

教材用自律型ロボットの改良とその評価

Improvement and Evaluation of an Autonomous Mobile Robot as Teaching Material

紅 林 秀 治*・室 伏 春 樹†

Shuji KUREBAYASHI and Haruki MUROFUSHI

（平成20年10月6日受理）

In previous studies, we have developed an autonomous mobile robot as teaching material for all students. However, the robot had three problems, which were that the motors were unable to rotate in opposite directions at the same time, the circuit board didn't save the programs and the circuit board did not have enough memory to save the programs. These controversial issues were pointed out by students during classes. To solve these issues, the microcontroller and firmware that were used in the circuit board were improved. As a result, we solved these issues without changing the circuit board and the improvements were appreciated by students.

1 はじめに

中学校「技術・家庭」では、「生活に必要な基礎的な知識と技術の習得を通して、生活と技術のかかわりについて理解を深め、進んで生活を工夫し創造する能力と実践的な態度を育てる。」と学習指導要領¹⁾の目標で掲げている。目標の中にある「生活」とは家庭生活だけでなく、学校や地域、社会など様々な場面を意味している²⁾が、今日の生活や科学・技術の発展において、コンピュータが欠かせない存在になっている^{3) 4)}。

筆者らは、中学校「技術・家庭」の学習領域「情報とコンピュータ」²⁾で扱う学習用教材として、モータを2個制御する（以下「2モータ制御」と呼ぶ）ことが可能な自律型制御ロボット基板を開発し、実践を行ってきた⁵⁾。プログラムがロボットの動作により確認できるため、実感を伴いながらプログラミング学習や自動制御の学習ができるという教育効果を小中学校における実践から確認できた^{6) 7)}。ところが、筆者らが使用した教材には、以下の問題点が学習者から指摘された。

1. 左右モータを同時逆回転ができない。
2. 不揮発性メモリーに生徒が作る制御プログラムを保存できない。

3. プログラムを保存できる記憶容量が少ない.

上記3点の問題は、使用している制御基板の問題であり、技術的に改良可能であるか検討を行った。その結果、回路基板を変更せず、マイクロコントローラとファームウェアの変更だけで問題点を解決できることがわかった。筆者らの行った改良方法の利点は、既存の制御基板をそのまま活用できるため、改良に伴うコストがほとんどかからないことである。ファームウェアの更新による製品改良は、様々な家庭用電化製品で行われている。ところが、教育用制御ロボット教材の実践が多く報告^{8) 9)}されてはいるものの、学習者から指摘された教材の問題点をファームウェアやマイクロコントローラの変更で対応した研究例は見あたらない。そして、改良した基板による制御教材の評価を、改良前の基板を使用し学習した児童に対し行った。本論文では、改良前の制御基板を使用した教材、改良の方法、評価実験の順に述べる。

2 自律型制御ロボット教材について

2.1 制御基板とロボットの材料について

使用した制御用の基板を図1(a)に示す。基板に採用したPIC12C672¹⁰⁾というマイクロコントローラには、CPUの他に、揮発性メモリー (RAM)、不揮発性メモリー (ROM)、入出力ポート (I/O) が搭載されている。ロボットは、制御プログラムをパソコンから転送することで、自律走行が可能になる。制御プログラムは、命令ごと1バイトの中間コードに変換し、インターフェースを通してPIC12C672の揮発性メモリーに送られる。

PIC12C672の不揮発性メモリーには、パソコン上で記述されたロボット制御命令の中間コードを解釈実行するファームウェアが書き込まれている。揮発性メモリーには39ステップ^{*1}のプログラムと、7ステップのサブルーチンプログラムを2個記憶することができる。

