

博士学位論文

題 目

計測・制御教育のための教材開発とその教材を活用した授業設計及び実践

担当指導教員名 魚井 宏高 ㊞

申請年月日 2013年 10月 2日

申請者専攻名 コンピュータサイエンス専攻

学 生 番 号 DT07A802

氏 名 森石 峰一 ㊞

大阪電気通信大学大学院

目 次

第1章 本研究の目的と概要	1
第1章の参考文献	3
第2章 中学校の技術・家庭科における計測・制御教育	5
2.1 学習指導要領に記載されている計測・制御教育の変遷	6
2.2 計測・制御教育の必修化による問題点	8
2.3 「プログラムによる計測・制御」の指導計画の提案	9
2.4 「プログラムによる計測・制御」で利用する実習用教材の開発	13
2.4.1 インタフェースの開発	16
2.4.2 計測・制御システムを操作するソフトウェアの開発	32
2.5 開発した実習用教材の評価	43
第2章の参考文献	47
第3章 大学における計測・制御教育の授業設計及び実践	51
3.1 「すだち方式」による計測・制御教育の授業設計	52
3.2 POI (PBL Oriented Instruction) の実践	54
3.3 POI を導入した効果	63
3.4 PBL (Project Based Learning) の実践	67
3.5 PBL 活動の成果	69
第3章の参考文献	72
第4章 結論	74
4.1 今後の課題	76
謝辞	77
付録 システム操作ソフトウェアのプログラム	
(1) 生徒がプログラムを作成する環境	78
(2) システム操作ソフトウェア本体のプログラム	80

第1章 本研究の目的と概要

計測・制御教育は、我が国のもの作り教育の重要な教育の1つであり、中学校から学習を始め大学で完結させることが望ましい。

現在、中学校では、2008年(平成20年)3月に告示された学習指導要領で、技術・家庭科の技術分野に「プログラムによる計測・制御」を必修項目として導入されたので、全生徒が計測・制御学習をしている^[1-1]。しかし、高等学校では、工業高校、農業高校、水産高校で計測・制御教育に関係する指導を行なっているものの、全生徒数の10.6%しか学習をしていない^[1-2, 1-3]。また、大学では、工学系の大学において社会の要望により基礎専門科目として導入されていることが多い。

このような状況であるが、我が国の計測・制御教育を発展させるためには、中学校で2012年度(平成24年度)から完全実施になったばかりの「プログラムによる計測・制御」を指導できるように、技術担当教員に対する支援を行なう必要がある。また、工学系の大学では、体系的に計測・制御教育を実施しているので、学生が学びやすい環境を整備することが望ましい。

中学校における教育は、学習指導要領で学習の方向と大まかな内容が定まり、教科書は文部科学省の検定合格済みのものを使用することが前提となっている。また、教授者は教育職員免許法で定められた教員免許を持たなければならない。つまり、統制された環境で指導が行われることになるので、主に木工、金工等を指導してきた技術・家庭科で技術分野を担当する教員(以下、技術担当教員と記述する)であっても、必修項目になった「プログラムによる計測・制御」の指導を、学習指導要領に従って行なわなければならないことになる。

このような状況下で、教職科目である技術科教育法(中学校)と情報科教育法(高等学校)を、指導する立場から、技術担当教員の指導に対する不安や戸惑いを払拭するために、指導計画の提案と、指導目標を達成するために必要な実習用教材を開発しなければならないと考えた^[1-4]。

また筆者は、教職科目を指導する教員であるだけでなく、工学部の一員として、大学で実施されている計測・制御教育にも関心を持っている。

このような立場の筆者であるが、O大学でHidden Curriculum企画委員会

が企画し、「コンピュータで制御する対象物を、学生自身が考案し製作すること」を目標にした「光・動き出す・未来」プロジェクトを担当することで、大学生を対象とした計測・制御教育を行なう機会を得ることができた。

大学での教育は、中学校や高等学校の教育環境とは異なり、学習指導要領はなく、文部科学省検定教科書もない。また、教授者が教員免許を持つ必要もない。したがって、大学の授業設計は、教育行政面での制約条件がほとんどなく、教授者の自由裁量で行える部分が多い。

この自由度の高い授業設計環境では、教育の目的や目標を明確にし、実績がある指導方法を活用することが効果的である。

「光・動き出す・未来」プロジェクトの目標を達成させるためには、参加する学生が自由な発想で製作ができるような環境作りが重要になるので、学生の主体的な学習を促し創造性を高めるために有効な指導方法である **Project Based Learning** (以下、**PBL**と略記する)を活用することが望ましい。

現在では、この**PBL**を創成科目という用語を用いて、多数の大学が学科単位で実施している。この創成科目の考え方と実施案は、都倉(2000)^[1-5]が報告している。

PBLを行なう場合に、学生に必要な基礎能力は、①**PBL**の到達目標までの計画を立案できること、②**PBL**の開始から学生自身が設定した到達目標までの構想を、ある程度段階的に考えられる知識、意識(価値観や動機付け)、スキル(技術)等を身に付けていること、③**PBL**の到達目標を、成果物の製作や成果報告の発表等の具体的な方法で、他者に伝えられることである。

しかし、この①～③までの基礎能力について、教員は自らの経験からその必要性を認識しているが、理解の乏しい学生はその必要性を十分認識していないことがある。このような状況であるので、**PBL**の効果を着実に高めるためには、学生に求めるこれら3つの基礎能力を高める授業を、**PBL**を実施する前に行なうべきであると考え、この授業に、**PBL Oriented Instruction** (以下、**POI**と略記する)と名づけた^[1-6]。

この**POI**は教員主導で進め、学生の持つ基礎能力が**PBL**を開始できるレベルに達した時点で、教員主体の指導から学生主体の**PBL**へとつなげる指導方法であることから、動物が巣立ちをする行動にたとえて、「すだち方式」と

名づけた。

そこで、「コンピュータで制御する対象物を，学生自身が考案し製作すること」を目標にした「光・動き出す・未来」プロジェクトを，POIからPBLへとつなげる「すだち方式」の指導方法で進めることにした^[1-7]。

本研究では，中学校の技術担当教員が，「プログラムによる計測・制御」の指導目標を達成するために必要な指導計画を提案し，同時に実習用教材としてのインタフェースと計測・制御システムを動作させるソフトウェアを開発した経緯と，この実習用教材の有用性について述べる。

また，工学系の大学で計測・制御教育を実施する際に，有効な指導方法であるPBLの効果を高めるための「すだち方式」についての理論づけを行なう。さらに，「すだち方式」の指導方法で授業設計を行ない，「プログラムによる計測・制御」の実習用に開発したインタフェースを利用した計測・制御教育の実践について述べる。

第1章の参考文献

- [1-1]文部科学省：“中学校学習指導要領(平成20年3月)”，東山書房(2008)
- [1-2]文部科学省：“高等学校学習指導要領(平成21年3月)”，東山書房(2009)
- [1-3]総務省 統計局：“高等学校(全日制・定時制)の学科別学校数及び生徒数(本科)”，<http://www.e-stat.go.jp/SG1/estat/List.do?bid=000001049724&cycode=0>
- [1-4]森石 峰一，横山 宏，魚井 宏高，“中学校の技術・家庭科の「プログラムによる計測・制御」で利用する学習教材の開発と試用”，教育システム情報学会，Vol.29, No.4, 2012年10月，pp.190-200, (2012)。
- [1-5]都倉信樹：“創成科目の考え方と実施案”，工学教育，工学・工業教育研究講演会講演論文集 平成12年度，pp.275-278, (2000)
- [1-6]森石 峰一，宇治 典貞，横山 宏，魚井 宏高，“PBLの成功を目的とした「すだち方式」による準備授業の設計と実践”，日本教育情報学会，第28巻，第3号，2013年3月，pp.15-22, (2013)

[1-7] 森石 峰一，都倉 信樹，横山 宏，魚井 宏高，“基礎指導を組み込んだ PBL 方式でのもの作り教育の試み”，日本工学教育協会，「工学教育」11月号，2013年12月，pp.22-28，（2013）

第2章 中学校の技術・家庭科における計測・制御教育

1986年(昭和61年)4月に出された臨時教育審議会の第2次答申において、「社会の情報化の進展に伴い、情報活用能力(情報リテラシー:情報及び情報手段を主体的に選択し活用していくための個人の基礎的な資質)をどの程度身につけるかによって、個人が情報手段を主体的に活用できるか、情報化の弊害の中に埋没してしまうかがかなり左右される」と述べられている^[2-1]。

また、「これまでの『読み・書き・算盤』のもつ教育としての基礎的・基本的な部分をおろそかにすることなく、新たに『読み・書き・情報活用能力』を基礎・基本として重視し、(中略)…情報活用能力の育成に本格的に取り組んでいくことが重要である」と指摘されている。

この報告書で重要なことは、行政文書に初めて「情報活用能力」という用語が使われ、その情報活用能力と情報リテラシーが同義語であると示され、さらに、情報活用能力を「読み・書き・算盤」と並ぶ全ての人に共通の基礎的資質であると示されたことである。

しかし、この答申では、情報活用能力の定義は具体的に記されていない。

1989年(平成元年)3月に告示された学習指導要領では、「情報活用能力の育成は、すべての教科で図られるべきものである」ことと、「基礎的な情報学を学習する新領域「情報基礎」を中学校技術・家庭科に新設する」ことが明示され、技術・家庭科の学習内容は、木材加工、電気、金属加工、機械、栽培、情報基礎、家庭生活、食物、被服、住居、保育の11領域になった^[2-1, 2-2]。

1997年(平成9年)10月には、文部省内に設置された「情報化の進展に対応した初等中等教育における情報教育の推進等に関する調査研究協力者会議」が、「体系的な情報教育の実施に向けて」という報告書が公表された。なお、初等教育は小学校を、中等教育は中学校と高等学校を示している。

この報告書の中で、「今後の初等中等教育段階における情報教育で育成すべき「情報活用能力」を以下のように焦点化し、系統的、体系的な情報教育の目標として位置づけることを提案する」と述べ、次のように情報教育の目標が示されている^[2-3]。

- ① 課題や目的に応じて情報手段を適切に活用することを含めて、必要な情報を主体的に収集・判断・表現・処理・創造し、受け手の状況などを踏まえて発信・伝達できる能力。（「情報活用の実践力」と略称する。）
- ② 情報活用の基礎となる情報手段の特性の理解と、情報を適切に扱ったり、自らの情報活用を評価・改善するための基礎的な理論や方法の理解。（「情報の科学的な理解」と略称する。）
- ③ 社会生活の中で情報や情報技術が果たしている役割や及ぼしている影響を理解し、情報モラルの必要性や情報に対する責任について考え、望ましい情報社会の創造に参画しようとする態度。（「情報社会に参画する態度」と略称する。）

この報告書の意義は、情報教育を情報活用能力の育成を行うための教育であると明確に位置づけたことと、情報活用能力を情報教育の目標として明確に位置づけたことである。

2.1 学習指導要領に記載されている計測・制御教育の変遷

中学校において、情報活用能力の育成を教科内容として取り扱うのが、技術・家庭科の技術分野である。

1998年（平成10年）12月に告示された学習指導要領では、技術分野の半分を「情報とコンピュータ」が占め、その学習内容は、①生活や産業の中で情報手段の果たしている役割、②コンピュータの基本的な構成と機能及び操作、③コンピュータの利用、④情報通信ネットワーク、⑤コンピュータを利用したマルチメディアの活用、⑥プログラムと計測・制御の6項目で構成された^[2-4]。

注目すべきことは、「プログラムと計測・制御」は選択項目であるものの、中学校に初めて計測・制御教育が導入されたことである。

「プログラムと計測・制御」の指導目標は、①コンピュータを働かせるプログラムの必要性とその機能を理解させる、②課題を解決するための簡単なプログラムの作成ができるようにする、③身近な生活の中に、コンピュータを用いた計測・制御が利用されていることを理解させる、④目的に応じた簡単

な計測・制御ができるように指導することとされている^[2-5]。

この指導目標を達成するための指導内容は、㉞プログラムの機能を知り、簡単なプログラムの作成ができること、㉟コンピュータを用いて、簡単な計測・制御ができることとされている。

