

論理回路設計におけるトップダウン教育の実現手法

藤井 慎裕[†] 今井 睦俊[†] 阿部 知己[†] 鈴木 二紀[†]
雪田 修一[†] 小池 誠彦[†] 國井 利泰[†]

トップダウン教育とは最終目標を掲示し、それをブレイクダウンする事で教育効果を高める教育方式とそれに必要な教育手法の体系を言う。

このブレイクダウン方式に基づき、セマンティック Web を補完するセル理論を用いたトップダウン教育システムを提案する。現在、論理回路設計において HDL (Hardware Description Language) を用いたモジュール設計が行なわれている。これらモジュールを XML ボキャブラリーで記述できるが、システムが巨大化した場合これらのモジュールの信頼性を保つために、論理的に検証するモデルが存在しない。これを補う方法として、これらのモジュールをセル理論によって構築されたセルデータベース上で実現する。これによりトップダウン方式での論理回路教育の効率化が期待できる。本論文では、これらモジュールを記述する XML ボキャブラリーの基本設計についても詳述する。

1. はじめに

近年、コンピュータサイエンス教育カリキュラムにおいて情報処理の機能階層各々に対応付けた科目の教育順序にトップダウン（上位から下位へ）履修するカリキュラム編成の重要性が見直されるようになった[1]。ここで言うトップダウンによる教育カリキュラムの順序は国内外の大学において実施され効果をあげている[2]。

ここでコンピュータサイエンスにおけるトップダウン教育にインターネットに代表されるネットワークシステムを利用し、教育教材を Web によって提供し目標とする学習成果を達成することが可能なシステムの実現を考えた。本論文では、トップダウン教育の対象として論理回路設計に利用されているハードウェア記述言語(HDL)を用いたハードウェア設計を選び、この設計に必要な情報をセマンティック Web を利用してXMLで表現し、目標とする回路の設計を効率よく行え、学習効果を考えた方法を提案する。

2. トップダウン教育

トップダウン教育とは、学習者の学習意欲を維持し最終目標を達成するための効率化を図る手法をいう。学習の効率化を可能とする方法として、学習の最終目標を明確にし、それを達成するために必要な教育要素に分解(要素分解)し、それらをデータベース化しセルモデルを用いて接合(要素接合)することで最終目標を達成するための最適化された学習方法が得ようとするものである[7]。これによって学習者が学習過程において最終目標を見失うことなく学習意識を維持し目標に到達することが可能となる。

3. 論理回路設計とハードウェア記述言語

トップダウン教育の例としてハードウェア記述言語を用いた論理回路設計の学習について考える。ここでは最終課題(目標)を設定しそれを設計実装する為にそれに必要なモジュール設計(要素分解)を行う。目的達成の為に必要なモジュールの設計、また既に設計されている

モジュールの再利用を行い構築(要素接合)する。

ここではハードウェア記述言語のひとつである VerilogHDL で記述されているモジュールを例に説明する。図 1 では最終目標回路を構成する2つのモジュール A, B に分解し、それぞれのモジュールの設計後最終目標であるターゲットモジュールに接合し回路を完成する例を示した。

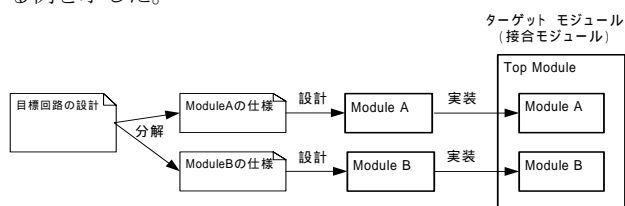


図 1 HDLによるハードウェア設計手順

具体例として最終目標回路をラッチ付き 3 x 8 Decoder (図 2、Module 名: DecoderwithLatch) を設計した。また、この構成要素である Module A (図 3)、Module B (図 4) に分解して設計を行い、Top Module においてインスタンス化することで結合が行える。Module A では可変長ビット幅のレジスタ (Reset 機能付) を構成し、Module B では 3x8Decoder を示す。

```
// Top Module [ DecoderwithLatch(in,clk,rst,out) ]
`resetall
`timescale 1ns/10ps
module DecoderwithLatch(in,clk,rst,out);
  input [2:0] in;
  input clk,rst;
  output [7:0] out;
  wire [2:0] temp;
  register i0(.d(in), //Instance of register
             .clk(clk),
             .rst(rst),
             .q(temp));
  defparam i0.WIDTH = 3;
  decoder3to8 i1(.data(temp), //Instance of decoder
                .out(out));
endmodule
```

