

プログラミング教育のファーストステップ —アンプラグドで学ぶプログラムの役割—

First Step in Programming Education — Learning the role of program in an unplugged —

梅野 貴俊

後藤 栄太

Takatoshi UMENO
技術教育ユニット

Eita GOTO
大学院教育学研究科

(令和元年9月30日受付, 令和元年12月12日受理)

「プログラミング的思考」を育むと同時に「プログラムの役割」を理解することを目的として、小学生を対象としたアンプラグドプログラミング教育を実施した。授業では、様々な課題に対して、その解決法を考え、課題解決に必要な組込みシステムの設計を行う教育手法^{1,2)}を用いた。日常的に生じる4つの課題解決に取り組み、学年(年齢)、性別、課題の難易度が、課題解決(正解)に必要な時間に及ぼす影響を調査した。教材は組込みシステムのしくみを簡単に理解でき、システムの基本構成を設計できるPIECE教材³⁾を用いた。この結果、すべての課題が終わるまでの時間は、小学1年生が最も長く、学年が上がるとともに、その時間が短くなり、小学4年生以上の学年では、課題解決までの時間に大きな差が認められないことが明らかとなった。「プログラミング的思考」を育てる授業計画として、小学1～3年生では、課題解決への様々な方策とその流れを学び、小学4～6年生では、情報を論理的に処理する考え方を積極的に取り入れることが望ましいと考えられる。

1. 緒言

IT業界の急速な発展により、コンピュータで使用するアプリケーションの開発、組込みシステムの制御、WEB関連のシステム開発やゲーム開発など、情報技術を必要とする仕事内容が急増している。このため、プログラマーやシステムエンジニアの需要が高まるとともに、この人材を育成する環境の充実が求められている。これら背景より、学校教育においても、情報教育が推進され、中学校技術・家庭科(技術分野)では、2012年より「プログラムによる計測・制御に関する内容」が必修化された⁴⁾。さらに、2020年より小学校におけるプログラミング教育が必修化された⁵⁾。

中学校技術・家庭科技術領域で実施する情報教育「プログラミングによる計測と制御」に関する研究は盛んに行われ研究報告も多い。これら研究では、教材の開発、教育手法や授業実践報告など、基礎的内容から授業実践まで、その内容は多岐にわたっている。一方、小学生を対象としたプログラミング教育に関する研究報告では、プログラミング教育の意義や課題に関する研究報告が多く、プログラミングに関する教育手法や授業実践を通じた教育効果の検証など、プログラミング教育の実践に関する研究報告は少ない。

これまでに筆者らは、小学校教員を目指す大学1年生の学生を対象に、組込みシステムの設計手法を用いたアンプラグドプログラミング教育を実施した²⁾。この授業は「プログラミング的思考」を育むことを目的とし、ある課題に対して、その解決に必要な入力、処理、出力を学び、PIECE教材を用いて設計させるものである。この結果、すべての学生において、組込みシステムの入力、処理、出力の流れ、処理(論理回路)の内容を正確に理解することができ、さらに、オリジナルの組込みシステムを構想し、設計することも可能になった。本研究では、この教育手法を用い、小学生を対象とした、プログラミング教育を実施した。

学年（年齢），性別，課題の難易度が課題解決時間に及ぼす影響を明らかにし，アンプラグドで行うプログラミング教育の効果について考察した。

2. 利用した教材

本研究では，組込みシステムのしくみを簡単に理解でき，システムの基本構成を設計することが可能なPIECE教材（エレキット，株式会社イーケイジャパン）³⁾を用いた。この教材はパソコンを使用することなく，入力，処理（論理回路），出力モジュールをパズルのように組み合わせ，希望する様々な動作を実現することができる。入力，処理，出力モジュールの接続例を図1に示す。入力，処理，出力モジュールを左より順に接続し，電源を入れると動作が開始される。図1では，光センサが光に反応してON信号が発生する。しかしながら，このシステムには光センサに論理否定を接続しているため，暗くなるとON信号が発生して音がでる。