㉞の内容を細分化し、㉠プログラムの機能については、簡単なサンプルプログラムを利用して、順次、反復、分岐などの基本的な情報処理の手順を理解させること、㉡基本的な操作ができるように、プログラムの呼び出し方、プログラムのリストの見方、プログラムの実行の仕方などを理解させること、㉢プログラミング言語や、アプリケーションソフトウェアのマクロ機能等を用いて、簡単なプログラムの作成ができるように指導することとされている。

また、㉟の内容を細分化し、㉠身の回りにある機器の中には、環境の変化を計測し、その計測結果をコンピュータが処理をして、機器の動作を制御しているものが多くあることを理解させること、㉡計測・制御システムは、機械や環境の状態を計測するセンサ、計測された情報を処理・判断するコンピュータ、コンピュータからの命令を受けて動作をするアクチュエータなどの要素で構成されていることを理解させること、㉢計測・制御システムを構成している各要素間で、情報の伝達が行なえるようにするために、インタフェースが必要であることを理解させることとされている。

1998年(平成10年)12月に告示された学習指導要領で、選択項目であった「プログラムと計測・制御」が、2008年(平成20年)3月に告示された現行の学習指導要領では、必修項目の「プログラムによる計測・制御」になった^[2-6]。

必修項目になったことで、全ての中学校で「プログラムによる計測・制御」の指導を行なわなければならないようになる。この必修化による問題点は、2.2節で述べるように、主に木工、金工を指導してきた技術担当教員であっても、計測・制御教育の指導を、学習指導要領に従って行なわなければならないので、指導の経験が乏しいこと等の理由により、指導に対する不安や戸惑いを感じている技術担当教員が少なくないことである。

「プログラムによる計測・制御」の指導目標は、㉠計測・制御のためのプログラムの作成を通して、コンピュータを用いた計測・制御の基本的な仕組みを

理解させる，②簡単なプログラムの作成ができるようにする，③情報処理の手順を工夫する能力を育成することとされている^[2-7]。

この指導目標を達成するための指導内容は，⑦コンピュータを利用した計測・制御の基本的な仕組みを知ること，④情報処理の手順を考え，簡単なプログラムが作成できることとされている。

⑦の内容を細分化し，⑧計測・制御システムが，センサ，コンピュータ，アクチュエータなどの要素で構成されていること，⑨計測・制御システムの中では一連の情報がプログラムによって処理されていること，⑩計測・制御システムを構成している各要素間で，情報の伝達が行なえるようにするために，インタフェースが必要であることを理解させることとされている。

また，④の内容を細分化し，⑧簡単なプログラムを作成できるように指導すること，⑨情報処理の手順を，目的や条件に応じて工夫する能力を育成すること，⑩情報処理の手順を考える際に，よりよいアイデアが生み出せるように，自分の考えを整理する方法を指導することとされている。

2.2 計測・制御教育の必修化による問題点

計測・制御教育の必修化による問題点の 1 つに，技術担当教員の指導力不足がある。この問題点については，山本ら(2011)^[2-8]は，1998年(平成10年)12月に告示された学習指導要領で，「プログラムと計測・制御」が選択項目であったことから，指導経験が不足した技術担当教員がいると述べ，必修項目になる「プログラムによる計測・制御」を指導するためには，適切な研修を行なう必要があると指摘している。また，村松ら(2006)^[2-9]は，計測・制御教育を指導する際の阻害要因を，①授業時数の不足，②予算の不足，③研修や経験の不足の3点であると指摘している。

さらに，政宗(2011)^[2-10]は，2つの調査結果を根拠に，技術担当教員が，「プログラムによる計測・制御」の指導に対して不安や戸惑いを感じていることや，教材開発が十分に行なわれていないのが現状であることを指摘している。この根拠になった調査結果は，国立教育政策研究所が2005年(平成17年)に実施された「音楽等質問紙調査」^[2-11]で，「プログラムの働きと簡単なプログラムの作成することについて」の選択を「現時点までに指導している」

にした技術担当教員が，第 1 学年 3.3%，第 2 学年 9.5%，第 3 学年 10.8%であったことや，「コンピュータを用いた温度などの計測や模型などの簡単な制御をすることについて」の選択を「現時点までに指導している」にした技術担当教員が，第 1 学年で 1.6%，第 2 学年 0%，第 3 学年 1.6%であったことであり，政宗が広島県内 102 校の技術担当教員を対象に 2010 年（平成 22 年）に実施された実態調査で，計測・制御学習の指導に不安を感じていると回答した教員が 66.0%おり，その理由の上位 2 つは，37 校が「教材開発ができていないこと」で 36 校が「今までに指導の経験がないこと」であった。

このような現状において，技術担当教員の指導に対する不安や戸惑いを払拭するために，仮に指導の経験が乏しい技術担当教員であっても，指導計画の提案と，指導目標を達成するために必要な実習用の教材を開発しなければならないと考えた。

2.3 「プログラムによる計測・制御」の指導計画の提案

技術・家庭科の標準の授業時数は，「学校教育法施行規則」により，第 1 学年が 70 単位時間とされ，第 2 学年が 70 単位時間，第 3 学年が 35 単位時間とされている^[2-6]。したがって，3 年間の全授業時数は 175 単位時間になる。なお，授業時数の 1 単位時間は，50 分とされている。

技術・家庭科を構成する各内容の授業時数は，表 2-1 に示す技術分野と家庭分野の各々 4 つの内容を，均等に配当すると仮定して計算をしてみると， $175(\text{全授業時数}) \div 8(\text{技術・家庭科の内容数}) = 21.875$ となり，その結果 21 単位時間程度は必要と想定できる。

「プログラムによる計測・制御」は，内容「D 情報に関する技術」を構成する

表 2-1 技術・家庭科の内容

技術分野	家庭分野
A 材料と加工に関する技術	A 家族・家庭と子どもの成長
B エネルギー変換に関する技術	B 食生活と自立
C 生物育成に関する技術	C 衣生活・住生活と自立
D 情報に関する技術	D 身近な消費生活と環境

1 項目である。この内容「D 情報に関する技術」は、「情報通信ネットワークと情報モラル」、「デジタル作品の設計・制作」、「プログラムによる計測・制御」で構成されているので、均等に配当するならば、「プログラムによる計測・制御」を指導する授業時数は、7 単位時間程度と想定できる。

この限られた授業時数で、「プログラムによる計測・制御」の指導目標である①計測・制御のためのプログラムの作成を通して、コンピュータを用いた計測・制御の基本的な仕組みを指導する。②簡単なプログラムの作成ができるように指導する。③情報処理の手順を工夫する能力を育成する必要がある [2-7]。

「プログラムによる計測・制御」では、基礎的・基本的な知識及び技術を、平均的な生徒が持つ学習能力で確実に定着させる必要がある。この指導は実習を取り入れ、生徒が容易に理解できる内容から徐々に高度な内容に進行する方法が望ましい。これを基にした計測・制御実習の導入部分では、入力信号が出力結果に影響されずに、入力信号から出力信号への一方向の流れで制御を行なう開ループ制御について指導し、学習が進行した発展部分では、入力信号から出力信号へと流れた信号を、元の入力信号に戻して制御を行なう閉ループ制御について指導することが想定できる。したがって、計測・制御実習を行なうには、開ループ制御と閉ループ制御の働きが理解しやすい教材としての機能を持ったインタフェースと、コンピュータ上で動作し計測・制御システムを操作するためのソフトウェア(以下、システム操作ソフトウェアと記述する)とが必要になる。また、これらの指導と並行して、計測・制御を行なっている電化製品が、日常生活を豊かにしていることに、気付かせる指導も必要である。

前述の指導目標に準拠しながら、開ループ制御から閉ループ制御へ学習が進行する実習を取り入れ、7 単位時間で実施が可能な指導計画を想定してみた(表 2-2, 表 2-3) [2-12]。

この指導計画では、2~4 回目の 3 単位時間で、開ループ制御と閉ループ制御の基礎的・基本的な知識と技術を習得させ、5~7 回目の 3 単位時間で、生徒自らが情報処理の手順を工夫できるようになっている。

表 2-2 想定した 7 単位時間の指導計画(基礎編)の一例

■ 1 回目 基礎的, 基本的な知識及び技術の定着(1)	
指導内容	指導方法
<p>①計測・制御を行なっている電化製品が, 日常生活を豊かにしていることを確認させる.</p> <p>②計測・制御システムの基本的な仕組みと, そのシステムを構成する各要素の働きを理解させる.</p>	<p>①計測・制御がどのようなところで利用されているかを生徒に考えさせる.</p> <p>②-1 システムは, アクチュエータ, センサ, コンピュータ, インタフェースで構成されており, それらの機能が相互に関係を持っていることを説明する.</p> <p>②-2 アクチュエータとセンサの働きを説明する.</p> <p>②-3 アクチュエータ, センサ, コンピュータの間で信号の伝達を行なう際に, インタフェースが必要であることを説明する.</p>
■ 2 回目 基礎的, 基本的な知識及び技術の定着(2)	
指導内容	指導方法
<p>①システム操作ソフトウェアの操作方法に慣れさせる.</p>	<p>①アクチュエータを制御するプログラムを予め用意し, 課題に沿って生徒にプログラムを変更させる.</p>
■ 3 回目 基礎的, 基本的な知識及び技術の定着(3)	
指導内容	指導方法
<p>①開ループ制御の簡単なプログラムを作成させる.</p>	<p>①アクチュエータを制御する課題に沿って生徒にプログラムを作成させる.</p>
■ 4 回目 基礎的, 基本的な知識及び技術の定着(4)	
指導内容	指導方法
<p>①センサを利用した閉ループ制御のプログラムを作成させる.</p>	<p>①-1 予め用意したセンサとアクチュエータで計測・制御を行なうプログラムを, 課題に沿って生徒に変更させる.</p> <p>①-2 センサとアクチュエータを利用する課題に沿って生徒にプログラムを作成させる.</p>

表 2-3 想定した 7 単位時間の指導計画(活用編)の一例

■ 5 回目 活用する能力と態度の習得 (1)	
指導内容	指導方法
<p>①「B エネルギー変換に関する技術」の復習を行なう.</p> <p>②目的や条件に応じた情報処理の手順を考えさせる.</p>	<p>①「(1) エネルギー変換機器の仕組みと保守点検」で指導した, 歯車やカム機構, リンク機構等の力や運動を伝達する仕組みについて説明する.</p> <p>②プログラムの作成を目的とし, 「(2) エネルギー変換に関する技術を利用した製作品の設計・製作」(以下, 「製作品の設計・製作」と記述する)で製作させた製作品を, どのような動作をさせたいかのアルゴリズムを生徒に考えさせ, その手順を箇条書きにまとめさせる.</p>
■ 6 回目 活用する能力と態度の習得 (2)	
指導内容	指導方法
<p>①情報処理の手順に沿ったプログラムを作成させる.</p>	<p>①「製作品の設計・製作」で製作させた製作品をインタフェースに接続させ, 箇条書きの手順通りに動作を行なうプログラムを生徒に作成させる.</p>
■ 7 回目 活用する能力と態度の習得 (3)	
指導内容	指導方法
<p>①情報処理の手順に沿ったプログラムを作成させる.</p> <p>②アクチュエータ, センサ, インタフェースの役割と, プログラムの重要性について理解させる.</p>	<p>①「製作品の設計・製作」で製作させた製作品をインタフェースに接続させ, 箇条書きの手順通りに動作を行なうプログラムを生徒に作成させる.</p> <p>②計測・制御を行なっている身の回りの機器を取り上げて説明する.</p>

2.4 「プログラムによる計測・制御」で利用する実習用教材の開発

「プログラムによる計測・制御」で利用する実習用教材(以下, 実習用教材と記述する)について, どのような事例が報告されているかを, 先行研究を調査することにした.

