通信プロトコルと符号化における 基本技術の教育方法と実践

A Method and Its Practice for Teaching the Fundamental Technology of
Communication Protocols and Coding

小林 哲 二^{※1}
Tetsuji KOBAYASHI

The education of information and communication technologies is important for engineering, and it includes terminals, communication media, transmission, switching, software, communication protocols, coding, etc. The proposed teaching method for protocols is based on the HDLC (High-level Data Link Control) procedures using our newly developed software “HDLC trainer”, and includes the extensions for understanding other protocols such as TCP/IP. As for teaching the coding theory that is applied for the error control in protocols, we use both of a mathematical programming language and a general-purpose programming language. We have practiced and evaluated the proposed teaching method in our college, and it is shown that the method has remarkable effects for understanding the fundamental technology of protocols and coding.

Keywords : Engineering Education, Faculty of Engineering, Teaching Method, Information Processing, Communication, Protocol, Coding

キーワード：工学教育，工学部，授業・講義法，情報処理，通信，プロトコル，符号化

1. はじめに

情報通信技術は，端末，通信媒体（有線，無線），伝送（変調，多重化等），プロトコル，符号化，ソフトウェア，および通信系装置などで構成され，プロトコル（通信規約であり，通信プロトコルともいう）は主要な技術の1つである¹⁾。しかし，プロトコルは直接には見えず，かつソフトウェアのように直接に操作できるものでないために，理解することが比較的困難である。従って，プロトコルを理解し易くする教育方法が必要であり，さらには，プロトコルによる通信の信頼性向上に使用されているビット誤り検出・訂正のための符号化²⁾の教育も考慮する必要がある。この論文では，データリンク層の主要プロトコルであるHDLC手順（High-level Data Link Control Procedures）³⁾について，自作の教育用ソフトで学習後に他のプロトコルを拡大比較で学習する教育方法と，数式処理ソフト（Mathematica）⁴⁾と汎用プログラミング言語（Visual Basic）⁵⁾を併用するプログラミング実習で符号化を学習する教育方法を提案し，実際の授業で実践⁶⁾した効果を述べる。

2. 通信プロトコルの教育方法と評価

2.1 プロトコルの概念

プロトコルは，ネットワークを介してノードが通信を行うための通信規約である。ノードは，通信を行う要素であり，例えば，端末，ルータ，交換機，コンピュータなどを表す。プロトコルは，階層構造を有しており，通信メッセージの形式（ビットの割当てや意味など），誤り制御（ビット誤りの検出，訂正，再送），フロー制御（通信量の制御），ルーティング（経路選択），データ圧縮，暗号化，認証，符号化，応用機能などの要素を有する。プロトコルを，ハードウェアまたはソフトウェアで実装することによって，多様な通信機能を有する装置やシステムを実現できる。プロトコルの階層ごとに国際標準や業界標準のプロトコルが定められて

アプリケーション層	アプリケーション層
プレゼンテーション層	
セッション層	
トランスポート層	トランスポート層
ネットワーク層	インターネット層
データリンク層	ネットワークインタフェース層
物理層	（規定なし）

OSI 系プロトコル

TCP/IP 系プロトコル

図1 プロトコル階層構成の対応付け（例）

平成 20 年 11 月 13 日受付

※1 日本工業大学工学部情報工学科

おり、ネットワークアーキテクチャに依存して、プロトコルの階層構成も異なる（図1）。

2.2 基本理解と拡大比較による教育方法

基本理解と拡大比較によるプロトコルの教育方法を述べる。

ステップ1（基本事項の理解）：教育対象の分野の基本事項を選定し、分かり易くする工夫をして、講義とコンピュータ実習によって教育する。

ステップ2（拡大比較による他の事項の理解）：基本事項以外について、拡大比較表を用いて授業を行う。拡大比較表の作成は次の順に行う：(1)基本事項の機能要素を抽出する、(2)基本事項の機能要素を上位概念に対応付ける（機能要素の上位概念は、その機能要素を包含し、かつ他の機能要素も包含できる概念であり、例えばプロトコルにおけるデータリンクの上位概念はコネクション）、(3)基本事項の拡大事項の機能要素を抽出し、それらを基本事項の上位概念と対応付ける、(4)上位概念を項目名として基本事項と拡大事項を拡大比較表にまとめる。

