

順序回路の学習支援システムにおけるカウンタ回路の検討

高木 章宏*, 出口 幸子**

Study on Counter Circuits in the Learning System of Sequential Circuits

Akihiro TAKAGI* and Sachiko DEGUCHI**

Abstract

This paper describes the learning system of counter circuits which was developed as an addition to the learning system of sequential circuits for beginners. This system explains each state of counter circuits by synchronizing a circuit diagram, an explanation text, a timing chart, and a state transition table. This system also uses audio recordings for the explanations. This system provides exercises to test the student's comprehension. Nine students used this system to evaluate the system. This system will be introduced to the experiment for the freshmen of the faculty of engineering.

Keywords: Learning System, Counter, Sequential Circuits, User Interface, Questionnaire

1. はじめに

論理回路を学ぶ上で組合せ回路と並んで順序回路は欠かすことができない。組合せ回路の学習支援システムは多く提案されている^{1) 2)}。順序回路の学習ではシミュレータが一般的に使われているが、近年、初学者向けの順序回路のe-learningシステムも開発されている³⁾。筆者らは、初学者に状態の概念を理解させるための順序回路の学習支援システムを開発して実験に導入した^{4) 5) 6) 7) 8)}。筆者らが開発した順序回路の学習支援システムは、以前までフリップフロップを対象としていたが、本研究では学習内容にカウンタ回路を追加し⁹⁾、システムを試用した結果について考察した。本

システムは以前のシステムと同様に、学習システムと確認テストで構成されている。本システムではJK-FFを用いてカウンタ回路を作成しており、クロックが入るたびに各教材を同一画面上に表示して動作を学習する。学習システムで各カウンタについて学習し、理解できたら確認テストに移動して理解の確認を行う。また、本システムを学生に使用してもらい、アンケートを実施した。

2. システムの概要

2.1. 学習コンテンツ

本システムは、回路図、タイミングチャート、状態

*近畿大学大学院システム工学研究科
電子情報システムクラス

**近畿大学工学部電子情報工学科

*Cluster of Electronic Engineering and Information Science,
Graduate School of Systems Engineering, Kinki University

**Department of Electronic Engineering and Computer
Science, Faculty of Engineering, Kinki University

遷移表, および説明文で構成されている. 学習する回路は, 非同期式 16 進カウンタ, 非同期式 5 進カウンタ, 非同期式 10 進カウンタ (BCD カウンタ) である.

2.2. 本システムの全体の流れ

本システムでは, 学習システムで各カウンタの学習を行い, 理解できたら確認テストに移動して, 学習したカウンタの問題を解く. 確認テストで合格になれば次のカウンタの学習を行うことができる. 不合格ならば再度学習を行う. 図 1 に本システムの流れを示す.

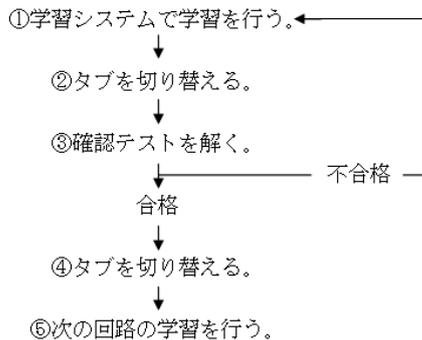


図 1 システムの流れ

2.3. 学習履歴の取得

本システムでは, 学習者の理解度や学習の流れ, および学習時間等を知るために, 以下に示す学習履歴をテキストファイルに記録している.

- ・ 選択されたボタンやアイコンと選択時刻
- ・ 確認テストで選択された解答および解答時刻
- ・ 確認テストの正誤と点数

3. 学習システム

3.1. 複数教材の同期

非同期式 5 進カウンタの学習画面を図 2 に示す. 同一画面上に, 回路図, タイミングチャート, 状態遷移表, および説明文を表示し, 「Clock 入力」ボタンを押すごとに各教材を変化させる. 例えば, クロックが 1 個入った場合 (クロックの立ち下がり時), 回路図では出力が赤で表示され, タイミングチャートでは $Q_0=1, Q_1=0, Q_2=0$ の箇所が赤くなる. 同様に状態遷移表でも対応している箇所が赤くなる. 説明文では対応する説明が表示される.