PIC12C672の出力ポートには2個のモータをそれぞれ正逆転可能な制御用IC¹¹⁾を接続している。モータを介して左右の車輪を回転させる機構を持つロボットにこの制御用基板を組み込んだ場合、前進、後進、および旋回運動を実現できる。また、PIC12C672の入力ポートにマイクロスイッチなどのセンサーを接続することによって、外部との接触状態を検出できる。図1(b)にプログラム転送用のインターフェースを示す。中間コードに変換されたプログラムは、EIA-232規格の端子からインターフェースに送られ、そこで赤外線信号に変換する。変換された信号は、赤外線発光部から発信し、基板上の赤外線受光部より受け取る。制御用基板を取り付けたロボットとロボットに使用した部品を表1と図2に示す。図2の1から8の数字は表1の部品番号に対応している。

2.2 制御プログラムについて

プログラミング言語にはドリトル^{12) 13)}を採用した。ドリトルは教育用に設計された言語であり、簡潔な日本語でプログラムを記述できるのが利点である。学習者は、ドリトルで作成したプログラムをロボットに転送して、ロボットを制御する学習を行った。図3(a)に制御プログラムの例を示す。

*1 1ステップは命令+引数

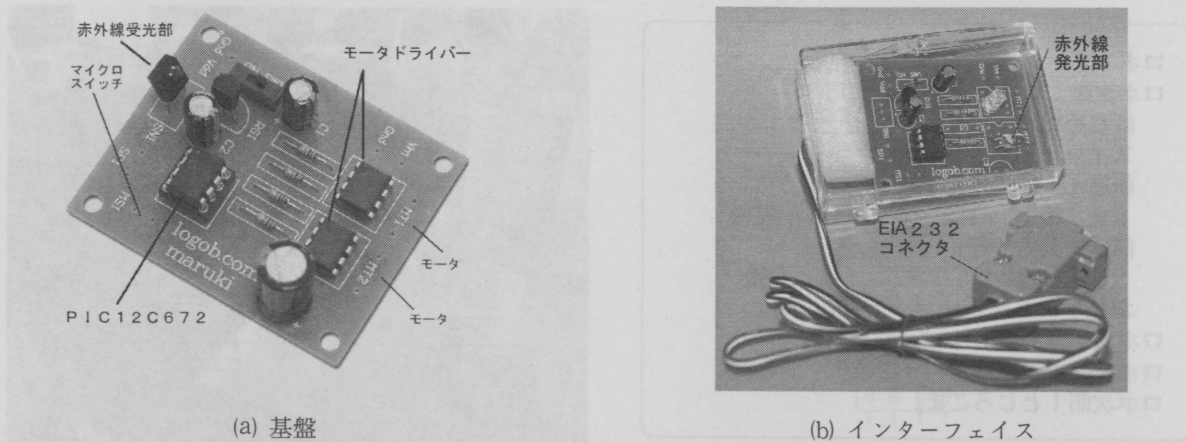


図1 基盤とインターフェース

表1 主な使用部品

No.	使用材料	規格
1	田宮ツインギアボックス	ITEM 70097
2	田宮ユニバーサルプレート	ITEM 70098
3	田宮 TRACK & WHEEL	ITEM 70100
4	電池ボックス	単3×2本用
5	電池スナップ	006P用
6	圧電スピーカ	PKM11-4A0
7	006P電池ホルダー	006P用
8	マイクロスイッチ	SS-5GL2
9	制御用基板	logob.com

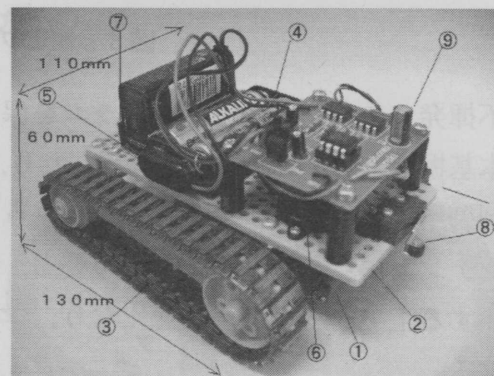


図2 ロボット製作例

このプログラムでは、“ロボ次郎”という名前の通信オブジェクトを生成し、その内部に“転送命令”という名前でロボットに命令を送るメソッド（オブジェクト内部に記憶する小さなプログラム）を定義している。このメソッドは、通信ポートを開いた後で実行される。