西ヶ谷ら(2008)^[2-13]は, ドリトルを用いて自律型ロボットを制御するプログラムの作成を行なっている. ドリトルは, 学校教育用として開発されたオブジェクト指向プログラミング言語の1つで, 一般的な中学生の学習能力で, プログラムを容易に作成できる. また, 自律型ロボットを用いた計測・制御学習に対して, 生徒が興味関心を持つことは認める. しかし, この仕組みでは, プログラムを作成するパーソナルコンピュータと, 作成されたプログラムを実行する自律型ロボット上のコンピュータの2つが必要になるので, 計測・制御システム上に用途別のコンピュータが存在することになる. 中学校学習指導要領解説 技術・家庭編^[2-7]のねらいに, コンピュータを利用した計測・制御の基本的な仕組みを理解させるためには, 計測・制御システムが, センサ, コンピュータ, アクチュエータなどの要素で構成されていることを指導できるようにすることと記述されているが, 自律型ロボットを用いると, 計測・制御システム上に2つのコンピュータが存在することが, 構成要素としてのコンピュータの役割を, 生徒が理解する妨げになる. さらに, プログラムを自律型ロボットに転送しなければ動作確認ができないことから, プログラムを転送する作業が不要な計測・制御システムと比較して, 生徒が期待した動作を実現できるプログラムであるかの確認が迅速に行ないにくい.

また, 吉田ら(2007)^[2-14]は, 自律型ロボットの製作と, C言語を用いて自律型ロボットを制御するプログラムの作成を行なう授業実践を行なっている. この自律型ロボットは, 前述の西ヶ谷ら(2008)^[2-13]と同様の問題を抱えている. また, C言語によるプログラム作成については, 2011年(平成23年)度まで教科書として使われていた『新編 新しい技術・家庭 技術分野』(東京書籍)の第5章「プログラミングと計測・制御」^[2-15]や, 2012年(平成24年)度から教科書として使われている『技術・家庭 [技術分野]』(開隆堂出版)の第7章「プログラムによる計測・制御」^[2-16]に記述されていない. これは, 計測・制御教育を指導する授業時数が7単位時間程度と少ないことや, 中学校

学習指導要領解説 技術・家庭編^[2-7]で、プログラムを作成する際に、命令語の意味を覚えさせるよりも、課題を解決するための処理手順を考えさせることに重点を置いて指導する必要があると記述されていることを、考慮したからである。つまり、C 言語の命令語は、一般的な中学生の学習能力では直感的に理解しにくいので、限られた授業時数では C 言語でのプログラム作成は容易ではないと判断されていることになる。

さらに、山岸ら(2004)^[2-17]は、HSP (Hot Soup Processor)を用いて作成したプログラムで、インタフェースに接続した制御対象物の計測・制御を行なっている。HSP は、BASIC を基本にした文法を用いて、効果的な GUI プログラムを簡単に作成できるので、実用ツール、ゲーム等が作成できるスクリプト言語の 1 つである。このインタフェースの特徴は、USB でコンピュータと常時接続する方法であるので、プログラムをコンピュータからインタフェースに転送する作業が不要なことである。このことによって、生徒が期待した動作を実現できるプログラムであるかの確認が迅速に行なえる上に、プログラムの転送エラーが発生することはない。また、コンピュータから USB を通じて給電する方法であるので、外部電源が不要である。しかし、HSP の仕様により、予約語と同等レベルで用いることができるシステム操作ソフトウェアの命令を、生徒の学習能力や発達段階に沿って技術担当教員が作成することは、困難である。

続いて、市販されている LEGO MINDSTORMS, Arduino, Gainer が、実習用教材として利用することの可能性について調査をした。

評価の観点には、①低価格であること、②2.4.1 で述べるインタフェースの基本仕様をある程度満たしていることにした。価格を評価する理由は、山本ら(2013)^[2-60]が、「プログラムによる計測・制御」で利用する教材を、学校備品にすることが望ましいと考えている技術担当教員の割合が 80%であり、3 年前の調査と比較して高くなっていると報告していることと、村松ら(2006)^[2-9]が、計測・制御教育を指導する際に「予算の不足」が阻害要因の 1 つであると指摘していることを、勘案したからである。つまり、実習用教材は学校備品にするので、低価格で購入できる製品でなければならない。

学習教材として世界的に有名な LEGO MINDSTORMS は、嶋田ら(2003)

[2-18]の報告にあるように、制御対象物の製作が非常に容易で、技術担当教員が設定する課題に対して生徒が改造を行なうことも容易である。また、作成した制御対象物を制御するソフトウェアである ROBOLAB または NXT-G は、アイコン操作を主体としたドラッグ・アンド・ドロップを行なうことで簡単にプログラムの作成ができる。ただし、作成したプログラムを LEGO MINDSTORMS に転送しなければ動作確認ができないので、前述の西ヶ谷ら(2008) [2-13]が開発した自律型ロボットの制御と同様の問題を抱えている。

LEGO MINDSTORMS を用いた授業実践は、古平ら(2007) [2-19]が、自律型ロボットを生徒に製作させる授業を行なったと報告している。この授業実践の結果から、生徒の興味・関心を高く継続させることが容易で、技術担当教員の指導を行なう上での負担を軽減できる理想的な実習用教材と言えるが、高価なために、恒常的に使用する学校備品として購入しにくいのが現状である。しかし、価格的な問題を除けば理想的な実習用教材であるので、山本ら(2011) [2-20]が行なった身近にある自動制御された製品の模型を、LEGO MINDSTORMS で製作し制御する実験授業や、山本ら(2012) [2-21]が行なった LEGO MINDSTORMS を利用した教員研修等の様に、限られた範囲で一定の期間だけ使用される場合が多い。

学習教材として世界的に注目されている Arduino [2-22]は、ワンチップマイコンの AVR ATmega 328P (Atmel Corporation 製)を実装したマイコンボードである。制御対象物は、ブレッドボード上で製作するか、シールドと呼ばれる基板を指導目的に合わせて購入する必要がある。この制御対象物を制御するソフトウェアは主として Arduino IDE になり、C 言語に似た構文で記述するので、前述の吉田ら(2007) [2-14]が行なった C 言語でプログラムを作成させる方法と同様の問題を抱えている。

作成したプログラムを Arduino に転送し、Arduino を単独で実行させる方法が主流であるが、コンピュータと USB で常時接続して Arduino を実行させる方法もある。主流である Arduino を単独で実行させる方法は、前述の西ヶ谷ら(2008) [2-13]が開発した自律型ロボットの制御と同様の問題を抱えている。

Gainer [2-23]は、フィジカル・コンピューティングを実現するためのハード

ウェアで、ワンチップマイコンの PSoC (Cypress Semiconductor 製) を実装したマイコンボードである。制御対象物は、主にブレッドボード上で製作する。製作した制御対象物を制御するソフトウェアは、Processing を利用するのが主流であるが、Flash や Max/MSP でも利用することができる。主流である Processing の構文は、C 言語に似た記述になるので、前述の吉田ら (2007) ^[2-14] が行なった C 言語でプログラムを作成させる方法と同様の問題を抱えている。

Gainer は、USB でコンピュータと常時接続する方法であるので、前述の山岸ら (2004) ^[2-17] が用いたインタフェースと同様の特徴を持っている。

なお、Arduino や Gainer を用いた「プログラムによる計測・制御」の実践に関する先行研究を、CiNii で調査した結果、2013 年 (平成 25 年) 12 月 19 日時点では発見できなかった。しかし、Arduino と Gainer については、多くの書籍が出版されており、「プログラムによる計測・制御」のインタフェース部分として十分活用することができる機能を有し、学校備品として購入できる価格帯でもあるので、今後は実践報告が出てくるであろう。

このような現状を踏まえて、配当される 7 単位時間で、生徒の学習能力や発達段階に沿った計測・制御実習が指導できるように、技術担当教員の要望に添ってカスタマイズすることが出来るインタフェースと、プログラムの作成が容易なシステム操作ソフトウェアとを、新たに開発する必要があると考えた。

前述の山本ら (2013) ^[2-60] の報告と、村松ら (2006) ^[2-9] の指摘を尊重して、本研究の実習用教材は、保護者が購入する副教材とするのではなく、理科の実験教具と同様に中学校が購入する備品とし、保護者の負担を軽減する方針で開発することにした。

2.4.1 インタフェースの開発

実習用教材としての計測・制御システムは、そのシステムを構成するセンサ、コンピュータ、アクチュエータ、インタフェースの機能を、容易に理解できるように、可視化し物理的に独立させなければならないと考えた。

特にインタフェースは、直接情報を伝達できないセンサとコンピュータ間

や、コンピュータとアクチュエータ間の橋渡しをして、情報を伝達するための役割を担っていると、容易に理解できるようにすることが重要である。

実習用教材としてのインタフェースの基本仕様を、次の 7 つに決定した。

- ① 可視化し物理的に独立させる。
- ② 開ループ制御と閉ループ制御の実習が行なえるようにする。
- ③ 入手が簡単な汎用品のアクチュエータやセンサが接続できるように汎用性を持たせる。
- ④ 技術・家庭科の内容「B エネルギー変換に関する技術」の項目である「エネルギー変換に関する技術を利用した製作品の設計・製作」^[2-6]で製作させた製作品を、接続させられる。
- ⑤ プログラムの転送エラーが発生する可能性がある転送作業を行なう必要がないように、コンピュータとインタフェースを、USB で常時接続する方法を採用する。
- ⑥ 生徒の考え通りに計測・制御システムが動作しない場合に、原因となる箇所の特定を容易にするために、目視で動作確認が行なえるようにする。
- ⑦ 動作電源は教室の電源事情を考慮して、コンピュータから USB を通じて給電する方法を採用することによって、外部電源を不要にする。

2011 年(平成 23 年)度まで教科書として広く使われていた『新編 新しい技術・家庭 技術分野』(東京書籍)の第 5 章「プログラミングと計測・制御」では、センサを使って室温を一定に保つ装置、迷路を脱出する自律型ロボット、障害物を避けて走るセンサカーを、計測・制御システムとして紹介されている^[2-15]。

これらの計測・制御システムをすべて指導することは現実的ではなく、これらの計測・制御システムを念頭に置き、指導する生徒の学習能力や発達段階に沿って、指導内容を選択する必要がある。したがって、指導内容をどの計測・制御システムに決定しても、対応できるインタフェースを開発しなければならない。

前述の教科書でセンサとアクチュエータの使用数を調べると、センサは最

大 4 個を使用していたことや、アクチュエータは DC モータが最大 3 個と、電気の ON/OFF で動作をする LED 等を 2 個使用していたことが分かった。

この結果から、開発するインタフェース Ver. 1.0 が計測・制御を行なえる機能を、前述のインタフェースの 7 つの基本仕様を維持した上で、センサを最大 4 個、DC モータを最大 3 個、電気を ON/OFF する回路を最大 2 個に決定した^[2-24]。

開発したインタフェース Ver. 1.0 で留意した点は、センサやアクチュエータの接続を、迅速に行なうことが出来るように、ドライバ等の工具を使用しないで接続できる汎用端子を採用したことや、各アクチュエータを接続する端子を、赤色と黒色の一对にしたことである。接続端子をこの色に決定した理由は、この赤色と黒色の接続端子から通電する場合に、生徒が日常生活で得られた経験知から、赤色の接続端子は+極で黒色の接続端子は-極を連想すると考えたからである。したがって、電気を ON/OFF するアクチュエータを接続する端子は、赤色の接続端子を+5V に、黒色の接続端子をグランド(0V)に設定した。また、DC モータを接続する端子は、赤色の接続端子に DC モータの+側を接続し、黒色の接続端子に DC モータの-側を接続すれば、2.4.2 項で述べるシステム操作ソフトウェアの追加命令である[モータ]の初期設定で、DC モータを正回転させられるようになる。

さらに、センサやアクチュエータの動作確認を、目視で行なえるように LED を配置した。特に、DC モータを接続する端子には、その接続端子付近に LED を配置し、接続している DC モータのどちらの接続端子が+極なのかを、LED の点灯で目視できるようにしている。

なお、DC モータを動作させる電源は、インタフェース Ver. 1.0 と同一の電源にせず外部電源を利用することにした。これは、コンピュータから USB を通じてインタフェースに供給される電流は 500mA までに制限され、DC モータを動作させるだけの電流が確保できないからである。今回の仕様により、開発したインタフェース Ver. 1.0 では、電圧が 20V 以下で電流が平均 1A(ピーク時 2A)以下で動作する DC モータの制御ができるように設計できた。