入力，処理，出力モジュールの仕様を表1に示す。入力モジュールは振動，音，光を感知する3つのセンサ，処理はNOT（論理否定），AND（論理積），OR（論理和），TIMER（時間制御），CONTROL（動作制御），出力はLED（光），VIBE（振動），MELODY（音），MOTOR（モータ駆動）で構成される。なお，出力モジュールの接続は1つだけとなる。本授業では，処理モジュールのTIMER（時間制御）とCONTROL（動作制御）は使用していない。

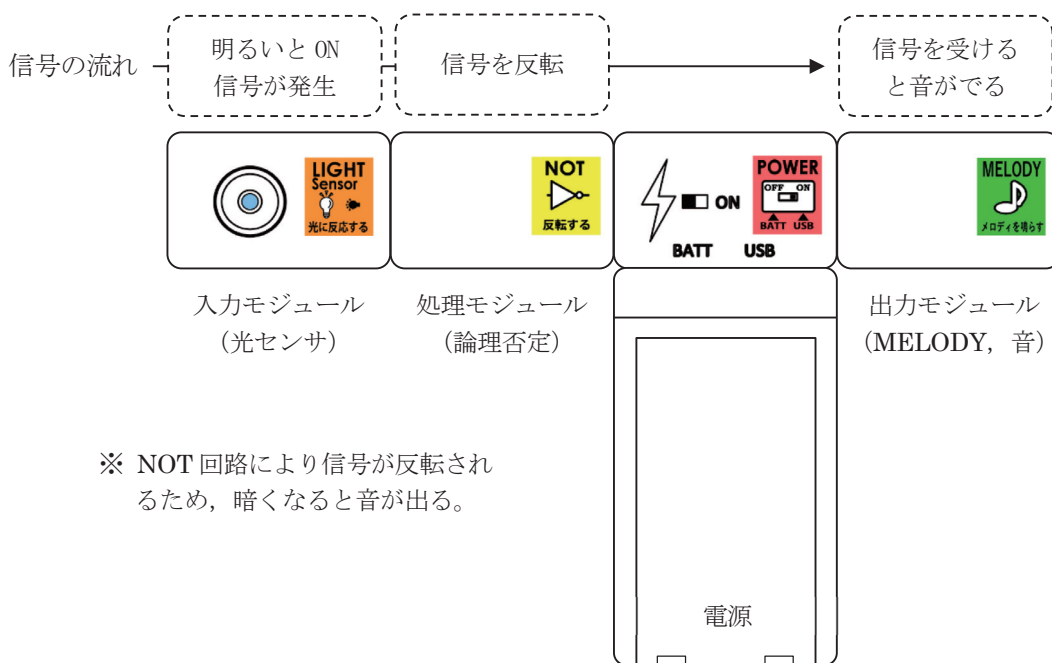


図1 入力，処理，出力モジュール接続の一例

表1 入力，処理，出力モジュール一覧

入力	処理 (論理回路)	出力 (動作)
振動センサ	NOT (論理否定)	LED (光)
音センサ	AND (論理積)	VIBE (振動)
光センサ	OR (論理和)	MELODY (音)
	TIMER (時間制御)	MOTOR (モータ)
	CONTROL (動作制御)	

3. 授業実践

被験者は、福岡県I市が主催する教育文化イベントに参加した48名（小学1年生8名、2年生5名、3年生10名、4年生6名、5年生9名、6年生5名、中学生5名）であった。本授業では、児童1名に対し1人の教師を配置し、マンツーマンで授業を行った。教師は本学中等教育教員養成課程の大学4年生2名、本学大学院中等技術専攻の大学院1年生1名、大学教員1名の合計4名であった。