3.2. 各ボタンの機能

各ボタンの機能について以下に示す.

- ① 「表示」 ボタン: 最初の状態の回路図, タイミングチャート, 状態遷移表, および説明文が表示される.
- ② 「再表示」 ボタン: 現在の状態が再表示される.

③ 「Clock 入力」 ボタン: 1 個の Clock が入力されそれに対応した各教材が変化する.

④ 「前の状態」 ボタン: 1 つ前の状態に戻る.

⑤ 「Clear」 ボタン: 最初の説明に戻る.

⑥ 「次の回路」 ボタン: 次のカウンタの学習に移動する.

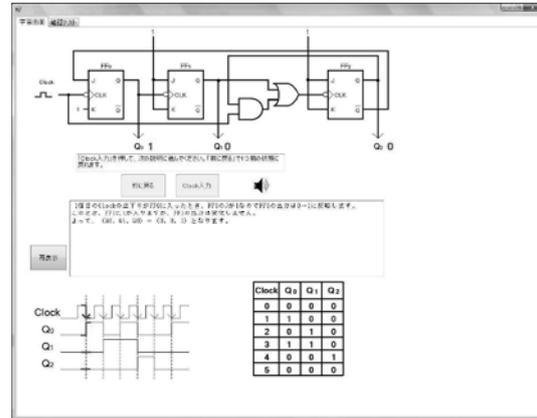


図 2 非同期式 5 進カウンタの学習画面

3.3. 説明文とリンク機能

説明文は, 一般的な教科書を参考に作成した^{10) 11) 12)}. 本システムでは JK-FF を用いてカウンタ回路を作成している. 2^n 進カウンタの例として 16 進カウンタを説明する. 次に N 進カウンタの例として 5 進カウンタをあげ, リセットのための回路を説明する. 最後に $2^n \times N$ 進カウンタの例として 10 進カウンタを説明する.

また, 説明文で初学者にとって理解することが難しいと思われるところにリンク機能を使い, 別ウィンドウでさらに詳しい説明が表示されるようにした. 説明文にある Clock を押したときの画面を図 3 に示す.

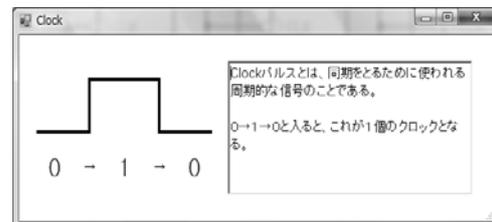


図 3 Clock を押したときの画面

3.4. 音声機能

学習システムの画面の各教材を順に説明している. まず, 回路図を見てもらい状態の変化を説明する. 次に, タイミングチャート, 状態遷移表の順に見てもらいそれぞれ対応した状態のところが変化していることを説明する. また, 音声の説明と同時に説明文が表示される. 何度も繰り返し学習していると音声がいらな

くなることもある。そこで、1 回目の説明のときは変化するごとに音声を再生し、2 回目以降は音声アイコンを押すことで現在の状態の説明が音声で流れるようにした。クリックすると音声の流れ、音声が続いているときにクリックすると一時停止し、ダブルクリックだと停止する。

4. 確認テスト

確認テストは学習システムで学んだことを理解しているかどうかを確認するために作成した。確認テストは学習が終了した時点でタブを切り替えて行う。確認テストは「1」～「3」のボタンで解答して行く問題である。確認テストに合格したら次の回路の学習をすることができる。確認テストで不合格になれば、学習システムに戻り再度学習してもらい、テストの2 回目を行ってもらう。2 回目も不合格であれば正解を確認してもらい、テストの3 回目を行ってもらう。問題数は各回路4 問ずつ（ただし、非同期式5 進カウンタは5 問）である。図4 に非同期式5 進カウンタの1 問目の問題を示す。