試作したインタフェース Ver. 1.0 の回路図を図 2-1 に示す。また、その試作品を図 2-2 に示す。

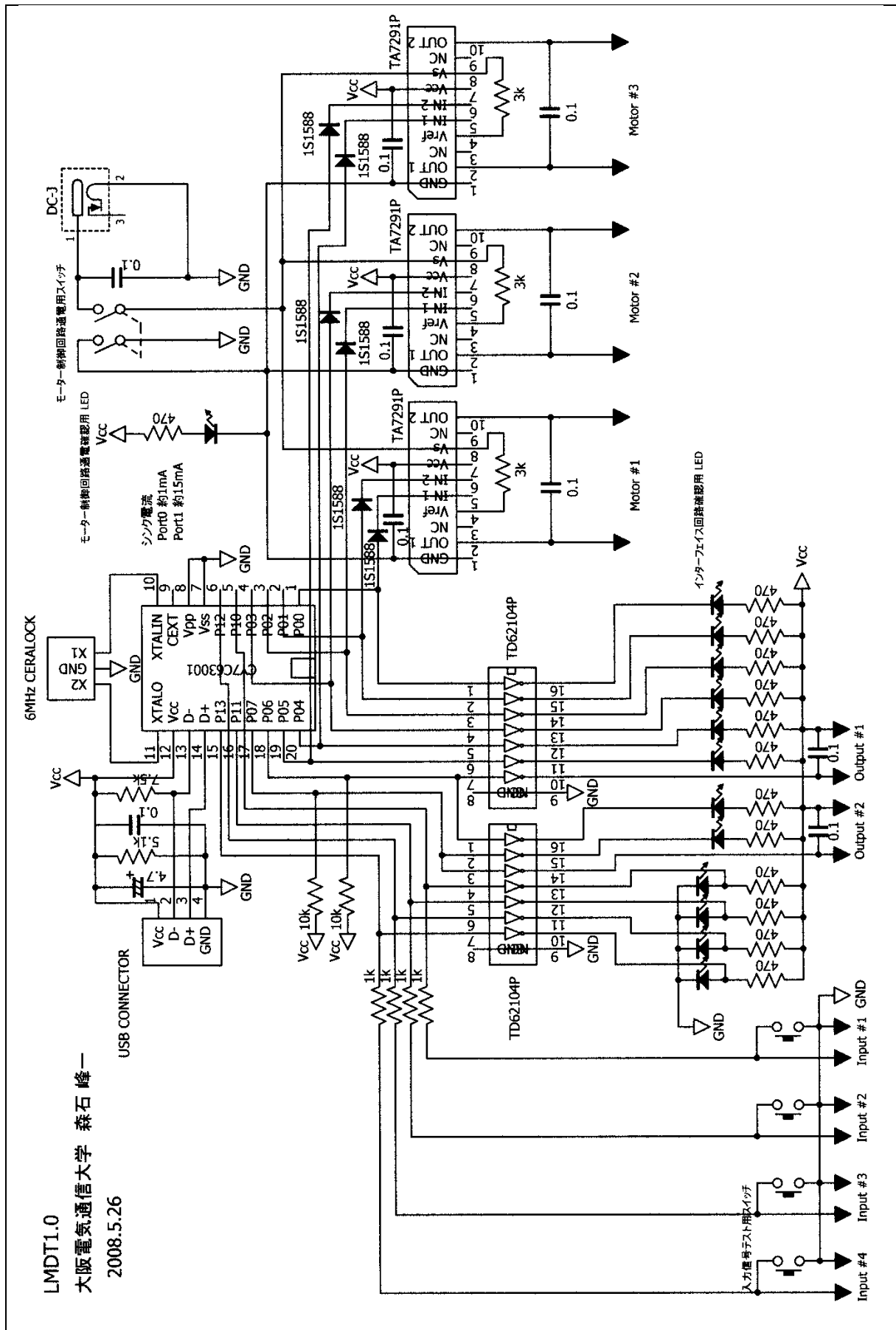


図 2-1 インタフェース Ver. 1.0 の回路図

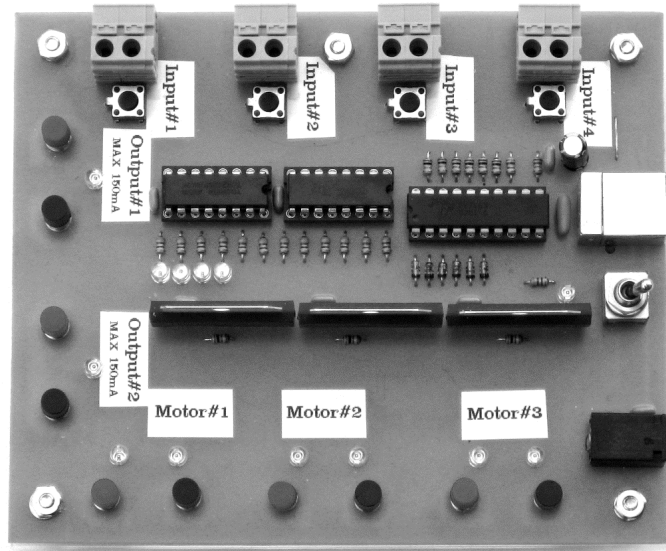


図 2-2 試作したインタフェース Ver. 1.0

制御工学を学んでいない教職生 4 名に、インタフェース Ver. 1.0 を用いて、フォークリフト (3 個のモータで構成され、駆動系に 2 個のモータを持ち、前進、後退、左右回転が可能で、残る 1 個のモータでフォークリフトを上下に移動させることができる) の制御を試行させたところ、デモンストレーション用に作成したプログラムを参考にして、筆箱を移動させるプログラムを短時間で作成していた。この教職生たちへの口答調査から操作環境が簡単に理解できることで、計測・制御教育に興味を持つことができる実習用教材であることが分かった。

教職生に対する内容と同一の試行を、技術担当教員 3 名にも行なった。

結果、インタフェース Ver. 1.0 が計測・制御学習を指導することが可能な実習用教材であることが分かったが、①計測・制御実習でも、簡単なもの作りが出来るようにすること、②電子部品の点数を少なくし、容易にインタフェースの説明ができるようにすること、③生徒 1 人が 1 台のインタフェースを使用できる環境が理想的であるので、中学校が購入する備品であれば、できるだけ安価なインタフェースにすること等の要望があることも分かった。

この要望を受けて、改良型のインタフェースを新たに **Base configuration unit Ver.2.0** と名づけ、開発に取りかかった^[2-25]。

Base configuration unit Ver.2.0 の設計で最も留意した点は、中学校が備品として容易に購入できるように、安価な部品で製作が出来るような構成に

することである。また、インタフェース Ver. 1.0 のように、様々な指導内容に対応できるインタフェースを開発するのではなく、できる限りシンプルなインタフェースを製作し、指導内容に沿って機能を拡張する方法に転換することも必要である。

インタフェース Ver. 1.0 を構成している電子部品の中で、比較的高価な部品は、DC モータ用フルブリッジドライバ IC とワンチップマイコン、接続端子である。

DC モータ用フルブリッジドライバ IC (TA7291P) は、DC モータを制御する実習を行なう場合のみに利用するようにし、Base configuration unit Ver.2.0 には実装しない方針にした。

ワンチップマイコンは、インタフェース Ver. 1.0 で使用した CY7C63001A (Cypress Semiconductor 製) を継続せず、その 1/10 の価格で購入できる ATtiny2313 (Atmel Corporation 製) に変更した。なお ATtiny2313 は、安価であるが 11 個のデジタル入出力ポートを制御することができる。

また、接続端子を使用せずに、様々なセンサやアクチュエータを生徒に実装させ、もの作りを体験させる方法としてブレッドボードに着目した。

ブレッドボードを使用することに決定した理由は、2009 年(平成 21 年)当時、三田市立長坂中学校の技術担当教員であった浅田が、生徒に計測・制御システムをブレッドボード上で製作させる指導を、数年間行なってきた実績があることを知ったからである^[2-26]。この取り組みから、中学生が、ブレッドボードの使い方を理解できることが分かった。

これらの方針をまとめると、Base configuration unit Ver.2.0 の仕様は、前述のインタフェースの 7 つの基本仕様を維持した上で、次の 6 つを追加したものになる。

- ① コンピュータからの命令でアクチュエータを制御する機能と、センサの計測結果をコンピュータに伝える機能のみの必要最低限の機能にする。
- ② センサやアクチュエータを実装したブレッドボードと接続できるように、14 本のピン(丸ピン IC ソケット・両端オスピン)を実装する。
- ③ 実装する 14 本のピンを用いて、後述する I/O control unit 等を増設で

きるように拡張性を持たせる。

④ 8ビットを基本単位とする考え方を，体験を通して学習させられるように，8個までのアクチュエータを接続できるようにする。

⑤ 3個までのセンサを接続できるようにする。

⑥ 自己復帰型の過電流保護用ヒューズであるリセットブル保護素子を実装し，Base configuration unit Ver.2.0内またはブレッドボード上で，短絡や過負荷等が発生することで過電流が流れても，コンピュータを破損しないようにする。

この仕様により，作成した Base configuration unit Ver.2.0 の回路図を図 2-3 に示す。また，その試作品を図 2-4 に示す。

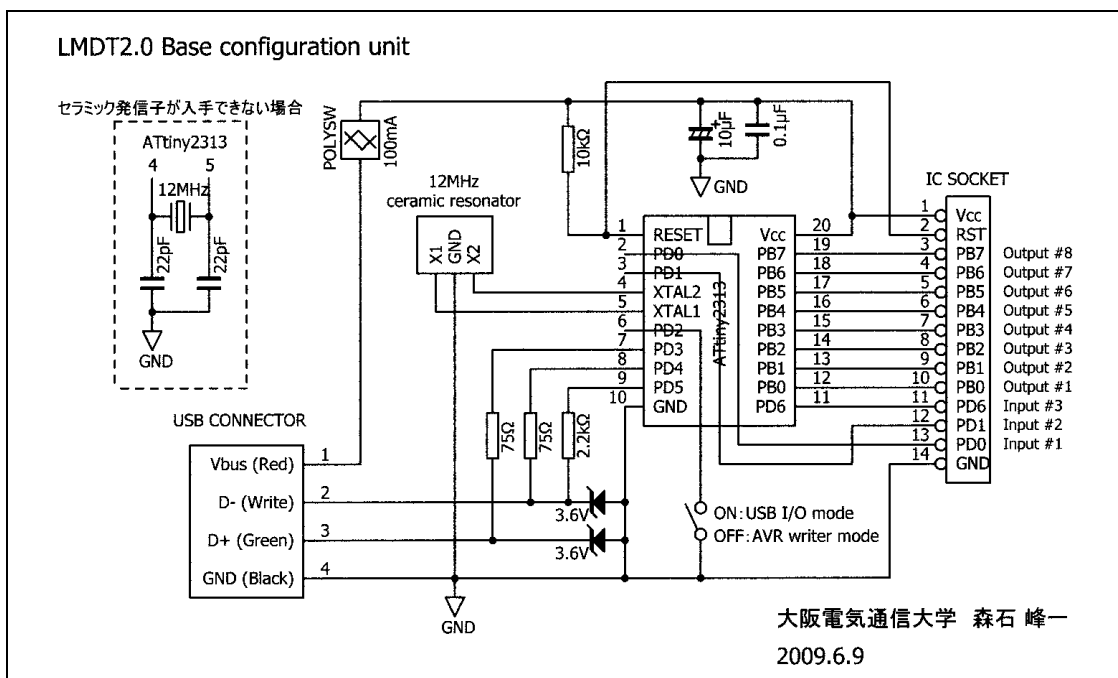


図 2-3 Base configuration unit Ver.2.0 の回路図

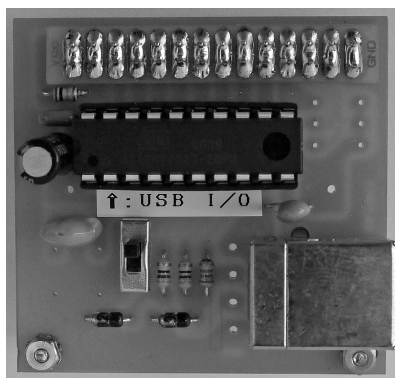


図 2-4 試作した Base configuration unit Ver.2.0

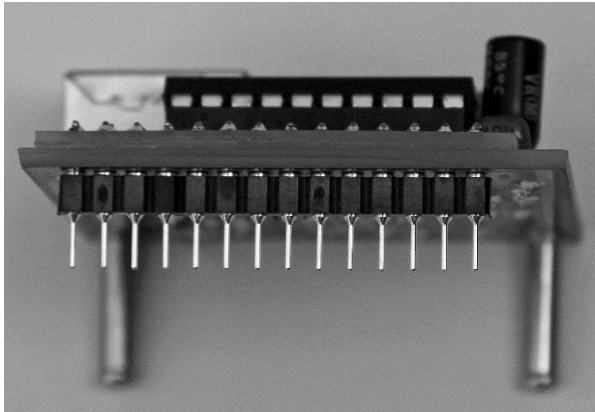


図 2-5 ブレッドボードや、I/O control unit と接続するピン

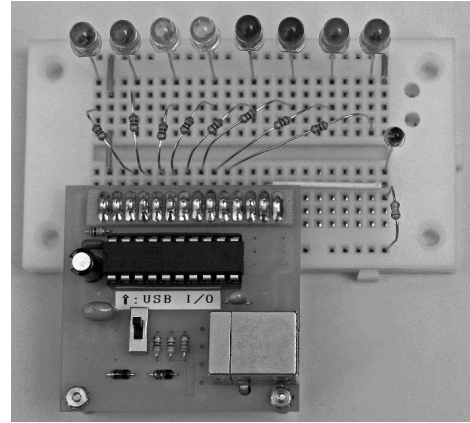


図 2-6 Base configuration unit Ver.2.0 にブレッドボードを接続した一例

ブレッドボードや I/O control unit と接続するための、Base configuration unit Ver.2.0 のピンを図 2-5 に示す。さらに、アクチュエータとして 8 個の LED と、センサとしてフォトランジスタ 1 個を実装したブレッドボードと、Base configuration unit Ver.2.0 とを接続した一例を図 2-6 に示す。