転送される命令により、ロボットは次の動作を行う。

- ・「スイッチスタート」により、スイッチが押されたら実行を開始する。
- ・「前進・入力で停止」により、スイッチが押されるまで前進する。
- ・何かにぶつかったら「10 後退 15 右前 15 左後」により後退し、左に向きを変える。
- ・「前進・入力で停止」により、再びスイッチが押されるまで前進する。
- ・何かにぶつかったら「10 後退 15 右前 15 左後」により後退し、右に向きを変える。

図3(b)は、制御プログラムでロボットの動作を確認する児童の様子である。

3 改良の方法

本制御教材を利用して、授業実践を試みた結果、以下の問題点が明らかになった。

1. 左右モータを同時逆回転ができない。

旋回命令は片方のモータを回転させることで実現していた。しかし、学習者が望んでいる旋回命令は左右モータを別々の方向に回転させるものであった。

```

ロボ次郎＝シリアルポート！作る。
ロボ次郎：転送命令＝「！
はじめロボット
スイッチスタート
前進・入力で停止
10 後退 15 右前 15 左後
前進・入力で停止
10 後退 15 左前 15 右後
おわりロボット」。
ロボ次郎！"com1" ひらけごま。
ロボ次郎！転送命令 うごけ。
ロボ次郎！とじろごま。

```



(a) 制御プログラム

(b) 児童の様子

図3 制御プログラムと児童の様子

2. 不揮発メモリーに制御プログラムを保存できない。

本基板は一度プログラムを実行したり、電源を切ったりするとRAM内に格納されたロボット制御用プログラムが消去されてしまうため、再度転送しなくてはならない。このようなアクシデントに見舞われる学習者の姿を、筆者らは実践中に数多く見かけた。これは、授業を実践する上で、学習効率が悪くなり、学習者が制御プログラムの学習においてつまづく要因になる。

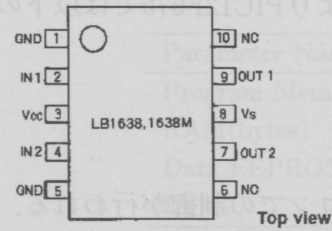
3. プログラムを保存できる記憶容量が少ない。

本基板ではメインルーチン領域として39ステップ+サブルーチン領域として7ステップ×2の命令が実行可能であるが、長いコースを走行するような課題、あるいは複雑な動きを多用する場合にはステップ数が少ないと訴える学習者がいた。

これらの問題点を解決するにあたり、本基板が教材用であることを考慮し、既存の基板から大きな変更が出ないようにするようにした。既存の基板自体や、赤外線通信インタフェースなどは改良の対象からはずし、これまでのプログラムや装置をそのまま使用できるような改良を行った。以下に具体的な解決方法について述べる。

3.1 問題点1の解決方法

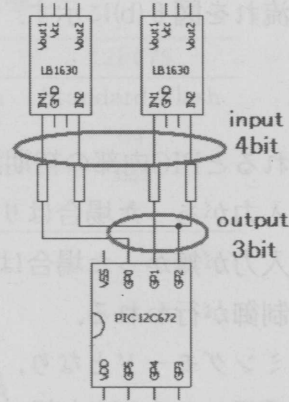
基板からモータを制御するために、モータドライバIC (LB1630)¹¹⁾を使用した。モータドライバの回路の特性を図4(a)に示す。図4(b)に、モータドライバとPIC12C672との接合状態を示す回路図を示す。このモータドライバ、直流DCモータを二つの制御信号により「正転」「逆転」「停止」を制御する。したがって、2モータを独立に制御するためには4本の信号線が必要である。しかし本基板では、1本の制御信号線を共通化して3本の信号線で制御しているため、独立した制御ができない仕様であった。そのため、旋回命令は図5(a)のように片方のモータを回転させることで実現していた。ところが実際に学習者が望む回転は図5(b)のように左「正転」右「逆転」、または左「逆転」右「正転」を同時に制御するものであった。そこで、左右独立にモータを回転させるのではなく、短時間(0.1ms)で「右モータ回転・左モータ停止」と「右モータ停止・左モータ回転」の二つ状態を繰り返すプログラムをファームウェアに付け加えた。