身近な家電製品を題材として、組込みシステムの動作原理を、入力、処理、出力の言葉を取り入れ説明した（10分程度）。次に、PIECE教材の使い方、各モジュールの役割を説明した後、システム動作の異なる4つの課題に取り組んだ。課題内容を表2に示す。各課題が求める動作に適した入力、処理、出力を考え、PIECE教材を用いて各モジュールを選択し組み立てた後、スイッチを入れて目的動作と同じ動作ができているか確認した。目的動作となるまで、トライアンドエラーを繰り返し、各課題終了（システム完成）までの時間を計測した。児童一人ですべての課題に取り組むが、児童からの質問があれば、すべて答え適宜助言を与えた。4つの課題内容を表2に示す。紙芝居と寸劇で課題内容を説明し、動作確認も寸劇で確認した（図2参照）。

表2 4つの課題（必要とする動作）

課題	内容	必要とする動作
1	浦島太郎が玉手箱を開けようとするとき警告する（玉手箱の中にシステムを入れ、蓋を開けようとするときブザーで警告する）。	明るくなると音（ブザー）が鳴る。※入力に光センサ、出力に音モジュールを使用する。
2	ゴミ箱をあさるカラス撃退する（ゴミ箱の蓋にカラスがのると警告する）。	振動もしくは音を検出すると音が鳴る。※入力に振動および音センサ、処理にOR（論理和）、出力に音モジュールを使用する。
3	街灯のしくみをつくる（夜になるとライトが点灯する）。	暗くなるとLEDが点灯する。※入力に光センサ、処理にNOT（論理否定）、出力にLEDモジュールを使用する。
4	犬の自動餌やり装置をつくる（犬が“ワン”と吠え、同時にお手を行うと、餌を運ぶモータが作動する）。	振動と音を同時に検出するとモータが動く。※入力に振動および音センサ、処理にAND（論理積）、出力にモータモジュールを使用する。



a) 紙芝居を用いた課題の説明



b) 課題2の動作確認（ゴミ箱とカラス）

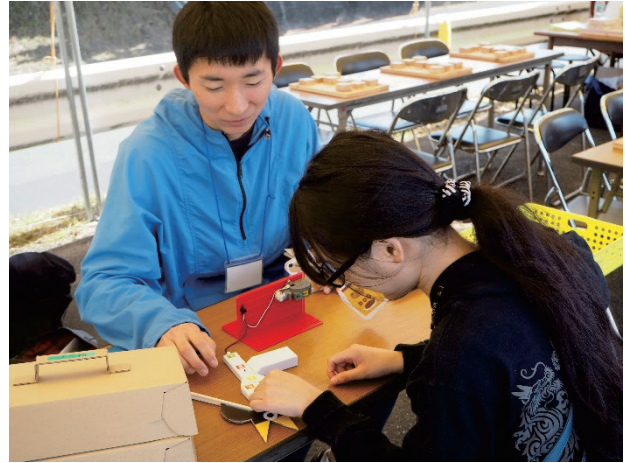
図2 紙芝居による課題説明と寸劇による動作確認

4. 結果と考察

授業に参加したすべての児童において、課題解決に積極的に取り組む姿勢が見受けられ、4つの課題すべてを完成させることができた(図3)。課題1~4に対応したモジュールの組み合わせを図4に示す。課題1は、入力(光センサ)モジュールと出力(ブザー)モジュールの簡単な組合せであるが、課題2以降は、処理モジュールが必要となるなど、使用するモジュールの数が増える。