また、計測・制御実習を指導する前に、Base configuration unit Ver.2.0 が正常に動作をするかを、技術担当教員が確認できるように、制御用に 8 個の LED を、計測用に 3 個の ON/OFF スイッチを実装した I/O control unit も作成した。この I/O control unit の回路図を図 2-7 に示す。また、その試作品を図 2-8 に示す。さらに、I/O control unit を Base configuration unit Ver.2.0 に接続した状態を図 2-9 に示す。

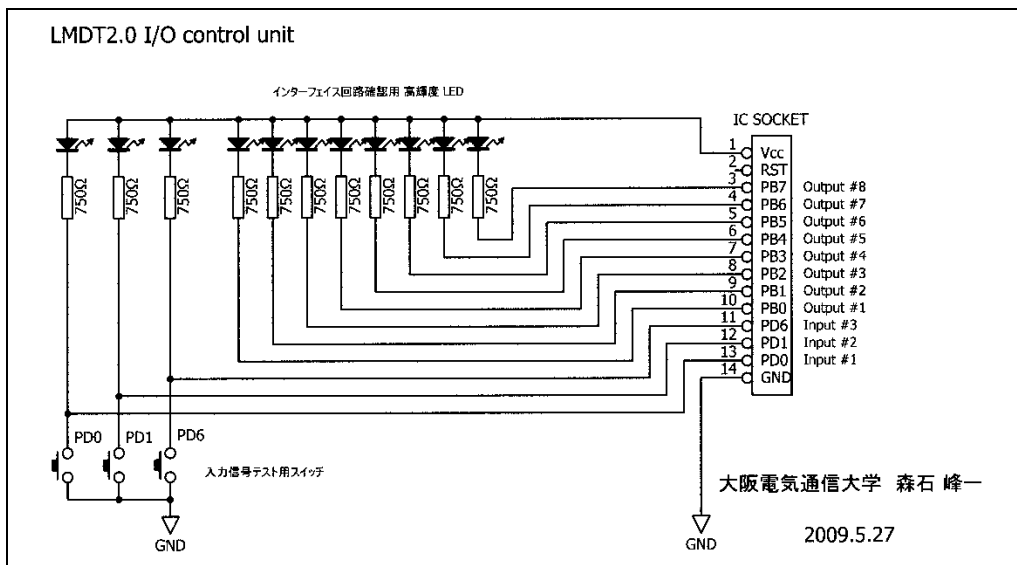


図 2-7 I/O control unit の回路図

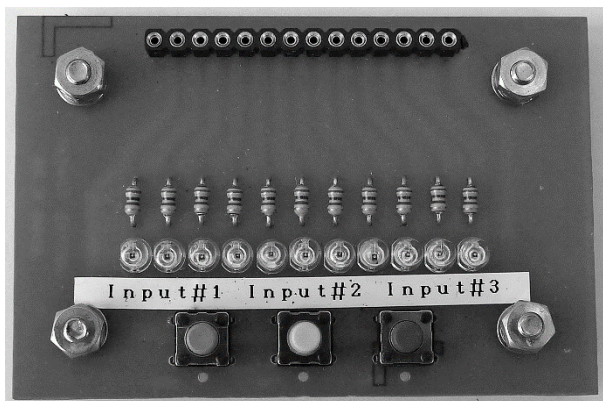


図 2-8 完成した I/O control unit

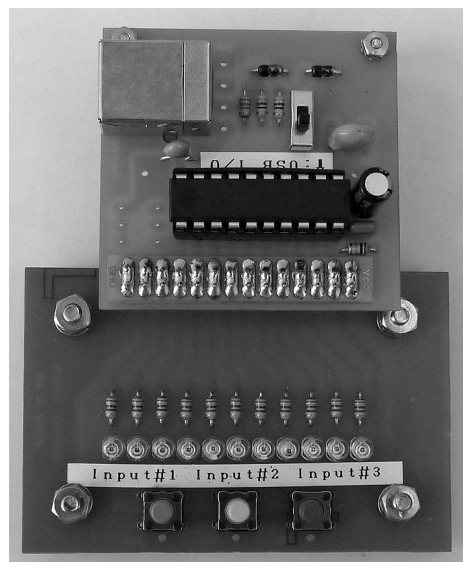


図 2-9 I/O control unit を Base configuration unit Ver.2.0 に接続した状態

技術担当教員5名に，計測・制御学習を指導することが可能な実習用教材であるかの確認をしてもらう目的で，ブレッドボード上にアクチュエータとして8個のLEDと，センサとしてフォトトランジスタ1個を実装(図2-6と同一の回路)してもらい，Base configuration unit Ver.2.0で計測・制御をする試行を行なった．試行後の口答調査では，計測・制御学習を指導することが可能な実習用教材であることが分かったが，次の意見や要望もあった．

- ・ブレッドボード上にセンサやアクチュエータを取り付け，回路を製作することができる生徒もいるが，本校では，ブレッドボードのどの穴同士が，導通しているかを憶え続けられない生徒もいるので，ブレッドボードの活用は難しい．
- ・Base configuration unit Ver.2.0の動作確認をするには，I/O control unitと接続しなければならないが，この作業は繁雑であるので，Base configuration unit Ver.2.0にI/O control unitの機能を付加してはどうか．
- ・I/O control unitのようなBase configuration unit Ver.2.0に容易に増設できる拡張ユニットを，実習内容に沿って製作してほしい．
- ・生徒は，Base configuration unit Ver.2.0に実装する14本のピンを，折る危険性がある．

この意見や要望を参考にして、Base configuration unit Ver.2.0の改良に取りかかった^[2-12]。なお、このインタフェースをBase configuration unit Ver.2.1bと名づけた。

Base configuration unit Ver.2.0でブレッドボードを採用した理由は、技術担当教員から「プログラムによる計測・制御」の実習であっても、もの作り教育の一環として、センサやアクチュエータ等の電子部品を用いた電子回路を製作する指導が必要であると、要望されたからである。

この要望の実現は必要なことであるかを、中学校学習指導要領解説 技術・家庭編で確認することにした。中学校学習指導要領解説 技術・家庭編では、「プログラムによる計測・制御」のねらいを「計測・制御のためのプログラムの作成を通して、コンピュータを用いた計測・制御の基本的な仕組みを知り、簡単なプログラムの作成ができるようにするとともに、情報処理の手順を工夫する能力を育成する」と記述されている^[2-7]。このねらいから、「プログラムによる計測・制御」でのもの作りは、センサやアクチュエータ等の電子部品を用いて回路を製作することではなく、プログラムの作成であることが読み取れる。したがって、電子部品を用いて回路を製作することの教育的効果は認めながらも、中学校学習指導要領解説 技術・家庭編のねらいに沿ってプログラムの作成に重きを置くことにしたので、今回は、電子部品を用いて回路を製作させる指導は行なわない方針にした。

この方針で、Base configuration unit Ver.2.1bの仕様を決定し、様々な計測・制御システムの指導が行なえるように、Base configuration unit Ver.2.1bに増設することで、その機能の拡張ができる用途別の拡張ユニットを開発することにした。なお、Base configuration unit Ver.2.1bと用途別の拡張ユニット等の接続は、ピンが折れる危険性を最小限にするために、D-SUB25ピンコネクタを使用することに決定した。

Base configuration unit Ver.2.1bの仕様は、前述のインタフェースの7つの基本仕様を維持した上で、次の5つを追加したものである。

- ①用途別の拡張ユニットの増設ができる拡張性と、多様なアクチュエータやセンサが接続できる汎用性を持たせる。

- ② 8ビットを基本単位とする考え方を，体験を通して学習させられるように，8個までのアクチュエータを接続できるようにする．
- ③ 3個までのセンサを接続できるようにする．
- ④ 単体で動作確認が行なえるように，制御用に8個のLEDを，計測用に2個のON/OFFスイッチと1個のフォトトランジスタを実装する．
- ⑤ 自己復帰型の過電流保護用ヒューズであるリセットブル保護素子を実装し，Base configuration unit Ver.2.1b内で，短絡や過負荷等が発生することで過電流が流れても，コンピュータを破損しないようにする．

試作した Base configuration unit Ver.2.1b を図 2-10 に示す．また，その回路図を図 2-15 に示す．

用途別に増設する拡張ユニットとして，DC モータ 4 個を，正転・逆転や停止の制御ができる Motor control unit Ver.2.1b と，生徒にとって高電圧または大電流(アンペア単位)が必要なアクチュエータ 8 個の制御ができる High voltage control unit Ver.2.1b を開発した．

Motor control unit Ver.2.1b は，様々な電圧で動作をする DC モータを接続することが予想できるので，接続する DC モータの動作電圧に対応した外部電源を用いる方法にし，電圧が 20V 以下で電流が平均 1A(ピーク時 2A)以下で動作する DC モータの制御ができるように設計した．

その他の設計する上で留意した点は，① DC モータを接続する端子は，感電や短絡を防止する目的で，家庭用の分電盤でも利用可能な定格が 250V-10A の接続端子を実装すること，② その接続端子付近に LED を配置し，接続