2.3 HDLC手順を基本にする根拠

2.2節の教育方法に基づいて、コンピュータ通信のプロトコルとして基本的存在であるHDLC手順を受講者に詳細に理解させ、他のプロトコルはHDLC手順との類似点と相違点を拡大比較して理解させる教育方法を実践する。HDLC手順を基本的事項とする根拠は、(1)HDLC手順がISO（国際標準化機構）の国際標準のデータリンク層プロトコルの1つであること、および(2)HDLC手順を基にして、パケット交換網のX.25LAP-B、ISDNのLAP-D、LANのLLC、インターネット接続用のPPP、ブロードバンド接続用のPPPoEなどのように、過去および現在の情報ネットワークにおける主要なデータリンク層のプロトコルが開発されており、かつHDLC手順がデータリンク層以外のプロトコルにも大きな影響を与えていることである。

2.4 HDLC手順の教育支援ソフト

HDLC手順の教育支援のソフトウェアとして、「HDLCトレーナ」を自作して、授業で使用する。受講者は、HDLCトレーナを使用することによって、通信動作を順次に確認しながら、HDLC手順の基本機能の仕組みを学べる。HDLCトレーナは、ノード間の「データリンク接続と切断、情報の送受信・監視」などのコマンドの動作を機能ボタンのクリックによって入力し、送信順序番号、および受信順序番号などを表示できる。事前に用意したファイルを読み込んでコマンドを連続実行するスクリプト機能もある。画面には、通信シーケンス（通信メッセージが送受信される様子を時間と共に示す図）がフレーム（通信メッセージ）のコマンドとレスポンス、送信順序番号、受信順序番号などで表示されるので、平易に学習できる。図2は、HDLCトレーナの実行画面例であり、ノードA

図2 HDLCトレーナの実行画面例

表1 HDLCトレーナの機能概要

機能ボタン	機能の説明
SABM (Set Asynchronous Balanced Mode)	データリンク接続要求SABMフレームを送信。
UA (Unnumbered Acknowledgement)	確認応答UAフレームを送信。
DISC (Disconnect)	データリンク切断要求DISCフレームを送信。
I (Information)	データの送信のための I (情報) フレームを送信。
RR (Receive Ready)	受信可能状態を通知するRRフレームを送信。
RNR (Receive Not Ready)	受信不可状態を通知するRNRフレームを送信。
REJ (Reject)	指定の受信順序番号からの I (情報) フレームの再送を要求するために REJフレームを送信。
情報設定	ウィンドウサイズの変更機能。 情報フレームの誤り発生確率の変更機能。
カラー変更	フレーム送信を表示する線分の色を指定。
スクリプト実行	コマンド／レスポンスをノードごとに記述してファイルに設定し、実行ボタンをクリックすると通信シーケンスをアニメーションで表示。
実行画面保存	画面の保存。
クリア	画面の消去。
終了	HDLCトレーナの終了。

表2 HDLC手順と他のプロトコルの拡大比較例

プロトコル 機能要素	HDLC	IPv4	TCP	UDP
プロトコル階層	第2層 (データリンク層)	第3層 (ネットワーク層)	第4層 (トランスポート層)	第4層 (トランスポート層)
コネクション	有り (データリンク)	無し	有り	無し
送信元アドレス	アドレス部が1個	送信元IPアドレス	送信元ポート番号	送信元ポート番号
宛先アドレス	上記と共用	宛先IPアドレス	宛先ポート番号	宛先ポート番号
アドレス長	8ビット×1個	32ビット×2個	16ビット×2個	16ビット×2個
ビット誤り検出	FCS (CRC符号)	チェックサム	チェックサム	チェックサム
誤り検出部	16ビット, 32ビット	16ビット	16ビット	16ビット
送信順序番号	送信順序番号	無し	送信シーケンス番号	無し
受信順序番号	受信順序番号	無し	応答確認番号	無し
再送制御	有り	無し	有り	無し
フロー制御	ウィンドウ制御	無し	ウィンドウ制御	無し
送達確認	有り	無し	有り	無し
バイナリデータの送信	可能	下位のプロトコルに依存	下位のプロトコルに依存	下位のプロトコルに依存
メッセージ単位	フレーム	IPデータグラム	TCPセグメント	UDPセグメント