a) 課題4に取り組む男子児童



b) 課題4の動作確認を行う女子児童

図3 課題に取り組む児童

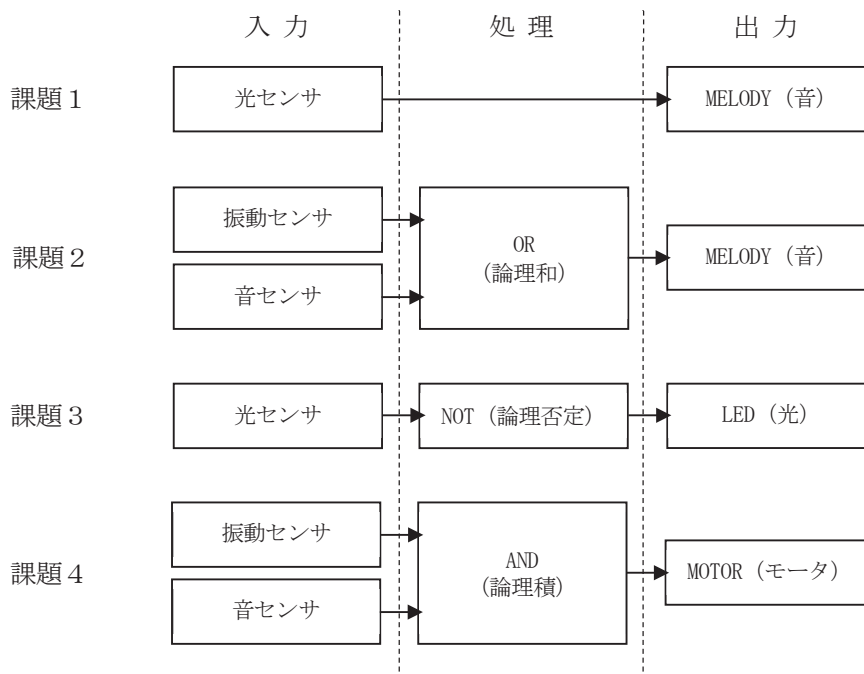


図4 課題1~4に対応した入力, 処理, 出力モジュールの組み合わせ

課題1~4においてシステムが完成(解答)するまでの平均時間を図5に示す。課題1~4における解答時間の平均値と標準偏差は、それぞれ 66.9 ± 34.8 , 89.5 ± 44.3 , 91.3 ± 36.8 , 81.6 ± 42.4 秒であった。課題1は、入力(光センサ)モジュールと出力(ブザー)モジュールの簡単な組合せであるが、課題2以降は、処理モジュールを取り入れなければならない。この処理モジュールの選択に時間を必要としていた。特に、

OR および AND モジュールの選択において、多くの児童が間違っていた。これら処理モジュールを間違えるとまったく異なる動作となる。課題3において、入力と出力モジュールの選択はすぐにできるが、NOT モジュールを選択できない児童が多数見受けられた。図6は、すべての課題が終わるまでの時間を各学年別で比較したものであり、学年が上がるとともに解答時間が減少していた。本授業では、システム完成に要する時間の男女差は認められなかった。1年生から6年生までの平均時間はそれぞれ、 487.6 ± 68.5 , 389.3 ± 111.3 , 324.9 ± 76.9 , 274.3 ± 72.9 , 270.7 ± 68.3 , 262.2 ± 78.3 秒であった(図7)。なお、中学生は 263.4 ± 83.8 秒であった。学年が上がるとともに、システム完成に要する時間は短くなるが、小学4年生以降中学生までは、大きな違いが認められないことが明らかとなった。