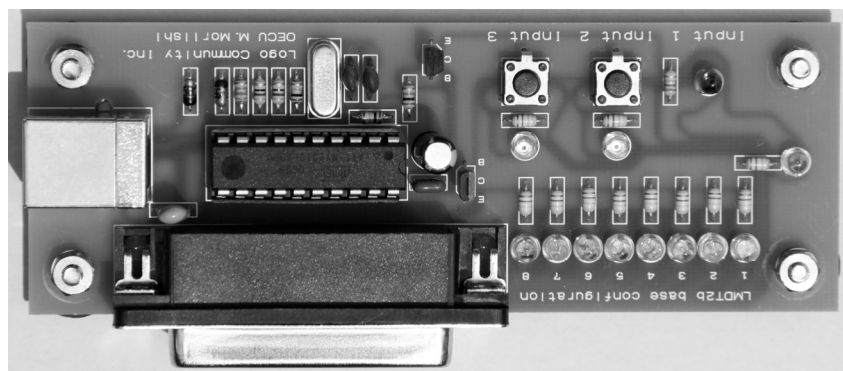


図 2-10 試作した Base configuration unit Ver.2.1b

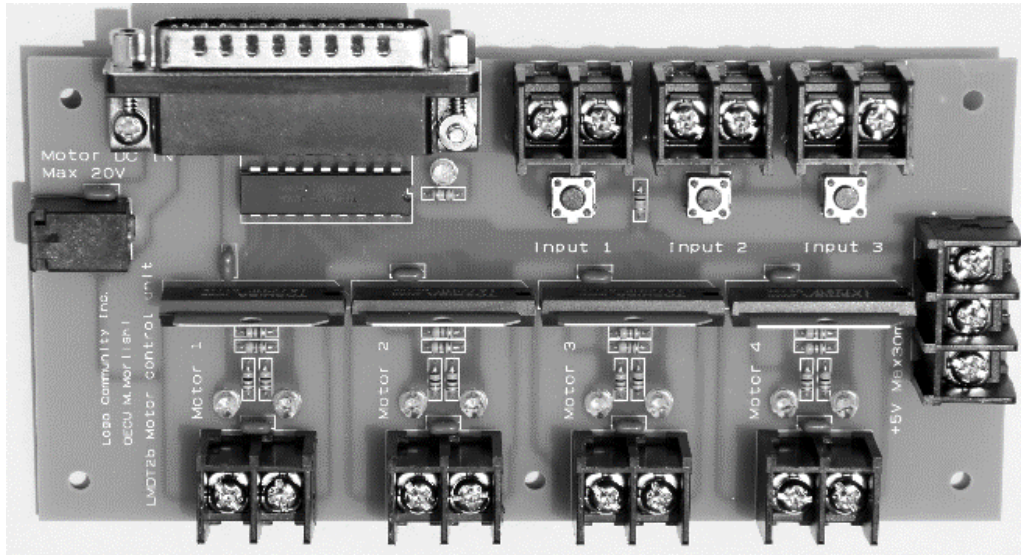


図 2-11 試作した Motor control unit Ver.2.1b

している DC モータのどちらの接続端子が+極なのかを，LED の点灯で目視できるようにすること，③センサを取り付けることなく計測の動作確認が行なえるように，3 個の ON/OFF スイッチを実装することである。

試作した Motor control unit Ver.2.1b を図 2-11 に示す．また，その回路図を図 2-16 に示す．

Motor control unit Ver.2.1b を増設すれば，技術・家庭科の内容「B エネルギー変換に関する技術」の項目である「エネルギー変換に関する技術を利用した製作品の設計・製作」^[2-6]で製作させたロボットや自動車等の DC モータをアクチュエータとした作品を制御することが可能になる．

High voltage control unit Ver.2.1b は，家庭用電源である AC100V で動作をするアクチュエータの制御ができるように，AC250V または DC30V 以下の電圧で 5A 以下の電流であれば ON/OFF 可能な電磁リレーを実装している．このように，高電圧・大電流を制御することを想定しているので，接続端子は，感電や短絡を防止する目的で，家庭用の分電盤でも利用可能な定格が 250V-10A の接続端子を実装することにした．また，アクチュエータの動作確認を，目視で行なえるように各電磁リレーに対応した LED を実装した．

試作した High voltage control unit Ver.2.1b を図 2-12 に示す．また，その回路図を図 2-17 に示す．

2012 年(平成 24 年)度から教科書として広く使われている『技術・家庭 [技

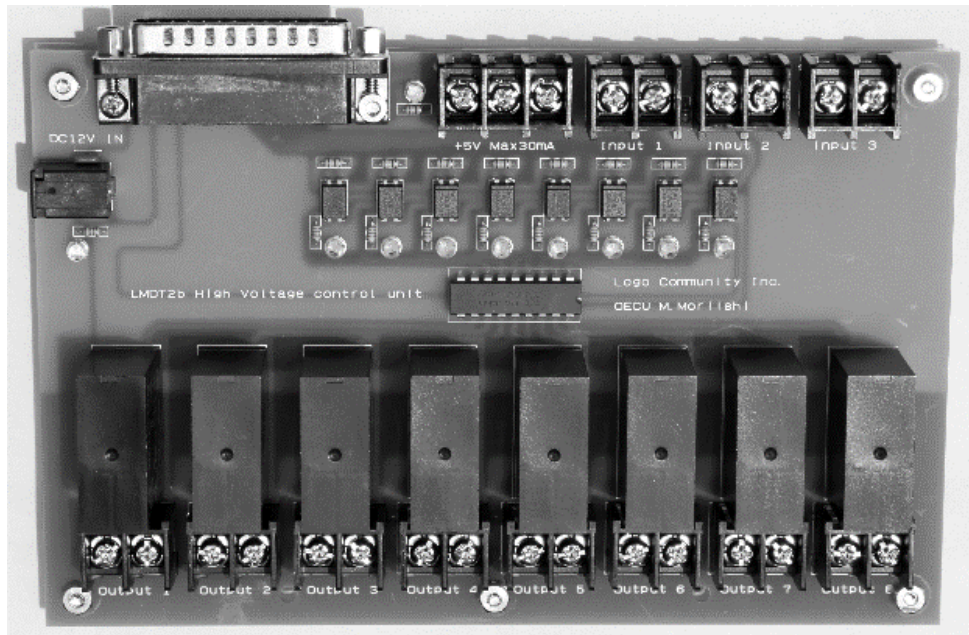


図 2-12 試作した High voltage control unit Ver.2.1b

術分野]] (開隆堂出版)の第 7 章「プログラムによる計測・制御」に、参考実習例として「ホームオートメーションに挑戦しよう」が紹介されている [2-27]。High voltage control unit Ver.2.1b を増設すれば、ホームオートメーションとして取り上げられている外灯の ON/OFF 制御等を製作することができる。

なお、各拡張ユニットの増設は、D-SUB25 ピンコネクタを用いて Base configuration unit Ver.2.1b と直接接続する (図 2-13, 図 2-14)。

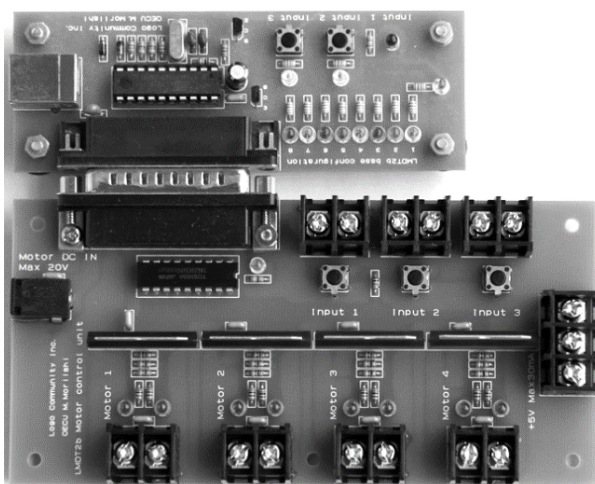


図 2-13 Motor control unit Ver.2.1b との接続

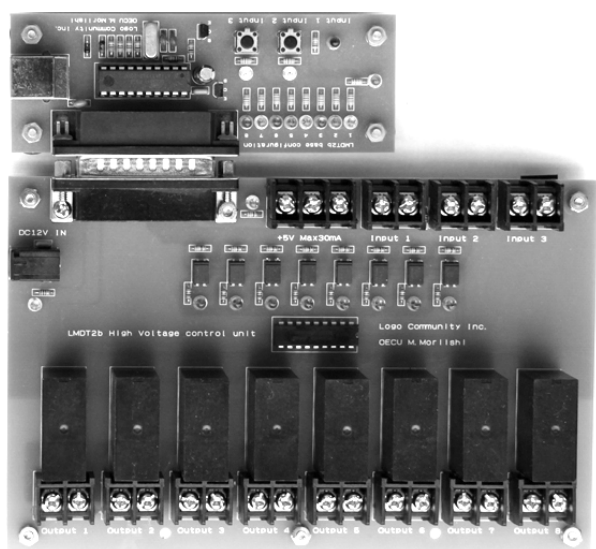


図 2-14 High voltage control unit Ver.2.1b との接続

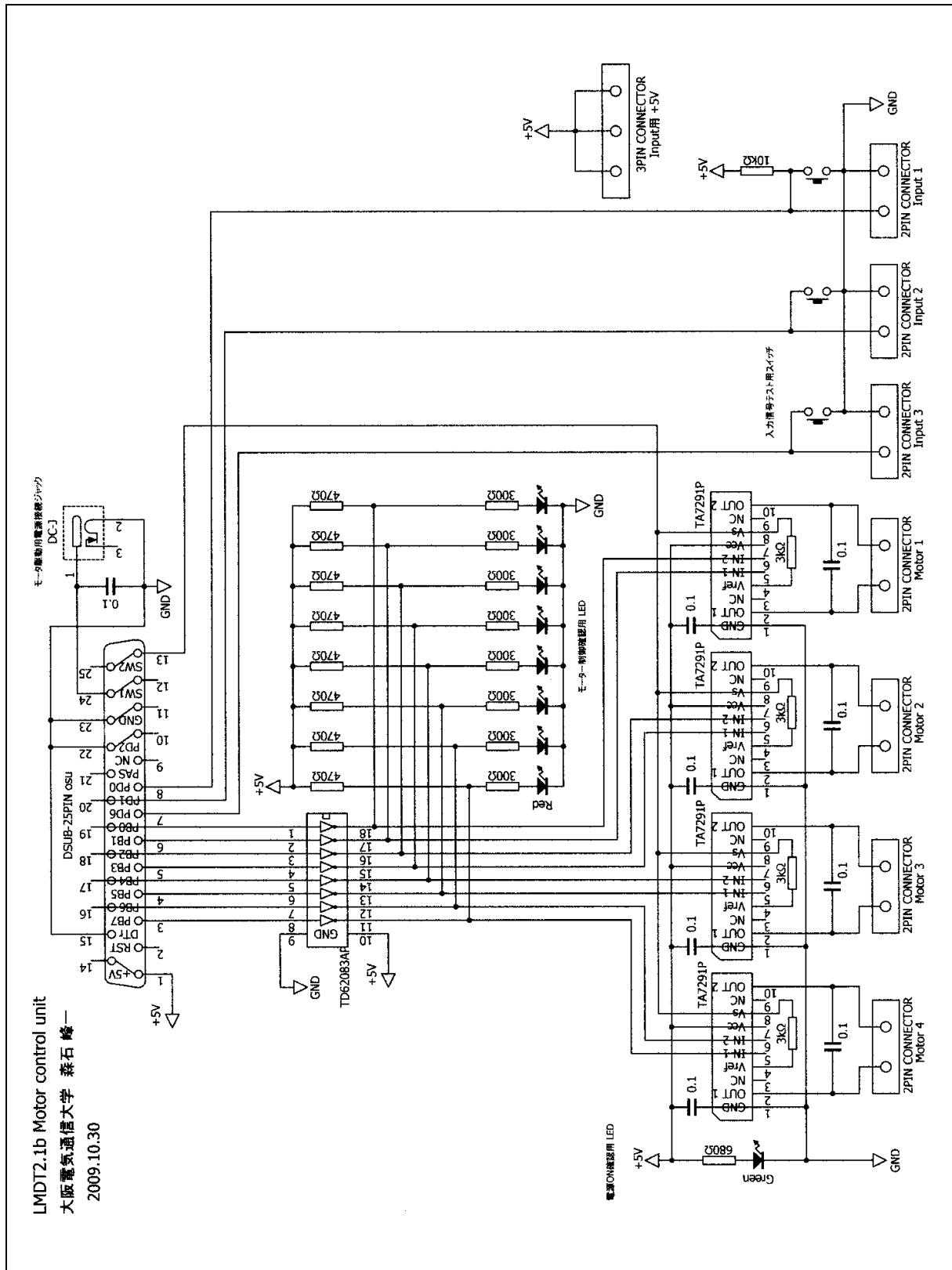


図 2-16 Motor control unit Ver.2.1b の回路図

2.4.2 計測・制御システムを操作するソフトウェアの開発

システム操作ソフトウェアに求められる条件としては、簡単なプログラム作成で開ループ制御と閉ループ制御が体験させられ、目的や条件に応じて、生徒がプログラムを簡単に変更できることが重要である^[2-7]。この条件を実現するためには、生徒が簡単にプログラムを作成できるように、記述する命令語の働きを容易に推測できることも必要になる。さらに、生徒の学習能力や発達段階に合わせて、技術担当教員が命令語を追加できることが望ましい。