とノードBの通信において、利用者がコマンドとレスポンスを機能ボタンで入力することによって、SABMコマンドとUAレスポンスでデータリンクが接続し、I (Information) フレームで情報を送信時の送信順序番号Sと受信順序番号Rの変化をI (S, R) で自動的に表示し、送信誤りが発生時にはREJ (Reject) コマンドによって、誤りが発生したI (情報) フレーム以後の情報フレームの再送を要求し、DISCコマンドとUAレスポンスでデータリンクを切断する場合である。フロー制御機能のウィンドウサイズ (相手ノードからの確認の受信が無い状態で送信可能なI (情報) フレーム数、即ち、最大アウトスタンディングフレーム数) もユーザが随時に変更できるので、ウィンドウ制御の学習も可能である (表1)。HDLCトレーナはVisual Basic 言語で作成し、実行プログラムのファイルサイズは約100キロバイトである。HDLCトレーナ初版 (2004年) から改良版 (2007年) への主な追加機能は、REJコマンド機能とウィンドウ制御機能である。

2.5 拡大比較による他のプロトコルの教育

階層の異なる複数のプロトコルでも機能に類似性がある場合には機能比較が可能である。このことに着目して、HDLC手順以外のプロトコルについて、HDLC手順との拡大比較で作成した表を用いる講義を行う (表2)。

2.6 HDLC手順と拡大比較の教育効果の評価

(1) アンケートによる評価

著者が授業を実施した科目は「情報ネットワーク工学」であり (受講者は主に3年生)、以前は資料だけでプロトコルを説明していたが、受講者には理解が比較的困難であったので、基本理解と拡大比較の考えに基づいて、HDLCトレーナを授業で使用する事などによって改善した。図3にHDLCトレーナと拡大比較のアンケート結果を示すが、どの項目も有用性あり

項目	アンケート内容	アンケート評価
1	HDLC トレーナ (初版) による HDLC 手順の理解向上 (2004 年度)	A : 2 人(5%), B : 22 人(49%) C : 15 人(33%), D : 6 人(13%) E : 0 人(0%), (合計 45 人)
2	HDLC トレーナ (改良版) による HDLC 手順の理解向上 (2007 年度)	A : 5 人(9%), B : 30 人(56%) C : 15 人(28%), D : 4 人(7%) E : 0 人(0%), (合計 54 人)
3	拡大比較による HDLC 手順の理解向上 (2004 年度)	A : 1 人(2%), B : 17 人(38%) C : 16 人(36%), D : 11 人(24%) E : 0 人(0%), (合計 45 人)
4	拡大比較による TCP, UDP の理解向上 (2004 年度)	A : 0 人(0%), B : 21 人(47%) C : 14 人(31%), D : 10 人(22%) E : 0 人(0%), (合計 45 人)
【アンケート評価の選択肢】 A : 非常に有用, B : 有用, C : 若干は有用, D : どちらとも言えない, E : 役に立たない		

図3 HDLCトレーナと拡大比較のアンケート結果

(「非常に有用」、「有用」、または「若干は有用」) が多数である。

(2) 練習問題による評価

通常の講義後にテストを行い、さらに、HDLCトレーナと拡大比較による学習を実施後に同じ問題で再度テストを行って、教育効果を調べた。その結果、[2回目全員合計点/1回目全員合計点]は、HDLC手順の

設問では約1.088, TCP / IPプロトコルの設問では約1.083になったので, それぞれ, 約9%と約8%の理解向上がある。

3. プロトコル用の通信路符号化の基礎教育

ビット誤り検出・訂正技術は, 通信路符号化で実現され, プロトコルに適用される技術であるが, 理解することが比較的難しい技術である。そこで, 数式処理と数値計算のプログラミングが可能なMathematicaと汎用プログラミング言語Visual Basicによる実習を併用して, 基礎教育を行う。

3.1 ビット誤り検出・訂正技術の概念

ビット誤り検出・訂正技術は, 元の情報にビット誤りを検出するためのデータを付加(または混合)して, 冗長性を持たせることによって1ビット以上のビット誤りを検出し, さらにビット誤りの訂正を行う技術である(図4, 表3)。

HDLC手順の誤り検出機能FCS(Frame Check Sequence)に適用されているCRC(cyclic redundancy check)符号では, ビット誤りを検出するためのデータを元の情報に付加し, 送信側と受信側で規定の計算を行うことによって, フレームの誤りを検出する。誤り訂正符号では, 情報に冗長性を持たせる符号化を行い, 復号では符号化方式に依存して, 1ビットまたは2ビット以上のビット誤りの検出と訂正を行う。誤り訂正符号には, 例えば, ハミング符号, 巡回ハミング符号, BCH符号, リード・ソロモン符号, 畳込み符号, ターボ符号, およびLDPC符号などがある。ハミング符号では, 1ビットだけのビット誤りを検出し, 訂正できる。巡回ハミング符号は, ハミング符号と機能は同じであるが, 生成多項式または生成行列などによって, 符号語を生成する。これらの符号化方式の内, 基