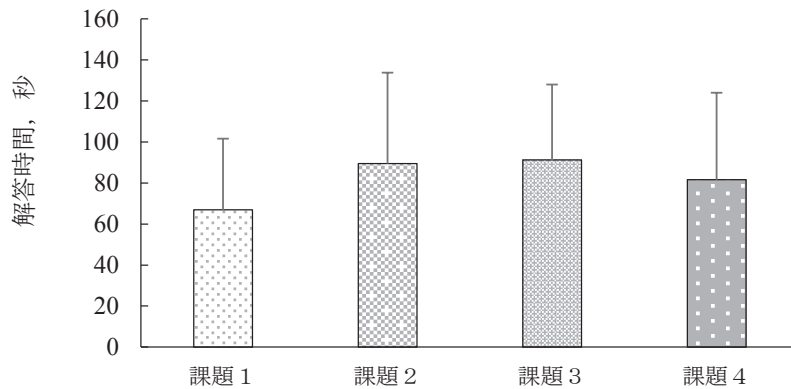


図5 4つの課題の解答時間

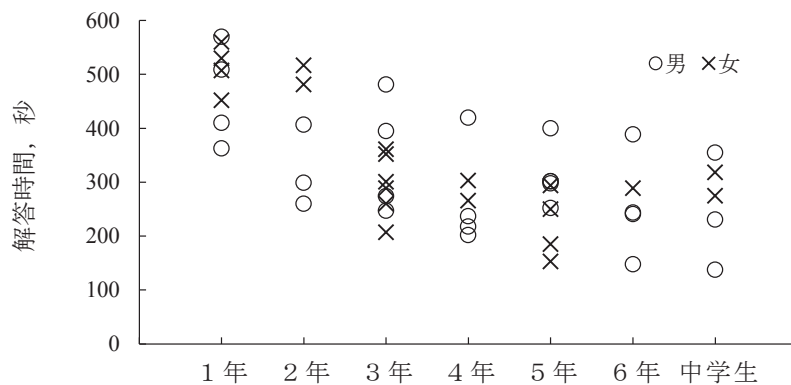


図6 全課題の解決に要した時間の学年比較

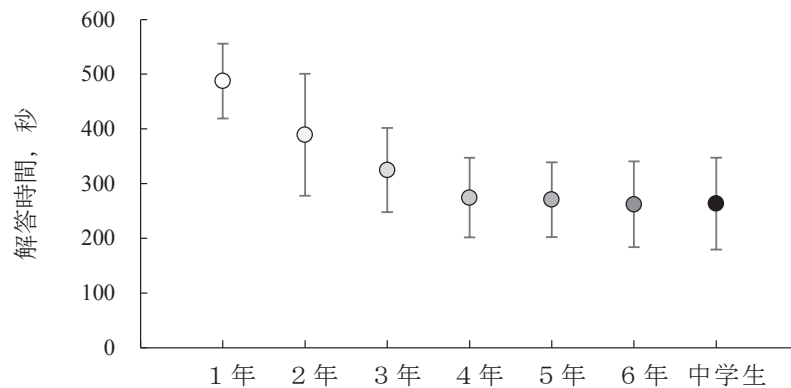


図7 全課題の解決に要した平均時間の学年比較

本授業は、身近な生活の中にある課題を取り上げ、その解決方法を考えシステムの基本形を完成させるものである。児童が完成させたシステムは、警報器や家電製品等、身近にある様々な装置等に組み込まれるコンピュータシステムの基本形と同じである。小学生を対象としたプログラミング教育は、プログラミングスキルの向上を目的としたものではなく、身近な問題に対して、どのように解決すべきかを考え、解決を目指す「プログラミング的思考」を育てることが重要である⁶⁾。本授業で使用した教材は、一つ一つのモジュールに特定の動作が存在する。入力、処理、出力モジュールの動作と役割を正しく理解し、一つのシステムとして組み上げる過程は「プログラミング的思考」を育むと考えられる。特に処理モジュールの正しい理解と選択は、各モジュール間の信号の流れの理解につながると考えられ、信号の流れを制御するプログラムの役割を理解することができる。また、実際にモジュールを組み合わせて、動作確認を行うことによって、目的とした動作となるかを即座に確認することができ、間違いを簡単に見つけることができる。このトラブルシューティングとトライアンドエラーを重ねることも、プログラミング教育において重要な内容である。本授業のような組込みシステムを完成させる教育手法は、アンブラグドプログラミング教育の一つであり「プログラミング的思考」を育てる有効な方法であると考えられる。

本研究では、小学4年生以降、課題解決までの時間が変わらないことが明らかとなった。すなわち、小学4年生以降では、論理回路を短時間で理解できると考えられる。小学4～6年生を対象とした授業では、様々な教科において、情報を論理的に処理する考え方を積極的に取り入れ、「プログラミング的思考」を育てるような授業展開を行うことが望ましいと考えられる。