多くの生徒は、システム操作ソフトウェアで初めてプログラムの作成を経験する。したがって、システム操作ソフトウェアは、①プログラムを作成することに対する苦手意識を固定化させないように配慮すること、②興味関心を持続させられること、③プログラムの働きを理解させることが容易であることが望ましい。また、システム操作ソフトウェアでプログラムを作成した経験者が、他のソフトウェアでプログラムを作成する場合の基礎・基本を形成することも重要である。これらを勘案して、システム操作ソフトウェアの開発を、汎用性に富んだプログラミング言語を基にし、簡単にプログラムの作成ができるように配慮して行なうことが望ましいと考えた。

技術・家庭科で用いるプログラミング言語は、1989年(平成元年)3月に告示された学習指導要領で「情報基礎」^[2-2]の導入が決定されて以来、伝統的にBASICとLogoが利用されている。

BASICについては、田中ら(2004)^[2-28]が、事前にBASICインタプリタが書き込まれているPIC16F877(Microchip Technology製のワンチップマイコン)モジュールを利用した教材を開発し、それを用いたライントレーサーを制御する授業計画を考案し実践している。プログラミングは、コンパイルを行わずPIC16F877に書き込むことができるBASIC言語による総合開発環境ソフトを使用している。また、Logoについては、本郷(1999)^[2-29]が、日本語Logoを用いて、タートルグラフィックス教材、音楽教材、制御教材及び数量関係教材を取り入れた学習を行い、各教材の相違がプログラミングの情報処理手順の理解に及ぼす影響について検討している。結果は、制御教材(信号の点滅を制御する手順の作成)を取り入れて学習したグループが、それ以外の教材で学習したグループに比べ、学習内容の理解に関する評価問題で

高い得点を得たと述べている。

中学校での利用実績やこれらの先行研究を勘案すると、BASIC と Logo は、中学生の能力で学習することが可能なプログラミング言語として、一定の評価を得ていると言える。

森山(1999)^[2-30]は、BASIC 処理系と Logo 処理系を用いた制御構造の学習が中学生の思考過程に及ぼす影響について検討した結果を、①BASIC 学習群に比べ Logo 学習群では、プログラミングの思考過程に対するリフレクションが有意に深まる。②BASIC 学習群では、制御構造の学習後に十分な思考のスキル(課題を分割するスキル、エラーを探索・修正するスキル、問題表象を修辭的な観点から見なおすスキル等)が獲得されにくいのに対して、Logo 学習群では、中局的なプログラムの設計と点検に有効な思考のスキルが比較的短期間の内に獲得されやすい等の点から、BASIC 処理系に対する Logo 処理系の認知的な優位性を示唆していると述べている。また、生徒にとってプログラミングの基本的な考え方を「思考のスキル」として学ぶのに Logo が適しているとも述べている。

この報告から、生徒がプログラムの設計や作成する能力で BASIC と比較すると、短期間でその能力の向上が期待できる Logo に注目した。

また、Logo は、プリミティブ(予約語)だけではなく、プロシージャ(ルーチン化したプログラム)や変数の名称も、日本語で記述できる特徴がある。

林ら(1991)^[2-31]は、技術・家庭科において、従来から行われている機械領域の機構学習に、制御学習を組み合わせるための教材を開発している。この報告の中で、プログラミング言語を日本語で記述することができる Logo を用いることによって、プログラミングに起因するトラブルを除くことができたと述べている。この内容から、中学生にプログラムの作成をさせる場合は、日本語で記述することができる Logo を使用することが望ましいと言える。

また、2013年(平成25年)12月に開催された、公立中学校教育研究会の技術科研究部主任会において、次のような意見があった。

- ①当初 BASIC で指導していたが、生徒は英語の命令を憶えるのが苦手のようで、タイプミスが多くあった。日本語の命令が使える Logo に変更

すると、タイプミスが減り指導がしやすくなった。

- ② 日本語の命令は、生徒にデバッグさせる際に、教員が訂正箇所の確認をするのが容易で指導がしやすい。
- ③ 日本語の命令を用いることで、生徒同士でデバッグができる。

したがって、技術担当教員は、中学生の特性から日本語でプログラムを作成することが望ましいという意見である。

これらのことから、基礎的・基本的な知識と技術を習得させる期間や、生徒自らが情報処理の手順を工夫しプログラムの作成ができる期間が、2.3節の表 2-2 及び表 2-3 で示したように各 3 単位時間程度であっても、Logo を用いることで、開ループ制御学習から閉ループ制御学習へと発展できる環境を、整えやすいと考えた。

本研究では、システム操作ソフトウェアの開発を中学校に導入実績のある Logo Writer を用いて、計測・制御用の追加命令(以下、追加命令と記述する)をプロシージャで作成する方法で行なった^[2-10, 2-16, 2-17]。

この追加命令は、生徒がその機能を容易に推測でき、技術担当教員が指導しやすいように、個数を必要最小限に止め、日本語で記述することにした。また、追加命令は、生徒の不注意で変更されないようにする配慮が必要である。そこで、追加命令が保存されているシステム操作ソフトウェア本体のファイル(付録：LMDT_2_main_routine.LGW)と、生徒がプログラムを作成するファイル(付録：プログラム作成用ファイル.LGW)を分離し、生徒がプログラムを作成するファイルを開くと、自動でシステム操作ソフトウェア本体のファイルを読み込み、実行できる状態にする方法を採用した。

Logo Writer でシステム操作ソフトウェアを開発することに決定したが、インタフェースの中核をなすワンチップマイコンの動作を決定するファームウェアと直接交信する方法を避け、DLL (Dynamic Link Library) を橋渡しにして交信することにした。これは、Logo Writer を使用できない中学校があることを想定して、Excel の VBA (Visual Basic for Applications) 等からでも利用できるように、汎用性を持たせることが望ましいと考えたからである。

この DLL が橋渡しをしてワンチップマイコンと交信する方法により、Logo

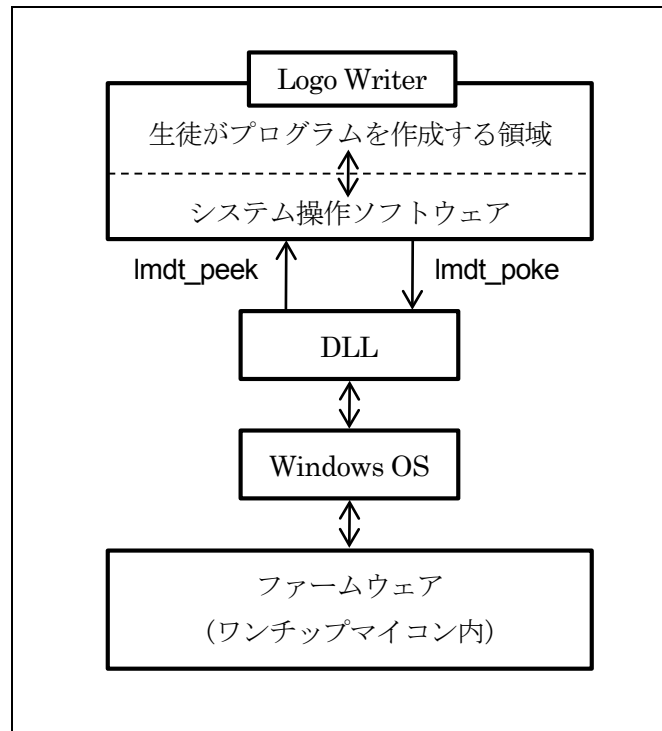


図 2-18 生徒が作成したプログラムとファームウェアの通信 (イメージ図)

Writer は DLL に対して、センサからの計測結果を得る [lmdt_peek] と、アクチュエータを制御する命令を送る [lmdt_poke] の、2 つのコマンドのみで計測・制御システムを動作させることができる。

Logo Write 上の生徒が作成したプログラムと、ワンチップマイコン内のファームウェアの通信をイメージしたものを図 2-18 に示す。

本研究で利用する Logo Writer は、図 2-19 に示すように「ページの表側」、「ページの裏側」、「コマンドセンター」の 3 ブロックで構成されている。

上段左側にある「ページの表側」は、本来タートルの軌跡を描くタートルグラフィックを見るための場所であるが、システム操作ソフトウェアでは、各センサの計測状況を確認する場所に変更している。この場所は閉ループ制御の実習時に利用し、インタフェースに接続している各センサが OFF の場合は、黒色の ■ を表示するが、ON になると赤色の ■ に変化するので、ディスプレイ上で各センサの計測状況を確認することができる。下段にある「ページの裏側」は、生徒がプログラムを作成する場所である。また、上段右側にある「コマンドセンター」は、入力したプリミティブやプロシージャを直ちに実行する場所である。ここでは、生徒が作成したプログラムに付けたプロシージャ名や、単独で動作確認をしたい追加命令を入力することができる。

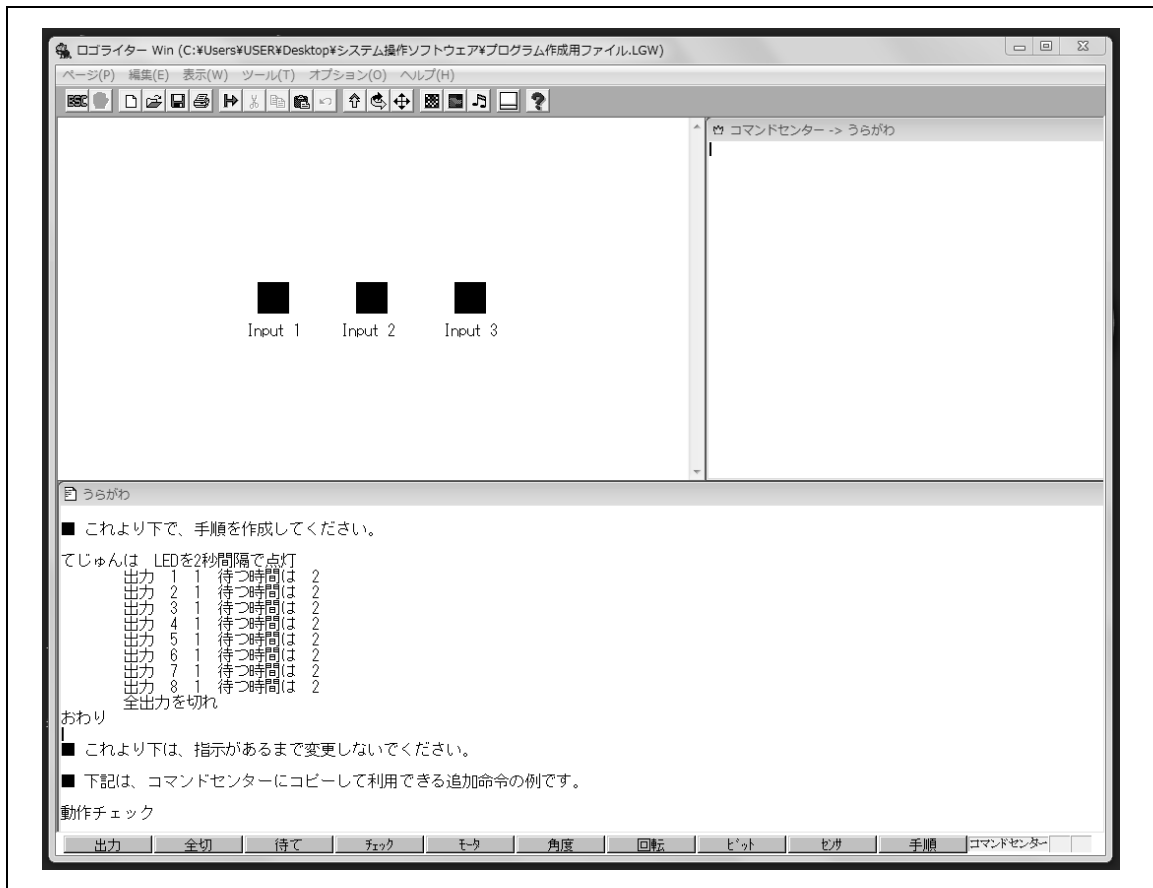


図 2-19 Logo Writer の操作環境

Logo Writer を用いてプログラムを作成する場合は、図 2-20 に示すルールに従う必要がある。「てじゅんは」は作成したプログラムの始まりを示しており、「おわり」はそのプログラムの終了を示している。したがって、プログラムの作成は、「てじゅんは」と「おわり」の間で行なわなければならない。また、この作成したプログラムに名前を付けなければならない。この名前を図 2-20 ではプロシージャ名と記述している。