Truth Table

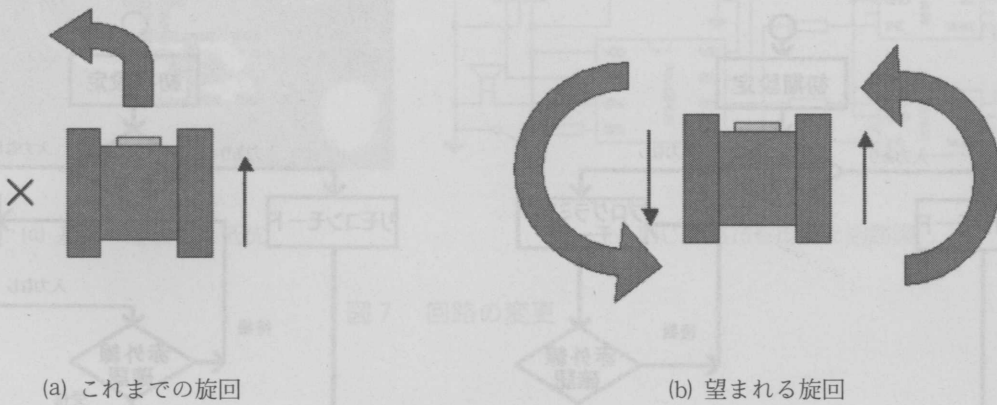
IN 1	IN 2	OUT 1	OUT 2	Mode
H	L	H	L	Forward
L	H	L	H	Reverse
H	H	L	L	Brake
L	L	OFF	OFF	Standby

(a) モータドライバの仕様



(b) モータドライバとの接続回路

図4 旋回的方式



(a) これまでの旋回

(b) 望まれる旋回

図5 旋回的方式

3.2 問題点2と問題点3の解決方法

本基板で使用しているPIC12C672内に記録されているファームウェアは以下の動作を行う。

1. 電源を入れるとPIC内部の初期設定が行われる
2. スイッチ入力があった場合はリモコンモードとなり、リモコンでの制御が行われる
3. スイッチ入力が無かった場合はプログラミングモードとなり、パソコンからプログラムを転送しての制御が行われる
4. プログラミングモードとなりスイッチ入力が無い場合は、赤外線受信のため待機する
5. 赤外線を受信したらPIC内部のRAMに命令を記憶し、転送されたプログラムを実行する

ファームウェアによる動作の流れを図6(a)に示す。本基板では転送したプログラムを保存していないため、基板の電源を落とすと全ての制御プログラム(中間コード)が消去されてしまう。そこでEEPROMに制御プログラム(中間コード)を保存する方式に変更した。

なぜならば、EEPROMならば回路の電源を落としても記録された内容が記憶されるからである。ところが、PIC12C672には、EEPROMが内蔵されていないため、同型のPIC12F675に変更した。表2にPIC12C672とPIC12F675の主な仕様を示す。また、改良したファームウェア

による動作の流れを図6(b)に示す。改良したファームウェアによりPIC12F675では以下の動作を行う。

1. 電源を入れるとPIC内部の初期設定が行われる。
2. スイッチ入力があった場合はリモコンモードとなり、リモコンでの制御が行われる。
3. スイッチ入力が無かった場合はプログラミングモードとなり、パソコンからプログラムを転送しての制御が行われる。
4. プログラミングモードとなり、スイッチ入力がない場合は赤外線受信のため待機する。
5. 赤外線を受信したらPIC内部のEEPROMに命令を記憶し、新規のプログラムを実行する。
6. プログラミングモードとなりスイッチ入力がある場合は、PIC内部のEEPROMの内容を呼び出し、既存のプログラムを実行する。