プログラミングの本質を理解するには、パソコンを用いた計測と制御実習が効果的である^{7,8)}。すなわち、実際に制御装置に触れて動かすことが重要である。しかしながら、小学生にテキストエディタによるプログラミング学習を行うことは難しく、パソコンや制御教材など、プログラミング学習に必要なハード、ソフトウェアが充実している小学校も少ない。小学生を対象としたプログラミング教育では、アンブラグドプログラミング教育を導入し、「プログラミング的思考」を育むと同時に「プログラムの役割」を理解させることが望ましいと考えられる。また、中学校技術・家庭科技術領域で実施する情報教育「プログラミングによる計測と制御」を始めるにあたり、生徒の興味・関心をよびおこさせるための準備的段階およびプログラムの流れや役割の理解を深めるものとして、本研究のようなアンブラグドプログラミング教育は有効であると考えられる。ただし、アンブラグドプログラミング教育は、プログラミング教育の導入および基礎学習としては有効であるが、プログラミング教育の入り口にすぎないことを教育者は理解しなければならない。

本授業実践では、児童1人に対し教師1人のマンツーマン体制で実施しており、通常の学校教育における授業体制とは異なる。本授業では、組込みシステムの動作原理の説明から、すべてのシステムが完成（課題解決）するまでの平均時間は約15分であった。児童からの質問に応え、適宜助言を与えることで理解が進み、短い時間で目的を達成できたと考えられる。今後、本教育手法を学校教育へ応用し、その教育効果について検証する予定である。

5. 結言

パソコンを使用せずに組込みシステムのしくみを簡単に理解できるPIECE教材を用い、小学生を対象としたアンブラグドプログラミング教育を実施した。「プログラミング的思考」を育むと同時に「プログラムの役割」を理解することを目的として、日常的に生じる4つの課題解決に必要なシステムを考え構築させた。この結果、すべての児童において、組込みシステムの入力、処理、出力の流れ、処理（論理回路）の内容を理解し、4つの課題を解決（システムを完成）させることができた。4つの課題が終わるまでの時間は、小学1年生が最も長く、学年が上がるとともに、その時間が短くなることが明らかとなった。さらに、小学4年生以上の学年では、問題解決までの時間に大きな差が認められないことが明らかとなった。「プログラミング的思考」を育てる授業計画として、小学1～3年生では、課題解決への様々な方策とその流れを学び、小学4～6年生では、情報を論理的に処理する考え方を積極的に取り入れることが望ましいと考えられる。

<謝辞>本研究成果は、福岡教育大学教育総合研究所研究プロジェクト助成事業によるものである。ここに記して感謝の意を表する。

参考文献

- 1) 梅野貴俊, 白石正人: 組込みシステムの設計手法を用いたプログラミング教育, 福岡教育大学紀要 68号 (第3分冊): 41-46, 2019.
- 2) 原未希子, 下戸健, 梅野貴俊ほか: 組込みシステムをテーマとした学士力育成ロボット教材の開発, 日本産業技術教育学会第27回九州支部大会: 103-104, 2014.
- 3) (株) イーケイジャパン: つなぐ・つくる・まなぶ PIECE [ピース], <https://www.elekit.co.jp/piece/> (令和元年9月30日確認済み).
- 4) 文部科学省: 小学校学習指導要領 (平成29年3月告示).
- 5) 文部科学省: 中学校学習指導要領 (平成20年7月告示).
- 6) 赤堀 侃司: プログラミング教育における論理的な思考とは何か, 学習情報研究論文誌 261-4: 56-31, 2018.
- 7) Takatoshi Umeno, Takeshi Shimoto: Teaching Material Imitating the Advanced Driver-Assistance System for Measurement and Control Education, Journal of Robotics and Mechatronics, Vol.31-3: 412-418, 2019.
- 8) Takeshi Shimoto, Chika Miyamoto, Takatoshi Umeno: Development of the Biological Information Measurement System for STEM Education and High School/University Articulation, Journal of Robotics and Mechatronics, Vol.31-4: 594-602, 2019.