図 2 回路要素を接合したトップモジュール

[†] 法政大学大学院 情報科学研究科

```
// Module A [ register(d,clk,rst,q) ]
`resetall
`timescale 1ns/10ps
module register(d,clk,rst,q);
  parameter WIDTH = 4;//default
  input [WIDTH-1:0] d;
  input clk,rst;
  output [WIDTH-1:0] q;
  reg [WIDTH-1:0] q;
  always @ (posedge clk) begin
    if(rst) q <= 0;
    else q <= d;
  end
endmodule
```

図3 回路要素 モジュールA

4. XML を利用した HDL の論理回路設計学習への利用

HDL で論理回路を設計し、これを **FPGA (Field Programmable Gate Arrays)** に実装する。この際使用するライブラリーが既に **FPGA** のメーカーや開発ソフトのメーカーから **HDL** で記述されたモジュールを **XML** で表記し **Web** を利用して提供されている。ここでは **XML** 文書定義として **DTD(Document Type Definition)** を利用し、**EdaXML.DTD** で宣言しているものなどがある[3]。このように **DTD** や **XML Schema** を利用し **HDL** で設計された回路モジュールを **XML** でラッピングし **Web** 上に公開することで論理回路設計の学習に役立てることが可能となってくる。

5. セマンテック Web とトップダウン教育

前述したようにトップダウン教育は、最終目標を明確にし、それを達成するための要素を効率よく利用あるいは設計しこれらを組立てることで実施できる。この要素を **Web** 上に公開しそれを効率よく検索し再利用することで実現でき、これによって学習者が学習過程において最終目標を見失うことなく学習意欲を維持し目標に到達することが可能となる。

```
// Module B [ decoder3to8(data,out) ]
`resetall
`timescale 1ns/10ps
module decoder3to8(data,out);
  input [2:0] data;
  output [7:0] out;
  assign out = dec(data);
  function [7:0] dec;
    input [3:0] data;
    begin
      case (data)
        3'b000: dec = 8'b00000001; //0
        3'b001: dec = 8'b00000010; //1
        3'b010: dec = 8'b00000100; //2
        3'b011: dec = 8'b00001000; //3
        3'b100: dec = 8'b00010000; //4
        3'b101: dec = 8'b00100000; //5
        3'b110: dec = 8'b01000000; //6
        3'b111: dec = 8'b10000000; //7
        default: dec = 8'b00000000;
      endcase
    end
  endfunction
endmodule
```

図4 回路要素 モジュールB

6. セマンテック Web の現状

—XMLによるセマンテック Web の問題—

Web 上に学習要素を公開し利用可能にする場合、**XML** を用いたセマンテック **Web** の利用が有効と考えられる。**XML** は柔軟で、入れ子のペアのブラケットから構成されているが、データ構造が拡大したり入れ子の構成が複雑化し、そのペアのブラケットの入れ子構造に欠落が起きたりした場合の検証メカニズムが不十分であるためデータ構造がばらばらになる恐れがある。[6]

7. サイバー世界

サイバー世界とは、**Web** 上に意図的、あるいは自然に、設計される。また、設計なしで生成される世界群であり、情報だけから成り立つ世界である[5]。多くの場合、サイバー世界は様々なローカルな **Web** サイト上での活動が関連して作られる。情報の観点からサイバー世界は現実世界あるいは仮想的な世界、時にはその両方を指すことがある。サイバー世界は極端なスピードと複雑さで変化を繰り返している為、適切なモデリングなしでは無秩序になり、人のコントロールと理解を超えてしまう事が予想される[5]。よって、サイバー世界の大量なデータを利用できる効率的なデータ操作と複雑な変化に対応できる柔軟性を持ち、共通部分を的確に特徴付けるモデリングを必要とする。このモデリングに **Cellular Model**[5]を導入することにより、従来、リレーショナルモデルのように集合論のレベルで定義され、そのレベルに限定された定義しか出来なかったテーブル間の関係、テーブル内の関連を **homotopy**、**quotient mappings (identification functions)**[4]で動的に構成する事ができ、サイバー世界の複雑な変化に対応し、共通部分を的確に特徴付ける適切なモデリングを可能としたサイバー世界を統合できるデータモデルの提供ができる。