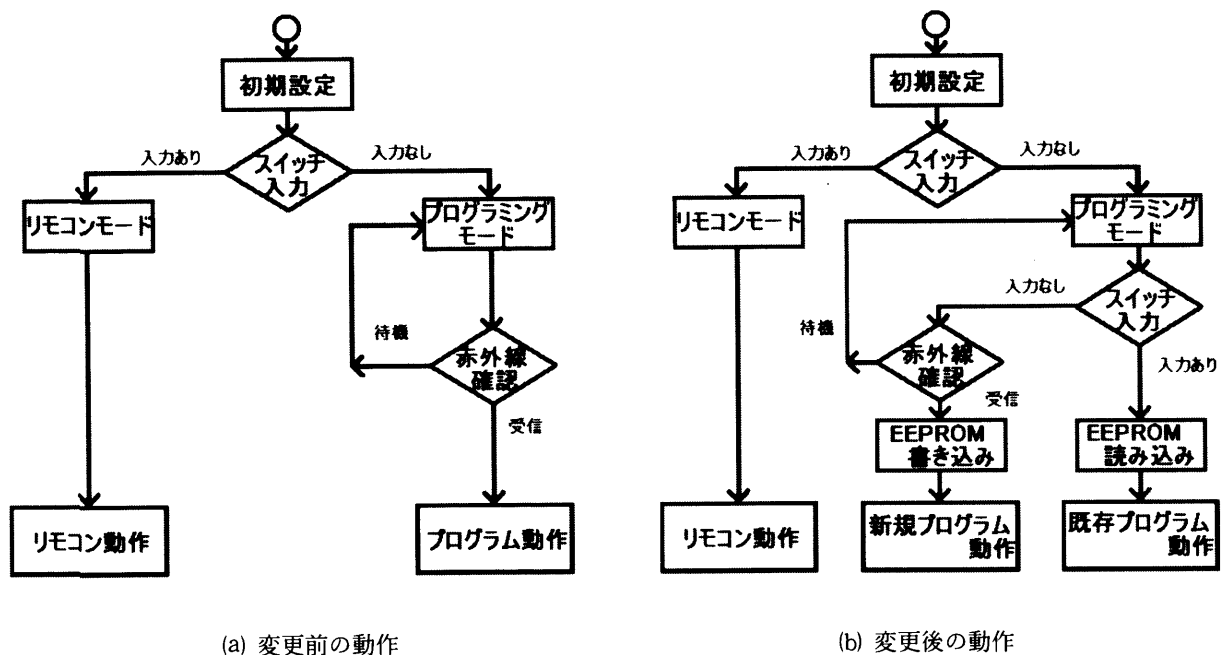


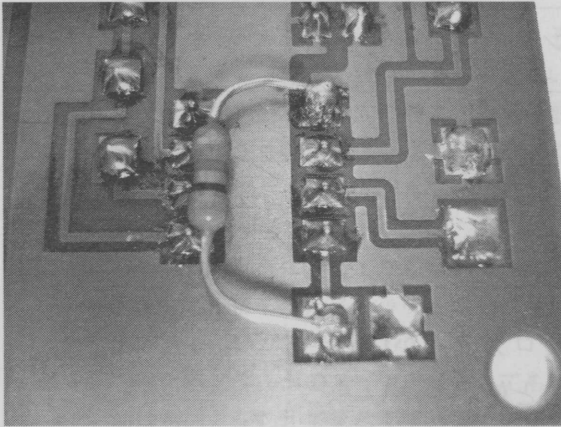
図6 動作処理の変更

さらに、PIC12C672からPIC12F675へ変更することで、制御プログラム（中間コード）をすべて内蔵のEEPROMへ保存することが可能になった。そのため、制御プログラムの保存量を39ステップから47ステップまで上げることができた。本基板では、制御プログラムを1 byteの命令コードと引数（合計2 byte）で書いているため、128byteのEEPROMには最大64ステップまで記録可能である。しかし、制御プログラムのサブルーチン用の領域を28byte（7ステップ×2）と、制御プログラムの開始と終了を宣言する制御命令（「始めロボット」と「終わりロボット」）2 byteが必要であるため、47ステップとした。