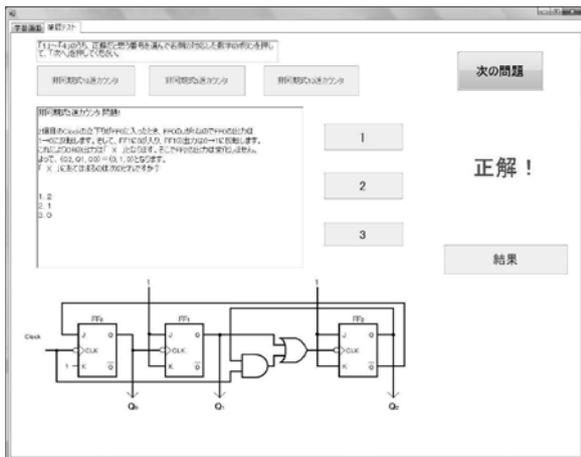


図4 非同期式5進カウンタの1問目の問題

4.1. 確認テストの操作手順

まず、学習した回路が書かれているボタンを選び「次の問題」ボタンを押す。そうすると選択した回路の1 問目が表示されるので「1」～「3」のボタンを押して解答する。解答すると正解・不正解が表示される。また、「1」～「3」のボタンを押した後の解答の変更はできないようにしている。解答して正解・不正解が表示されたら「次の問題」ボタンを押して次の問題へ進んで行く。すべての問題が終われば「結果」ボタンを押す。「結果」ボタンを押すと回路の点数が表示される。問題数が4 問の場合は3 問以上正解（75 点以上）ならば「合格」と表示され、また問題数が5 問の場合は3

問以上正解（60 点以上）ならば「合格」と表示される。上記未満であれば「不合格」と表示される。

4.2. 不合格の処理

確認テストで合格だった場合、「解答」ボタンを押すことで解いた問題の正解を見ることができる。しかし、不合格であった場合「解答」ボタンは表示されず、不合格の回路に対して「再度学習してください」と表示される。学習システムに戻り該当する回路を再度学習してもらい、もう1 度同じ問題を解いてもらう。2 回目も不合格なら「解答」ボタンを表示させ、正解を確認してもらい、もう1 度解いてもらう。

5. システムの実現方法

本システムの開発環境は Windows 7 上で、Microsoft Visual C++2010 を使用している。GUI 作成には Windows フォームアプリケーションを使用している。

5.1. ボタン・タブ・画像・説明文の実現

ボタン、タブ、画像および説明文などを表示させるには機能をツールボックスから選択する必要がある。ボタンの場合は「Button」、タブの場合は「TabControl」を選択する。画像の場合は「PictureBox」、説明文の場合は「RichTextBox」を選択する。そして、ウィンドウに選択したものをドラッグするとウィンドウに表示される。表示されたものをダブルクリックするとコードエディタが表示され、押した時のイベント処理を記述できる。この処理は各ボタンと音声アイコンで行なっている。

5.2. リンク機能の実現

リンク機能は説明文で理解することが難しいと思われる箇所で使用しており、クリックされた時に別ウィンドウを表示するようにしている。

リンク機能を実現するには、まず機能をツールボックスから選択する必要がある。この場合は、「linkLabel」を選択する。そして、ウィンドウに選択したものをドラッグするとウィンドウに表示される。表示されたものをダブルクリックすると押した時のイベント処理を記述できる。別ウィンドウは以下のコードで表示している。

```
Clock^ ClockObj = gcnew Clock0;
ClockObj->ShowDialog0;
```

5.3. 学習システムの同期の実現

学習システムは「Clock 入力」ボタンを押すことによって回路図の出力が変化する。回路図の出力が変化するると共に、他の図表と説明文も回路図の出力に対応

して変化する。以下にこの同期の実現方法を述べる。本システムでは変数の値によって他の図表を変化させている。現在の回路の種類を表している `bangou`、「Clock 入力」ボタンを押した回数を表している `kaisuu` がある。「Clock 入力」ボタンが押されると `kaisuu` の値を 1 増やす。次に `bangou` 内の値によって現在の回路を判定する。`kaisuu` と `bangou` 内の値で表示する状態を決定している。