8. セルモデルとリレーショナルモデルとの比較

セルモデルにおいては、**homotopy** 理論を用いる事により、動的にセル空間を構成できる[5]。これはリレーショナルモデルと比べて非常に高い柔軟性を持っている。リレーショナルモデルでのデータの関連性は、テーブルとして表され、テーブル間の関係はデータベース管理者によりデータ依存関係に基づいて正規化され[9]、固定されてしまう。また、リレーショナルモデルでは **homotopy** を保存しないので、テーブルの射影 (**projection**)により、必要な属性を取り出した後はもとのテーブルとの関連性を保つ事ができない。リレーショナルデータベースではキーによって管理を行っており、キーの管理とデータ管理が同義となっているのでデータの入出力時には管理者権限など制約を大きく受けている。これに対して、セルデータベースでセルモデルを基礎として、セル情報を用いてデータ管理を行っているため、柔軟性が高く管理者を必要とせず、利用者にとって自由度と安全性の高いシステムが構築できる。

リレーショナルモデルの場合は、同じテーブルの変形を要求した場合においても要求があるたびにデータ操作を繰返さなければならないが、セルモデルの場合は保存している写像をたどる事により1度使用したセル空間を再利用できるので、効率的な要求処理ができると共に情報を有効利用できる。

サイバー世界には大量のデータが関連しあっているの
で操作量も当然増える。その為、写像の保存によるセル
空間の再利用はサイバー世界においてより有効性が増す
ことが期待できる。

9. セル空間とセル空間構造の構築

セル空間理論とは、抽象階層の概念を取り入れている
ので、階層において継承される不変量を定義する事によ
り、様々な状況をモデル化できる。このセル空間理論に
ついてその概要を以下に示す。

9.1 セル空間構造

セル構造空間(**cellular structured spaces**)についての概
略を説明する。セルはトポロジー的に n 次元の開(**Open**)
ボール $Int B^n$ と同等なトポロジー空間にあり n -cell e^n
と表記する。 X からセル接合により有限あるいは無限の
セル列 X^p を **inductive** に構成することができる。 X^p は
 X の部分空間であるように構成し整数 Z で索引付ける。
この様にして得られる $\{X^p \mid p \in Z\}$ を **filtration** と呼ぶ。
記法では

X^p covers X (or X^p is a covering of X)

すなわち

$$X = \bigcup_{p \in Z} X^p$$

X^{p-1} は、 X^p の部分集合

すなわち、

$$X^0 \subseteq X^1 \subseteq X^2 \subseteq \dots \subseteq X^{p-1} \subseteq X^p \subseteq \dots \subseteq X$$

この様にして X から得られるセル構造空間 $\{X; X^p \mid p \in Z\}$ を **filtration space** と呼ぶ [4]。

9.2 セル結合(cell attachment)によるセル空間構造

Open n-cell e^n を既に構築されたトポロジー空間 X に
surjective かつ **continues** な写像 f により結合することに
よりセル構造空間 Y を構築できる。写像 $f: X \rightarrow Y$ が
surjective であるということは、

$$(\forall y \in Y) (\exists x \in X) [f(x) = y]$$

を意味する。

写像 $f: X \rightarrow Y$ が **continues** であるということは、

“a subset $A \subset Y$ is open in Y if and only if $\{f^{-1}(y) \mid y \in A\}$
is open in X ”を意味する。

$$X \sqcup_f Y = X \sqcup Y / \sim$$

は、接着空間(**attaching space**)と呼ばれる **transitivity** から
同値関係により空間を **equivalence classes** の排他的和
に分割できる。一つの **equivalence class** を x / \sim と表記
すると

$$x / \sim = \{y \in X \mid x \sim y\}$$

である。すべての **equivalence class** の集合を X / \sim と表
記するとそれは X の **quotient space** あるいは
identification space と呼ばれ