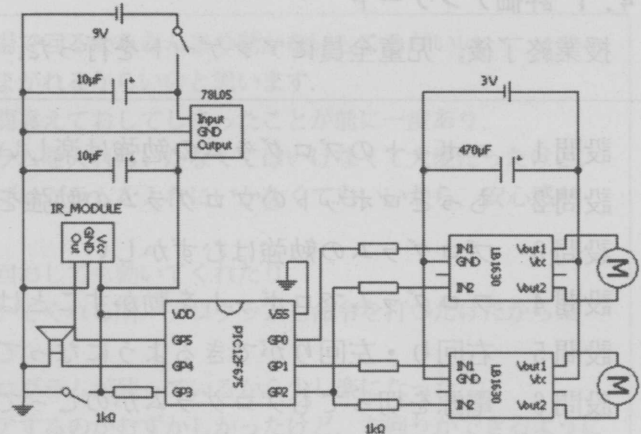
このように、マイクロコントローラとファームウェアの変更により問題点2（不揮発性メモリに制御プログラムを保存できない。）は解決することができた。問題点3（プログラムを保存できる記憶容量が少ない。）に関しては、十分とは言えないが、8ステップ分のプログラムを増やすことができた。問題点3に関しては、EEPROM容量を多く内蔵する同型のPIC12F

表2 12C672 と 12F675の仕様

Parameter Name	12C672	12F675
Program Memory Type	One Time Program	Standard Flash
RAM(bytes)	128	64
Data EEPROM(bytes)	-	128
I/O	6	6



(a) 基板へ追加した抵抗



(b) PIC12F675を用いた回路図

図7 回路の変更

683 (EEPROM 256byte 内蔵) を利用することで解決することが可能であるが、コストを抑える理由から今回は使用しなかった。

3.3 部品の追加

PIC12C672ではI/O端子GP 3がプルアップされているが、PIC12F675のI/O端子GP 3にはプルアップがかかっておらず、改良した基板での入力スイッチの動作に不具合が生じることが明らかになった。そこでPIC12F675のGP 3にプルアップ抵抗 (1 k Ω) を加えた。図7(a)は基板に抵抗を半田付けした様子であり、図7(b)にプルアップ抵抗を加えた基板の回路図を示す。

4 改良した教材の評価

改良した教材の評価を、改良前の教材で学習した経験がある小学生を対象に行った。評価を実施した小学校では2006年6月から7月にかけてプログラミング言語「ドリトル」とlogob.com基板を使用したロボット教材を用いた授業実践を行っていたため、今回の評価では以前のロボットとの比較をする事を目的に行った。評価実験を行った時期と学校名を下記に示す。

日時：2006年12月

場所：静岡県藤枝市立大洲小学校

人数：小学校6年生 32名

時間：2時限（45分×2）

授業は総合的な学習の時間において行った。授業は、室伏が担当した。

4.1 評価アンケート

授業終了後、児童全員にアンケートを行った。アンケートの内容を以下に示す。

-
- 設問1 ロボットのプログラムの勉強は楽しい
 設問2 もっとロボットのプログラムの勉強をやりたい
 設問3 プログラムの勉強はむずかしい
 設問4 プログラムでロボットを動かすことはすごいことだ
 設問5 右回り・左回りができるようになって、プログラムが作りやすくなった
 設問6 電源を切ってもプログラムがのこっていると使いやすい
 設問7 今回のロボットは、これまでのロボットよりも使いやすい
 設問8 今回のロボットで、使いやすくなったことを下にご書いてください（自由記述）
-