5.4. 回路図の実現

以前のシステムでは回路図と数値を `Graphics` クラス内のメソッドを使い描画を行っていた。回路図のゲートや配線は、直線や曲線、および円を組み合わせ描画している。直線の描画には `Graphics.DrawLine` メソッド、曲線の描画には `Graphics.DrawEllipse` メソッド、円の描画には `Graphics.FillRectangle` メソッドを使用している。

本システムでは回路図を描画するのではなく、回路図の画像データを作成しておき、プログラムコードで画像を表示するようにしている。また、各出力の数値は `Graphics` クラス内のメソッドを使い描画を行っている。回路図は以下のコードで表示させている。

```
pictureBox1->Image=Image::FromFile("5進カウンタ.jpg");
```

また、数字の描画には `Graphics` クラスの `Graphics.DrawLine` メソッドを使用している。本メソッドを使い、指定した座標に赤色で「0」と表示するコードを以下に示す。

```
grap->DrawString("0", myFont, Brushes::Red, x1, y1);
```

`DrawString` メソッドの第一引数には描画したい数字を入力する。第二引数には描画する線の太さやフォントを指定するオブジェクトを入力し、第三引数には描画される数字の色、第四・第五引数には数字を表示させる `x` 座標・`y` 座標を入力することで指定した数字を描画できる。

5.5. タイミングチャートの実現

以前のシステムではタイミングチャートも回路図と同様に描画を行っていた。本システムでは描画するのではなく、タイミングチャートの画像データを作成しておき、プログラムコードで画像を表示するようにしている。タイミングチャートでは `Clock` が変化するたびに、対応している箇所が赤色になっている画像が表示される。以下のコードで表示させている。

```
pictureBox3->Image = Image::FromFile("16 進タイミングチャート 1.jpg");
```

5.6. 状態遷移表の実現

状態遷移表は `Graphics` クラス内のメソッドを組み合わせて描画している。使用するメソッドは、直線を描画する `Graphics.DrawLine` メソッドと、文字を描画する `Graphics.DrawString` メソッドである。まず、直線を描画する `Graphics.DrawLine` メソッドを説明する。`Graphics.DrawLine` メソッドは指定された `Pen` オブジェクトで 2 つの座標点を結ぶ直線を描画する。以下に直線を描画するコードを示す。

```
grap->DrawLine(blackPen, a1, b1, a2, b2);
```

`DrawLine` メソッドの第一引数には線の太さや色を指定する `Pen` オブジェクト、第二・第三引数には直線の始点となる `x,y` 座標、第四・第五引数は終点となる `x,y` 座標を指定する。以下に、`Pen` オブジェクトのコードを示す。

```
Pen^ blackPen = gcnw
```

```
Pen(System::Drawing::Color:: Black, 3);
```

文字を描画する `Graphics.DrawString` メソッドは数値の表示方法と同じである。以下に指定した座標に黒色で「Q」と表示するコードを示す。

```
grap->DrawString("Q", myFont, Brushes::Black,x1, y1);
```

5.7. 音声の実現

音声は録音したデータが流れるようになっており、音声データは `MP3` 形式で保存している。音声は `mciSendString` 関数を使用して動作させている。以下に音声で使用しているコードを示す。

- ・音声データファイルのオープン

```
mciSendString(_T("open counter10-1.mp3 type mpegvideo "), NULL, 0, NULL);
```

ファイルを開くには `open` コマンドに開くファイル名とデバイスの種類を指定する。`MP3` の場合は `mpegvideo` を指定する。

- ・音声の再生

```
mciSendString(_T("play counter10-1.mp3"),NULL,0, NULL);
```

- ・音声の一時停止

```
mciSendString(_T("pause counter10-1.mp3"),NULL, 0,NULL);
```

- ・音声の再開

```
mciSendString(_T("resume counter10-1.mp3"), NULL,0, NULL);
```

- ・音声の停止

```
mciSendString(_T("stop counter10-1.mp3"), NULL, 0, NULL);
```

- ・音声データファイルのクローズ

```
mciSendString(_T("close counter10-1.mp3"), NULL,
0, NULL);
```