図4 通信における符号化と復号の概念

表3 基本的なビット誤り検出・訂正技術

分類	基本的な技術の例
検出技術	パリティチェック
	チェックサム
	CRC符号
検出と訂正技術	巡回ハミング符号
	BCH符号
	リード・ソロモン符号
	畳込み符号

礎事項であるCRC符号と巡回ハミング符号は, 数式処理ソフトのMathematicaを用いて, 分かり易く教育でき, かつ, 他の誤り訂正符号を理解するための基礎になる。

3.2 数式処理ソフトによる実習

CRC符号と巡回ハミング符号では, ビット列を多項式に置き換えて計算を行うのが有力な手法の1つである。多項式の係数は, 有限体GF(2)の場合は0または1であり, 係数に mod 2 の演算(2で割った剰余)を行う(例えば, 110010の多項式表示は $X^5 + X^4 + X$)。なお, 例えばリード・ソロモン符号のように, 多項式の係数が拡大体の場合は, 0, 1, a , a^2 , ... が係数(a は生成多項式の根)になり, 生成多項式(係数は a を含む多項式)で剰余を求める演算などを行う。Mathematicaには誤り検出・訂正を直接に計算する機能は無いが, 多項式計算の関数と, 多項式係数に mod 2 の計算を行う関数を組み合わせて多項式計算に使用可能であり, 拡大体の多項式計算にも使用できる。多項式計算の関数には, 例えば多項式Fの係数に mod N (Nで割った剰余)を計算して新たな多項式を得る関数, 多項式PとWについて, PをWで割った剰余の多項式を得る関数などを用いる。

ビット誤り検出・訂正技術のコンピュータ実習として, CRC符号の計算による誤り検出, 巡回ハミング符号によるビット誤り訂正などの実習課題を実施する。

```

(* CRC 符号の Mathematica プログラム例 *)

(* 16 ビット HDLC 手順用 CRC の生成多項式として,
   X^16+X^12+X^5+1 を使用する. *)

(* 送信側 : 情報ビット列 110010 の場合. 多項式 M *)
M = x^5 + x^4 + x
(* 生成多項式 G *)
G = x^16 + x^12 + x^5 + 1
(* M に生成多項式の最高次数を積算. *)
(* Expand 関数は, 多項式の展開を行う. *)
F = Expand[(x^6)*M, x]
(* CRC 符号の計算 *)
(* PolynomialRemainder 関数は, 多項式の剰余を計算. *)
(* PolynomialMod 関数は多項式の係数に mod 2 の計算. *)
CRC = PolynomialMod[PolynomialRemainder[F, G, x], 2]

(* 送信データ T0 として, M と CRC を送信. *)
T0 = Expand[(x^16*M) + CRC, x]

(* 通信中のノイズ Noise を定義する.
   Noise = 0 の場合は誤りなし. *)
Noise = x^16 + x^5 + x^2 + 1
(* 送信データ T にノイズを mod 2 で加算. *)
T1 = PolynomialMod[T0 + Noise, 2]

(* 受信側 : T1 を受信. *)
(* 判定式 J を計算. *)
J = PolynomialMod[PolynomialRemainder[T1, G, x], 2]
(* ビット誤り有無の判定. *)
(* J が 0 ならばビット誤りなし.
   J が 0 でなければ, ビット誤り有り. *)
MatchQ[J, 0]
  