$$X / \sim = \{x / \sim \in 2^X \mid x \in X\} \subseteq 2^X$$

である[4]。

9.3 セルラーデータモデルとセル ID

セルラーデータモデルとはセル空間理論を適用したデ
ータモデルである。セルラーモデルは抽象概念の階層に
基づいているため、既存の様々なデータモデルの性質を
包含している。セルは少なくともテーブルとして扱える。
 N 個の属性をもつテーブルは n 個の自由度をもつことか
ら **n-cell** として表わす。リレーショナルデータモデルで

はリレーションスキーマの次数に応じて n 項リレーシ
ョン(**n-ary relation**)と表わしている。セル空間理論には境界
という概念がある。 n 次元の **close-cell** とは境界を持つ
セルであり B^n と表わし、 n 次元の **open-cell** とは境界を
持たないセルであり e^n と表わす。境界とは1次元低い
セルであるので $n \geq 1$ の場合 B^n の境界とは

$$\partial B^n = B^n - e^n = S^{n-1}$$

で表わされ、 S^{n-1} を $n-1$ 次元の球体空間である。セルを
オープンにすると、 $n \geq 2$ の場合には属性が1つのセルに
なり、 $n=1$ の場合には境界は B^0 となり属性を持たない。

セルラーデータベースではオープンセルの属性をリ
レーショナルデータモデルにおける主キーの役割になるよ
うに設計することが可能である。セルラーデータベース
では不変量に基づく同値関係によって、 **Web** 上でセルの
結合が可能である。この時、どのセルと結合するか特定
するためにセル **ID** として **0-cell** を用いる。リレーシ
ョナルデータベースにおけるテーブル **ID** と同等の役割であ
る。

セルラーデータベースではセル **ID** をセルの **0-cell** に書
き込める。これは **0-cell** はどのセルにも必ず含んでいる
もので、 **0-cell** とは独立変数を一つも持たないセルで属
性を示さない。

10. VerilogHDL による回路を XML で表記

ここで図4で示した **Module B** の **3x8Decorder** を例に
XML 表記を考える。 **XML** による構造を図5に示す。

Module ID : モジュール固有の **ID** であるセル **ID**

Module Spec : モジュールの概要仕様

Module Name : モジュール名

Module Type : モジュールの種類。 組込み用モジュ
ール、シミュレーション用モジュールの指定

Port : モジュールの入出力ポートの指定

Signal Name : ポートの名前

Direction : ポートの入出力の指定

Bit Width : ポートのビット幅

Status : 信号の種類指定(**Data,Control,Clock**)

Memo : **Port** に関する補足

HDL Source Code : HDL ソースコード

この構造で **HDL** モジュールを **XML** でラッピングする
と図6のようになる。 **VerilogHDL** で記述する場合、
組込み用モジュールとそれをシミュレーションするため
のテストベンチ用のものが必要となる。図7にそれを示
す。基本構造として1つの組込みモジュールに対して1
つのテストベンチを組み合わせることで **Module** は構成さ
れるように規定した。ここでセルモデルと **XML** で表現
した組み込みモジュールを対比させる。前述したセル **ID**
をここでは **ModuleID** で示す。 **ModuleSpec** は **Module**
の機能説明を行うものである。次にセルラーデータモデ
ルを構成しているものに属性がある。この属性が
<module>...</module>間で示される。 **Web** 上に存在し
ているこのモジュールが利用可能であるかどうかはこの
属性をもとに判断する。このように表現することにより
Web 上でこの「回路モジュール」を一意に特定するこ
とができる。これをセル **ID** としモジュールに書き込み、
管理することで **Web** 上でセルを特定することが可能にな
る。