設問1から7までは下記の五段階の尺度を用意し、回答を求めた。

5：強く思う 4：思う 3：どちらともいえない 2：思わない 1：まったく思わない

回答の5, 4を肯定的, 2, 1を否定的な回答と捉えた。

4.2 アンケート結果

設問1から7の回答結果を表3に、肯定的な回答割合を図8に示す。

表3 アンケートの結果

No.	5	4	3	2	1
設問1	19	11	2	0	0
設問2	23	8	1	0	0
設問3	9	11	6	6	0
設問4	17	9	6	0	0
設問5	16	12	3	1	0
設問6	22	9	1	0	0
設問7	24	7	1	0	0

N=32

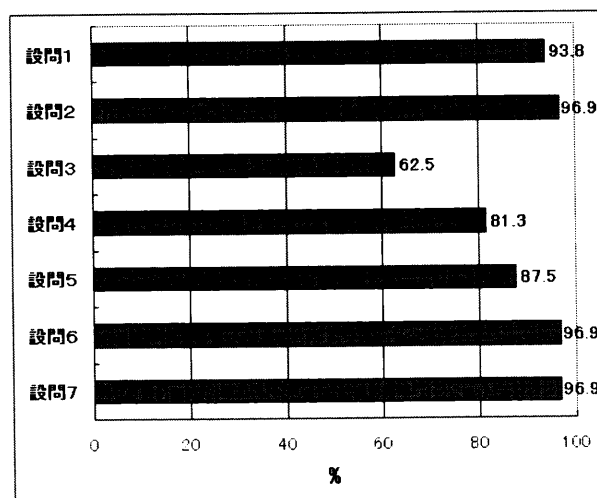


図8 肯定的な回答割合

図3より設問1から4にかけてはロボットを制御するプログラミング学習についての設問であるため、本研究の対象外であるが、肯定的な回答が多いことから、本教材を利用した制御学習が「難しいが楽しい」学習であることがわかる。また設問5から7にかけての回答が肯定的な回答が多いことから、ほとんどの児童が今回の改良により、プログラムが作りやすくなり、ロボットが扱いやすくなったと回答していることがわかった。

設問8における、特徴的な意見を図9に示す。図9より改良による制御教材が扱いやすくなったことが明らかとなった。

- (1) 右回り、左回りを使えるとすごくいい。その場で回るからあんまり動かないところがいい。あと、その場で回ると数字によってきれいにまがれるからいいと思います。
- (2) ロボットをコースの上ののせて走らせる前に間違えておしてしまったことが前に一度あり、勝手に動いたりして、さらにもう一度プログラムを入れにいかなくてはいけなくて大変だったけど、この新しいロボットは間違えても、もう一度プログラムを入れにいかなくてもいいから、安心感があって良かったです。
- (3) 命令をロボットに送ってからはスイッチを何回おしても動いてくれたり、電源を切ってもまたスイッチをおせばまた動いてくれる所。プログラムも命令を打つだけだから楽にできて良い。
- (4) 間違えてスタートのスイッチをおしても、プログラムが残っているから少し楽になった。前は右前とかしかできなくて、コースをクリアするのがむずかしかったけど、右回りができるようになったので、クリアするのにそんなに時間がかからなかった。

図9 設問8の回答

5 考察

ロボット制御用命令がEEPROMの利用により、再実行可能になったことで教材として扱いやすくなったと考えられる。また、改良前と改良後で比較して図9の児童の回答から、問題点の改良による制御教材としての使いやすさが増したことがわかった。ところが、問題点3（プログラムを保存できる記憶容量が少ない。）の改良に関する内容が、アンケートからはうかがえなかった。その原因として評価実験の時間が説明を含めて90分（45分×2）では児童にとって短かかったと考えられる。