```

図5 CRC符号のMathematicaプログラム例

```
(* 巡回ハミング符号の Mathematica プログラム例 *)

(* 送信側：情報ビット列(例) 1111 の場合、多項式 M *)
M = x^3 + x^2 + x + 1
(* 生成多項式 G *)
G = x^3 + x + 1
(* 生成多項式の最高次数を M に積算. *)
H = Expand[(x^3)*M]
(* H を G で割った剰余 R *)
R = PolynomialMod[PolynomialRemainder[H, G, x], 2]
(* 符号化された送信データ T0 は H + R *)
T0 = Expand[H + R]

(* 通信中のノイズ Noise を定義 (任意の位置). *)
(* Noise = 0 は誤りなしの場合. *)
Noise = x^6

(* 受信側：受信データ T1 *)
T1 = PolynomialMod[Expand[T0 + Noise], 2]
(* シンドローム Syndrome の計算. *)
(* Syndrome によって、誤りパターン ErrorPattern を判定. *)
Syndrome = PolynomialMod[PolynomialRemainder[T1, G, x], 2]
ErrorPattern = Switch[Syndrome,
  0,      0      (*ノイズ 0000000 ビット誤り無し*),
  1,      1      (*ノイズ 0000001 *),
  x,      x      (*ノイズ 0000010 *),
  x^2, x^2      (*ノイズ 0000100 *),
  1 + x, x^3     (*ノイズ 0001000 *),
  x + x^2, x^4   (*ノイズ 0010000 *),
  1 + x + x^2, x^5 (*ノイズ 0100000 *),
  1 + x^2, x^6   (*ノイズ 1000000 *)]
(* ビット誤りの訂正. *)
T2 = PolynomialMod[T1 + ErrorPattern, 2]
(* 訂正結果の検証：送信データが、
   受信データの誤り訂正後の値と等しければ OK *)
MatchQ[T2, T0]
```

図6 巡回ハミング符号のMathematicaプログラム例

表4 CRC符号と巡回ハミング符号の比較例

項目 \ 符号	CRC符号	巡回ハミング(N, M)符号 N：符号長, M：情報長
情報長	自由	固定 (Mビット)
ノイズに対する処理	任意のビット数の誤りを検出可能 (生成多項式に依存)	1ビットだけの誤りを訂正可能
出力	誤りの有無	誤り訂正後の情報

実習課題では、受講者はビット誤りが「無し」と「有り」の両方をプログラミングすることによって、符号化の処理を理解する。表4にCRC符号と巡回ハミング符号の比較を示すが、CRC符号と巡回ハミング符号のプログラムには共通部分が多い。図5にCRC符号、図6に巡回ハミング符号のMathematicaによるプログラム例をそれぞれ示し、(*)で囲まれた部分は「コメント文」、記号^は「べき乗」を表し、プログラムの行ごとの意味をコメントに記述してある。

3.3 汎用プログラミング言語による実習

CRC符号と巡回ハミング符号のVisual Basic言語によるプログラムは、入力、途中経過、および結果を表示して、処理内容を理解し易くしたGUI (Graphical User Interface) 形式のプログラムになる。受講者は、

図7 CRC符号のVisual Basicによる画面例

図8 巡回ハミング符号のVisual Basicによる画面例

実行画面に表示される説明文を読めば意味が分かるようになっており、Mathematicaでは習得できない「汎用プログラミング言語でビット誤り検出・訂正技術をプログラミングする能力」を向上することができるので、例えば、機能を通信系装置に組込む場合などに必要な技術を学習できる。実習課題では、教員がプログラムの一部分を除去したサンプルプログラムを示し、受講者は、その部分を埋めること、および自由な改造を行うことなどを行う。図7はCRC符号、図8は巡回ハミング符号の、それぞれVisual Basic言語のプロ

表5 画面の補足説明

区分	図7 (CRC符号) の説明	図8 (巡回ハミング符号) の説明
入力	1011は $X^3 + X + 1$	同左
入力	1111は情報ビット列	同左
入力	1000000は1の部分にノイズ	同左
出力	生成多項式の最高次数 $k=3$	同左
出力	情報ビット列の長さ4ビット	同左
出力	16進数78は2進数1111000	同左
出力	16進数7は2進数111	同左
出力	16進数7Fは2進数1111111	同左
出力	16進数3Fは2進数0111111	同左
出力	16進数5は2進数101	同左
出力	NG (ビット誤りを検出)	16進数7Fは2進数1111111
出力	—	OK (誤り訂正を正常に完了)

項目	アンケート内容 (2007年度, 受講者数 52人)	回答の 平均値
F1	CRC符号のMathematicaによる実習は, CRC符号の理解に役立ったか。	3.1
F2	CRC符号のVisual Basicによる実習は, CRC符号の理解に役立ったか。	3.5
F3	MathematicaとVisual Basicを併用する実習 は, CRC符号の理解に役立ったか。	3.2
F4	巡回ハミング符号のMathematicaによる実習 は, 巡回ハミング符号の理解に役立ったか。	3.0
F5	巡回ハミング符号のVisual Basicによる実習 は, 巡回ハミング符号の理解に役立ったか。	3.4
F6	MathematicaとVisual Basicを併用する実習 は, 巡回ハミング符号の理解に役立ったか。	3.2
【アンケート回答の選択肢】 5:非常に有用, 4:有用, 3:若干は有用, 2:どちらとも言えない, 1:役に立たない		