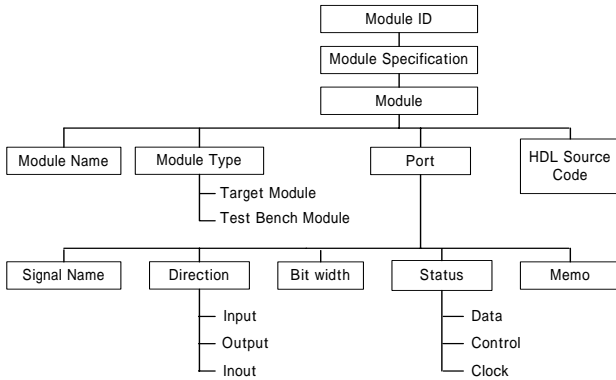


図 5 XML による構造

```

<?xml version="1.0" encoding="Shift_JIS"?>
<verilogmodule>
<moduleID number="131"/>
<moduleSpec>Decoder, input 3, Output 8 </moduleSpec>

<module>
<modulename>3x8Decoder</modulename>
<moduletype>target</moduletype>

<port>
<signalname>data</signalname>
<direction>input</direction>
<bitwidth>3</bitwidth>
<status>Data</status>
<signalname>out</signalname>
<direction>output</direction>
<bitwidth>8</bitwidth>
<status>Data</status>
</port>

<sourcecode>
`resetall
`timescale 1ns/10ps
module decoder3to8(data, out);
input [2:0] data;
output [7:0] out;
assign out = dec(data);
function [7:0] dec;
input [3:0] data;
begin
case (data)
3'b000: dec = 8'b00000001; //0
3'b001: dec = 8'b00000010; //1
3'b010: dec = 8'b00000100; //2
3'b011: dec = 8'b00001000; //3
3'b100: dec = 8'b00010000; //4
3'b101: dec = 8'b00100000; //5
3'b110: dec = 8'b01000000; //6
3'b111: dec = 8'b10000000; //7
default: dec = 8'b00000000;
endcase
end
endfunction
endmodule
</sourcecode>
</module>
</verilogmodule>

```

図 6 組み込み用モジュールの XML 表現

```

<module>
.
.
<moduletype>testbench</moduletype>
.
.
<sourcecode>
: //ここにテストベンチソースコードを記載する。
</sourcecode>
</module>

```

図 7 テストベンチの XML 表現

11. まとめと今後の課題

VerilogHDL で設計したハードウェア設計データをセル理論に基づくモデルを応用し XML スキーマを定義した。これを教育コンテンツを Web 上で効率よく利用できるように検索、取得できるシステムの概要を考えた。

今後、このシステム構成を実装し、既存のインフラである Web サービスシステム上での利用を目標に XML ボキャブラリーの定義、オーサリングツール、ナビゲーションツールなどの研究を進めている。

12. 参考文献

- [1] 大学の理工系学部情報系学科のためのコンピュータサイエンス教育カリキュラム J97 第 1.1 版 (社) 情報処理学会 (1999 年 9 月)
- [2] 國井利泰、1993、コンピュータサイエンスのカリキュラム bit 別冊、共立出版
- [3] "XML EDA Translators" Electronic Tools Company http://www.e-tools.com/content/xml_translators.html
- [4] Toshiyasu L. Kunii: VALID COMPUTATIONAL SHAPE MODELING: DESIGN AND Modeling, Vol. 5, No. 2 (1999) 123-133, December 1999.
- [5] Toshiyasu L. Kunii: A Cellular Web Model - For Information Management on the Web -, September 14, 2001. Corrected and Revised: September 18-20, 2001
- [6] Toshiyasu L. Kunii: Web Information Modeling: The Adjunction Space Model, Proceedings of the 2nd International Workshop on Databases in Networked Information Systems (DNIS 2002), pp.58-63, The University of Aizu, Japan, December 16-18, 2002, Lecture Notes in Computer Science, Subhash Bhalla, Ed., Springer-Verlag, December, 2002.
- [7] 鈴木二紀、國井利泰「サイバー教育と実教育について」、『情報処理学会研究報告』、2002-DBS-126(12)、情報処理学会データベース研究会
- [8] 児玉敏男、國井利泰、「セルモデルに基づいた新しい DBMS の開発 —データ出力の側面から—」、『情報処理学会研究報告』、2002-DBS-126(8)、情報処理学会データベース研究会
- [9] 早川耕二、國井利泰、「セルモデルによる、サイバー世界の情報モデリング」、『情報処理学会研究報告』、2002-DBS-126(14)、情報処理学会データベース研究会