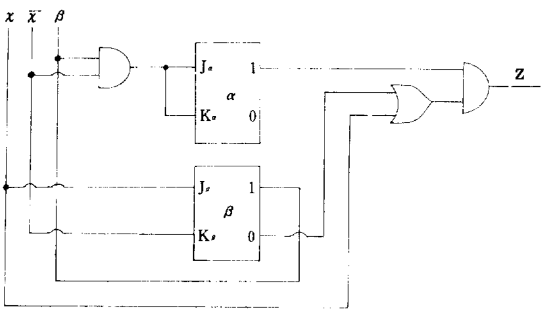


그림 13. 完成된 順序型 複合機械

〈例 2〉 言語理論에의 擴張[6]

이 複合機械 槪念을 言語理論 分野에 擴張 適用하려고 할 경우, 複合機械 槪念의 또 다른 特徵이 드러나게 된다.

앞의 예에서 생각해 본바와 같이, 複合機械 槪念下에 그려지는 狀態遷移圖는 內包된 元素機械의 函數機能이 모두 狀態點 속에 매겨지듯이, 우리는 어떤 그라프에 內包된 意味體들이 頂點에만 매겨지는 定向그라프를 생각할 수 있으며, 이를 點意味型 그라프(node-significant graph)라 부르기로 한다.

點意味型 그라프가 어떤 文章(sentence)을 表現하고 있을 경우, 各個의 頂點은 모두 어떤 意味의 單語(words)로 생각 하는 편이 자연스러우며, 그러할 경우에, 그라프 要素와 文章 要素間의 基礎的인 特性이 잘 어울릴 수 있다.

만약, 點意味型이 아닌 枝意味型 그라프를 쓰게되면, 어떤 單語 要素에 대하여 그라프의 定向枝가 對應해야 하기때문에, 그 定向枝가 거느리는 兩端의 두 頂點도 함께 따라 붙게 되며, 單一의 孤立點에 對應하는 文章 要素는 생각할 수 없게 된다.

더우기, 이 點意味型 그라프는 言語가 갖는 統辭論的 槪念과 意味論的 槪念 間의 學理的 意味區分을 明確히 設定해주는 手段이 되기도 한다.

5 結 論

複合機械의 槪念은 앞의 예로서 考察한 바와 같이, 順序回路 設計를 爲한 매우 솔직한 수단이 되며, 각 단계마다 다른 단계의 작업을 크게 意識하지 않는 單純作業으로 만들어 준다.

예로서 提示된 問題는 다만 論理設計를 위해서 必要로하는 알고리즘만을 엿볼 수 있도록 선택된 것이기 때문에, 매우 단순하며, 복잡한 기계를 設計하는데서 나타나는 특수문제는 介入되지 않는다. 이를테면 入出力 信號線의 數가 많아진다던가, 또는 다음 순서의 처리를 위해서 出力데이타를 기억시켜 놓거나 또는 入力쪽으로 다시 되 돌려 주는 일, 그밖에 시스템 水準의 설계문제 등등이다.

한편으로, 複合機械 槪念의 言語理論 分野에 대한 應用은 매우 廣範囲한 問題이기 때문에, 한마디 말로 日可日否할 수는 없지만, 點意味型 그라프 槪念으로 미루어 짐작할 수 있듯이, 形式言語 體系에도 意味要素를 導入하여서 따질 수 있게 됨으로써 自然語 體系와의 相違點을 解消시킬 수 있다고 본다.

즉, 兩 體系사이에 通用되는 理論이 서로 一致하고 相通할 수 있는 계기가 마련되리라고 생각된다.

〈P64에 계속〉