5.8. 確認テストの実現

確認テストはタブを切り替えて行う。確認テストでは初めに、テストを行うカウンタを指定するボタンを押す。このボタンには変数 (ennsyuu) を設定しており、変数の値 (非同期式16進カウンタは「0」、非同期式5進カウンタは「5」、非同期式10進カウンタは「11」)によって各カウンタに対応した問題文が表示される。そして、「次の問題」ボタンを押すことでennsyuuが1ずつ増えていき次の問題が表示されていく。また、確認テストの1問の点数は25点 (非同期式5進カウンタの場合は20点) であり、1問正解するごとに、点を格納する変数に25ずつ足されていく (非同期式5進カウンタの場合は20ずつ)。確認テストを行なっているときに、学習システムへ移動しても操作できないようになっており、以下のコードで実現している。

```
this->tabPage1->Enabled = false;
```

5.9. 再学習の実現

確認テストに合格すれば次の回路の学習を行うことができるが、不合格の場合学習システムに戻り、再学習をしなければならない (次のカウンタの学習には進めない)。結果ボタンを押した時の処理の概要を以下に記す。

- ・合格点であれば以下の処理をする。
 - －点数を表示し、「合格」と表示する。
 - －解答ボタンを表示してユーザに正解を確認させる。
 - －学習システムを実行可能にする。
 - －学習画面に「次の回路」ボタンを表示する。
 - －ユーザに、次の回路の学習に進むように指示する。
- ・不合格点であれば以下の処理をする。
 - －点数を表示し、「不合格」と表示する。
 - －1回目であれば以下の処理をする。
 - ・ユーザに再度学習するように指示する。
 - ・確認テストを実行不可にする。
 - ・不合格回数をセットする。
 - －2回目以降であれば以下の処理をする。
 - ・解答ボタンを表示する。
 - ・ユーザに、正解を確認して再度確認テストを行うように指示する。

5.10. 履歴の取得の実現

本システムでは学習者の理解度、学習の流れ、および学習時間などの学習履歴の取得を行なっている。以下に、履歴取得に使用しているコードを示す。

- ・時間取得の設定

```
DateTime tm = DateTime::Now;
```

- ・ファイルのオープン

```
StreamWriter^ writer = gcnew StreamWriter("学習システムログ.txt",true);
```

- ・ファイルへの書き込み

```
writer->Write("「Clock入力」が押されました。");
writer->WriteLine(String::Format("{0}時{1}分{2}秒
¥n",tm.Hour,tm.Minute,tm.Second));
```

- ・ファイルのクローズ

```
writer->Close();
```

6. システムの試用

本システムは工学部1年生の実験に導入する予定である。そこで、導入する準備として本システムを研究室の4年生9名に使用してもらった。

6.1. 実験方法

システムを実行する前に、簡単なシステムの流れを説明した。本システムでは音声流れるのでヘッドフォンをして学習を行った。本システム終了後にシステムに関するアンケートを実施した。

6.2. 実験結果

確認テストの非同期式16進カウンタ(1回目)の平均点は80.6点であった。また、非同期式5進カウンタ(1回目)の平均点は66.7点で、非同期式10進カウンタ(1回目)の平均点は77.8点あった。5進カウンタや10進カウンタではAND回路やOR回路によるカウンタのリセットが入ったことにより、理解が難しく平均点が下がったと考えられる。