図9 符号化実習のアンケート結果

グラムによる実行画面例であり, 表5に各画面の補足説明を上から順に示す。なお, 表計算ソフトなどに付属のVBA (Visual Basic for Applications) もこの実習の開発環境に使用可能なことを確認済みである。

3.4 符号化実習による教育効果の評価

著者が符号化の項目を実施した授業は, 2000年度が「コンピュータネットワークシステム」, 2001～2004年度が「情報処理ソフトウェア」, 2005～2008年度が「情報セキュリティ工学・演習」である (受講者は主に3年生)。

年度 (2000～2008)	'00	'01	'02	'03	'04	'05	'06	'07	'08
受講者数	79	45	32	23	40	52	37	55	28
実習	Mathematica	無	有	有	有	有	有	有	有
	Visual Basic	無	無	無	無	無	無	有	有
正答率 (%)	22.7	41.9	34.4	36.3	58.2	62.3	57.6	55.5	71.1
平均正答率 (%)	22.7	48.5						63.3	

図10 符号化設問の正答率の改善状況

(1) 符号化実習のアンケートによる評価

符号化実習のアンケート結果 (図9) では, 受講者回答の平均値がすべて3.0以上なので, 有用性があると判断できる。なお, Mathematicaでは, 受講者回答の平均値が比較的低い理由は, Mathematicaをこの授業で初めて使用した人が大部分のためと考えられる (Visual Basicは, この授業以前 (2年次) の著者の授業「オペレーティングシステム・演習」で教えている)。

(2) 期末試験の設問の正答率による評価

各年度の実習内容と, 期末試験の符号化設問の正答率 (= [受講者別得点の総和] / [設問の配点 × 受講者数]) によって教育効果を調べる (図10)。符号化設問は各年度で類似の問題である。

この結果, 2000年度「符号化実習無し」の正答率22.7%に比べて, 2001～2006年度「Mathematicaの符号化実習有り」では平均正答率が48.5% (約2.14倍), 2007～2008年度「MathematicaとVisual Basicの符号化実習有り」では平均正答率が63.3% (約2.79倍) にそれぞれ向上している。

4. むすび

通信プロトコルの基本であるHDLC手順を, 自作ソフトウェアのHDLCトレーナなどで受講者が十分に理解してから, 他のプロトコルは拡大比較で学習する教育方法を提案し, 授業で実施した。さらに, 数式処理ソフトと汎用プログラミング言語を用いて, 通信プロトコルのためのビット誤り検出・訂正技術の符号化の基礎教育を行う方法を提案し, 授業で実施した。これらの教育効果の評価を行い, 良好な結果が得られた。

参 考 文 献

- 1) 小林哲二：オペレーティングシステム [OS] 基本技術, 日本理工出版会, 2006
- 2) 今井秀樹：情報理論, 昭晃堂, 1984
- 3) ISO/IEC 13239 : Information technology – Telecommunications and information exchange between systems – High-level data link control (HDLC) procedures, 1997
(JIS規格は, JIS X 5203:1998, Webページ, <http://www.jisc.go.jp>, 参照日 : 2009.1.31)
- 4) Mathematica, Webページ, <http://www.wolfram.co.jp>, 参照日 : 2009.1.31 (MathematicaはWolfram Research Inc.の登録商標)
- 5) Visual Basic, Webページ, <http://www.microsoft.com/japan/msdn/>, 参照日 : 2009.1.31 (Visual BasicはMicrosoft Corporationの登録商標)
- 6) 小林哲二：情報通信技術における基礎事項の教育

方法と実践, 日本工学教育協会第55回年次大会工学・工業教育研究講演会, 講演論文集, 講演番号4-209, pp.260-261, 2007

.....

著 者 紹 介

小林 哲二

現 職 : 日本工業大学

工学部情報工学科教授

学 位 : 工学博士

専門分野 : 情報工学

所属学会 : 日本工学教育協会, 電子情報通信学会, 情報処理学会, 電気学会, 映像情報メディア学会, 教育システム情報学会, 米国IEEE, その他.

職 歴 : NTT (日本電信電話) の研究所に勤務して, 研究・開発に従事した.