6 まとめ

基板の改良により、教材用としてコスト^{*2}を抑えながら機能向上を果たすことができた。

小中学校の制御学習では、既製品の教材を用いて学習することが多いが^{14) 15)}、それらの不具合を感じたときに、使用している教材を変更するか、不具合を承知で使い続けるかのどちらかである。教材の変更には、コストがかかるうえ、学習者の細かいニーズに対応することは難しい。筆者らの取り組みは、既存の基板を交換することなしに学習者のニーズに合わせた改良をすることができた。これらの改良が可能なのは、マイクロコントローラを利用した回路設計に因るところが大きい。しかし、マイクロコントローラの進歩も日進月歩であり、制御機能の

*2 PIC12F675 120円 抵抗（1kΩ）10円 の計 130円

向上は目覚ましいものがある。その機能に合わせて基板の変更を行うのでは、そのたびに教材が変更され授業者は混乱する。筆者らの改良方法は、マイクロコントローラの進歩に合わせた形で、基板を変えずに行える。今後は、マイクロコントローラを利用し、改良に対応できる教材回路設計が更に望まれる。

参考文献

- [1]文部科学省. (平成10年) 中学校学習指導要領. 財務省印刷局, (1999)
- [2]文部科学省. 中学校学習指導要領 (平成10年12月) 解説-技術・家庭科編-. 東京書籍, (1999)
- [3]米田 完, 坪内孝司, 大隅 久: はじめてのロボット創造設計, 講談社 (2001)
- [4]末松良一, 雨宮好文: 制御用マイコン入門 (2005)
- [5]Shuji Kurebayashi, Toshiyuki Kamada, Susumu Kanemune: Learning Computer Program with Autonomous Robots, LNCS, Vol.4226, pp.138-149 (2006)
- [6]佐藤和浩, 紅林秀治, 兼宗 進: 小学校におけるプログラミング活用の現状と課題., 情報処理学会 コンピュータと教育研究会, CE (78), pp.57-63 (2005)
- [7]紅林秀治, 兼宗 進: プログラミング学習についての一考察: ロボット制御のプログラミング学習とソフトウェア作りのプログラミング学習を比較し, 情報教育シンポジウム論文集 (SSS2004), pp.21-28 (2004)
- [8]森慎之助, 山本 透, 森岡 弘, 白浜弘幸: 中学校技術・家庭科 (技術分野) におけるロボット技術を用いた動力伝達および機構学習, 日本産業技術教育学会誌, 第48巻 3号, pp.193-200 (2006)
- [9]伊藤陽介, 森 誉範, 菊地 章, 大泉 計: 「プログラムと計測・制御」のためのロボット学習材の開発と実践, 日本産業技術教育学会誌, 第49巻 3号, pp.213-221 (2006)
- [10]Microchip Technology Inc. <http://www.microchip.com/>
- [11]Low-Saturation Bidirectional Motor Drive for Low-Voltage Applications, SANYO Electric Co., Ordering number: EN3515A, Ltd. Semiconductor Business Headquarters, pp.1-4 (1996)
- [12]兼宗 進, 御手洗理英, 中谷多哉子, 福井真吾, 久野 靖: 学校教育用オブジェクト指向言語「ドリトル」の設計と実装, 情報処理学会論文誌, Vol.42, No.SIG11 (PRO12), pp.78-90 (2001)
- [13]兼宗 進. プログラミング言語「ドリトル」. <http://dolittle.eplang.jp/>
- [14]嶋田彰子, 山菅和良, 針谷安男, 鈴木道義: 自律型ロボット教材を活用したプログラムと計測・制御学習に関する授業方法の開発と評価, 日本産業技術教育学会誌, 第49巻 4号, pp.297-305 (2007)
- [15]森真之助: ロボット教材を用いた制御・プログラミングの授業実践と作業分析, 日本産業技術教育学会誌, 第47巻 3号, pp.201-207 (2005)