6.3. アンケート項目と結果

本システムを行なった後に4段階評価のアンケートを実施した。以下に、アンケートの内容を示す。

- ① 本システムの各機能 (ボタン等) の使いやすさ
 - ② 説明文の内容の分かりやすさ
 - ③ 音声の有無、複数教材での分かりやすさ
 - ④ 音声の速さ、画面や文字の見やすさ
 - ⑤ 確認テストの難易度 (簡単か)
 - ⑥ 本システムの難易度 (簡単か)・役立ち度・満足度
- アンケート項目全体の平均を求めると3.42であった。平均が高かった項目(3.72以上)は、操作手順のテキストの使いやすさ、ボタンの使いやすさ、および複数教材での分かりやすさであった。平均が低かった項目(3.12以下)は、音声の有無、音声の速度、およびタブの使いやすさであった。以上より、複数教材の評価が高かったので複数教材の利用が効果的であると考えられる。一方、今回のシステムでは音声の評価は

低かったが、フリップフロップの学習支援システムを1年生に使用してもらったときは音声の評価が高かった。つまり、初学者には音声が効果的であるが、4年生はカウンタを既に学んでいるので音声が必ずしも効果的ではないと考えられる。

7. おわりに

本研究では、順序回路の学習の追加としてカウンタの学習支援システムを開発した。カウンタ回路はJK-FFを用いて作成している。本システムは、学習システムと確認テストで構成されている。学習システムでは回路図、タイミングチャート、状態遷移表、および説明文を同一画面上に表示し、各教材を同期させて学習していく。また、初学者にとって理解することが難しいと思われる箇所にリンク機能をつけ別ウィンドウで表示するようにした。また音声による説明を行っている。学習システムでカウンタについて学習し、理解できたら確認テストに移動して理解の確認を行う。また、本システムを研究室の4年生数名に使用してもらい、4段階評価のアンケートを行った。その結果、複数教材の利用が効果的であることがわかった。また、カウンタを既に学んでいる人にとって音声が必要ではないことが示された。

本システムを1年生の実験の授業に導入する予定である。また、非同期式12進カウンタを作成する問題、非同期式10進カウンタの別説明、および同期式と非同期式の違いの説明等の追加を検討中である。

参考文献

- 1) 宮越隆, 松田秀雄, 東真吾, 中嶋芳雄, “論理回路学習支援アプリケーション,” 電気学会論文誌 A, 基礎・材料・共通部門誌, 121(1), pp.78-79, 2001.
- 2) 岸本誠, 金子敬一, “教育用論理回路シミュレータの試作,” 電子情報通信学会技術研究報告 ET, 101(609), pp.61-67, 2002.
- 3) 博多哲也, 大塚弘文, 松本勉, 柴里弘毅, 永田正伸, “補完学習のための基礎デジタル回路学習プログラムの開発,” 全国高専教育フォーラム概要集, O-32, 2011.
- 4) 出口幸子, 植木大介, 高木章宏, “順序回路の学習支援システムに関する研究,” 第36回教育システム情報学会全国大会講演論文集, pp.428-429, 2011.
- 5) 高木章宏, 植木大介, 出口幸子, “順序回路の学習支援システムにおけるUIの機能拡張,” 第62回電気・情報関連学会中国支部連合大会講演論文集, pp. 28-29, 2011.
- 6) 植木大介, 高木章宏, 出口幸子, “順序回路の学習支援システムにおける回路図問題に関する検討,” 情報処理学会 第74回全国大会講演論文集(4), pp.643-644, 2012.
- 7) 高木章宏, 植木大介, 川本真也, 出口幸子, “順序回路の学習支援システムの試用と評価,” 電子情報通信学会 総合大会 学生ポスターセッション予稿集, p.74,

2012.

- 8) 出口 幸子, 高木 章宏, “順序回路の学習支援システムの開発と評価,” 信学技報, ET, Vol.112, N0.66, pp.1-6, 2012.
- 9) 高木章宏, 出口幸子, “順序回路の学習支援システムにおけるカウンタ回路の追加,” 第63回電気・情報関連学会中国支部連合大会講演論文集, p. 422, 2012.
- 10) 大類重範, デジタル電子回路, 日本理工出版会, 東京, 2010.
- 11) 秋田純一, ゼロから学ぶデジタル論理回路, 講談社, 東京, 2003.
- 12) 久保田一, 大石邦夫, 回路シミュレータで学ぶ論理回路入門, オーム社, 東京, 2005.