このルールに従ってプログラムを作成すると、新たなプロシージャが完成する。完成したプロシージャの動作を確認する場合は、そのプロシージャ名をコマンドセンターに入力すると、記述した一連のプログラムが実行される。

計測・制御教育の導入部分である開ループ制御実習で主に用いる追加命令

てじゅんは プロシージャ名
 プログラムを作成する部分
 おわり

図 2-20 Logo Write でプログラムを作成するルール

を，表 2-4 に示す．表 2-4 から表 2-8 の p1 または p2 はパラメータで，追加命令の動作を決定する数値を入力する．アクチュエータを LED にした場合に利用する追加命令は，[出力]，[全出力を切れ]，[待つ時間は]のみでよい．なお，技術担当教員から，Logo Writer のプリミティブに[待つ]があるが，基本単位が 1/60 秒であり，生徒が容易に設定しにくいので，生徒が日常的に利用する 1 秒を基本単位にした追加命令を作成してほしいとの要望があった．この要望を受けて，基本単位を 1 秒にした[待つ時間は]を作成した．これらの追加命令で作成したアクチュエータを制御するプログラムの一例を図 2-21 に示す．

また，[8 ビットでは]は，[出力]を利用した開ループ制御を十分に理解し

表 2-4 開ループ制御用の追加命令

追加命令	動作説明																		
出力 p1 p2	Output 端子に接続した制御対象物の ON/OFF を制御する。 p1 : どの Output 端子を対象にするのかを 1~8 の番号で決定する。 p2 : ON にする場合は[1]を， OFF にする場合は[0]に設定する。																		
全出力を切れ	全ての Output 端子の出力を OFF にする。																		
8 ビットでは p1	8bit の 10 進数表記を用いて， Output 端子に接続した制御対象物の ON/OFF を制御する。 p1 : ON にする Output 端子を 0~255 の数値で設定する。 <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Output端子番号</th> <th>p1の値</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td></tr> <tr><td>2</td><td>2</td></tr> <tr><td>3</td><td>4</td></tr> <tr><td>4</td><td>8</td></tr> <tr><td>5</td><td>16</td></tr> <tr><td>6</td><td>32</td></tr> <tr><td>7</td><td>64</td></tr> <tr><td>8</td><td>128</td></tr> </tbody> </table>	Output端子番号	p1の値	1	1	2	2	3	4	4	8	5	16	6	32	7	64	8	128
Output端子番号	p1の値																		
1	1																		
2	2																		
3	4																		
4	8																		
5	16																		
6	32																		
7	64																		
8	128																		
待つ時間は p1	直前の動作を保持し，次の動作を行なうまでの時間を確保する。 p1 : 設定した数値と[動作時間の単位決定]で得られた数値を掛けることで，直前の動作を保持する時間を決定する．初期値では[1]の設定で約 1 秒間保持する。																		

```

てじゅんは LEDを2秒間隔で点灯
出力 1 1 待つ時間は 2
出力 2 1 待つ時間は 2
出力 3 1 待つ時間は 2
出力 4 1 待つ時間は 2
出力 5 1 待つ時間は 2
出力 6 1 待つ時間は 2
出力 7 1 待つ時間は 2
出力 8 1 待つ時間は 2
全出力を切れ
おわり

```

図 2-21 アクチュエータを制御するプログラムの一例

た上で、8ビットを基本単位とする考え方を学習するための追加命令である。

2.4.1項で述べた Motor control unit Ver.2.1b で利用する追加命令を表 2-5 に示す。DC モータを制御する追加命令は[モータ]のみで、2個の DC モータを利用して、制御対象物を左右に回転させる場合は、[回転角度は]を利用する。この[モータ]と[回転角度は]で作成したプログラムの一例を図 2-22 に示す。

[一回転の測定]は、[回転角度は]の精度を向上させるための追加命令である。また、[右モータは]と[左モータは]は、初期設定であるモータ 1 を駆動系モータの左側に、モータ 2 を駆動系モータの右側に取り付ければ、これらの追加命令を使用する必要はない。つまり、[一回転の測定]、[右モータは]、[左モータは]は、DC モータを制御する際に必ず利用する追加命令ではなく、技術担当教員が必要に応じて指導することを想定した追加命令である。

```

てじゅんは 右に90度回転して停止
右回転
回転角度は 90
全出力を切れ
おわり

てじゅんは 右回転
モータ 1 -1
モータ 2 1
おわり

```

図 2-22 2個の DC モータで制御対象物を駆動させるプログラムの一例

表 2-5 Motor control unit 関係の追加命令

追加命令	動作説明
モータ p1 p2	<p>Motor control unitのMotor端子に接続しているDCモータを、回転または停止させる。</p> <p>p1：どのモータを対象にするのかを決定する値で、モータ1を対象にする場合は[1]に設定する。同様に、モータ2の場合は[2]を、モータ3の場合は[3]を、モータ4の場合は[4]に、それぞれ設定する。</p> <p>p2：対象になるモータの動作を決定する値で、正回転をさせる場合は[1]を、逆回転をさせる場合は[-1]を、停止させる場合は[0]に設定する。</p>
回転角度は p1	<p>直前の動作を保持し、次の動作を行なうまでの時間を確保することで、回転する角度を決定する。</p> <p>p1：指定できるのは0以上の整数で、360で1回転となり、この数値が1回転に要する時間になる。入力した数値と[一回転の測定]で得られた数値を掛けることで、回転する角度を決定する。</p>
一回転の測定	<p>アクチュエータを[回転角度]で制御する場合に、必要な基準値を作成する。この追加命令を実行し、クチュエータが360度回転すれば、一ボードの任意のキーを叩くと、コマンドセンターに[■■一回転の測定終了■■]と表示し、業が完了する。この作業で360度に要した時間を決定し、回転角1度に要する時間を基準値として算出する。</p> <p>※ 初期設定では、モータ1を駆動系モータの左側に、モータ2を駆動系モータの右側に取り付けた状態を想定している。</p>
右モータは p1	<p>どのMotor端子に接続しているDCモータを、駆動系モータの右側に設定するかを決める。設定が完了すると、コマンドセンターに[×番のモータを右側に設定していましたが、○番のモータに変更しました。]と表示する。</p> <p>p1：どのモータを右側のモータに設定するかを決める値で、モータ1であれば[1]に設定する。同様に、モータ2であれば[2]を、モータ3であれば[3]を、モータ4であれば[4]に設定する。</p> <p>※ 初期設定では、モータ2を駆動系モータの右側に設定している。</p>
左モータは p1	<p>どのMotor端子に接続しているDCモータを、駆動系モータの左側に設定するかを決める。設定が完了すると、コマンドセンターに[×番のモータを左側に設定していましたが、○番のモータに変更しました。]と表示する。</p> <p>p1：どのモータを左側のモータに設定するかを決める値で、モータ1であれば[1]に設定する。同様に、モータ2であれば[2]を、モータ3であれば[3]を、モータ4であれば[4]に設定する。</p> <p>※ 初期設定では、モータ1を駆動系モータの左側に設定している。</p>

学習が進行した発展部分の閉ループ制御実習で用いる追加命令を、表 2-6 に示す。主に用いる追加命令は[センサを使う]のみであり、[計測]は、発展的に高度な計測・制御を行なうことを想定した追加命令である。

閉ループ制御のプログラム作成の基礎は、2.3節の表 2-2 で示したように 1 単位時間程度と想定している。この限られた時間で、条件分岐等を多用するプログラムの作成は、平均的な生徒が持つ学習能力では複雑すぎる。最終的には、[計測]を用いてプログラムを作成させることが望ましいが、閉ループ制御実習の初期段階では、複雑なプログラムを作成させる指導を行わずに、簡単なプログラムで閉ループ制御を体験させることが重要であるので、次の 3 つの手順で実現するようにした。

- ①センサが動作した場合に実行する[センサ 2 だけが ON の場合]等の、予め作成してあるプロシージャを確認する。
- ②そのプロシージャ内に、アクチュエータが意図した通りに動作をするためのプログラムを生徒に作成させる。
- ③再帰するプロシージャを生徒に作成させ、その中に[センサを使う]を記述させる。

この手順で作成したプログラムの一例を図 2-23 に示す。

てじゅんは センサカー センサを使う 前進 センサカー おわり	てじゅんは 前進 モータ 1 1 モータ 2 1 おわり
てじゅんは センサ 2 だけが ON の場合 右回転 おわり	てじゅんは 後退 モータ 1 -1 モータ 2 -1 おわり
てじゅんは センサ 3 だけが ON の場合 左回転 おわり	てじゅんは 右回転 モータ 1 -1 モータ 2 1 おわり
てじゅんは センサ 2 と 3 が ON の場合 後退 おわり	てじゅんは 左回転 モータ 1 1 モータ 2 -1 おわり

図 2-23 センサを用いたプログラムの一例

表 2-6 閉ループ制御用の追加命令

追加命令	動作説明
センサを使う	<p>各 Input 端子間をショートさせた場合に、下記のプロシージャ内に作成したプログラムを実行する。</p> <p>■ プログラムを作成することができるプロシージャ</p> <p>[センサ 1 と 2 と 3 全てがOFFの動作] →このプロシージャ内に、センサ1～3の全てがOFF状態であるときのプログラムを作成する。</p> <p>[センサ 1 だけがONの動作] →このプロシージャ内に、センサ1のみがON状態であるときのプログラムを作成する。</p> <p>[センサ 2 だけがONの場合] →このプロシージャ内に、センサ2のみがON状態であるときのプログラムを作成する。</p> <p>[センサ 1 と 2 がONの場合] →このプロシージャ内に、センサ1とセンサ2がON状態であるときのプログラムを作成する。</p> <p>[センサ 3 だけがONの場合] →このプロシージャ内に、センサ3のみがON状態であるときのプログラムを作成する。</p> <p>[センサ 1 と 3 がONの場合] →このプロシージャ内に、センサ1とセンサ3がON状態であるときのプログラムを作成する。</p> <p>[センサ 2 と 3 がONの場合] →このプロシージャ内に、センサ2とセンサ3がON状態であるときのプログラムを作成する。</p> <p>[センサ 1 と 2 と 3 全てがONの場合] →このプロシージャ内に、センサ1～3の全てがON状態であるときのプログラムを作成する。</p>
計測	<p>Input 端子間をショートさせた場合に、変数 [計測結果] に下記の値が代入される。</p> <p>Input 1 をショートさせた場合 → 変数 [計測結果] に[1]が代入される。</p> <p>Input 2 をショートさせた場合 → 変数 [計測結果] に[2]が代入される。</p> <p>Input 3 をショートさせた場合 → 変数 [計測結果] に[4]が代入される。</p> <p>※ 同時に複数の入力があっても対応している。したがって、Input 1 と Input 2 を同時にショートさせた場合は、変数 [計測結果] に[3]が代入される。</p>

表 2-7 動作チェック用の追加命令

追加命令	動作説明
動作チェック	Output1 から Output8 までを順番に 0.5 秒間 ON にする。
光センサチェック	<p>Input1に接続している光センサのON/OFF状態をチェックする。正常な動作では、光センサを指先で覆うと、画面上の「ページの表側」に表示しているInput1はON(赤色)になり、指先を離すとInput1はOFF(黒色)になる。</p> <p>なお、光センサを指先で覆っていないのに、Input1がONになった場合は、光センサをより明るい位置に移動する必要がある。</p> <p>また、キーボードの任意のキーを叩くことで、チェックを終了することができる。チェックを終了した場合は、コマンドセンターに[光センサのチェックを終了!!]と表示する。</p>

表 2-7 に示す[動作チェック]は、インタフェースの全出力ポートが、正常に動作することを確認するための追加命令である。また、[光センサチェック]は、計測・制御実習を行なう教室の明るさで、Input1 に接続している光センサが、正常に動作することを確認するための追加命令である。これらの追加命令は、主に技術担当教員が、授業の準備を行なう際に利用することを想定して作成した。

表 2-8 に示す[動作時間の単位]は、表 2-4 で示した[待つ時間は]の基本単位を変更することができる追加命令である。また、[動作時間の調整]は、計測・制御実習で使用するコンピュータの処理速度によって、[待つ時間は]の差異が生じるので、その[待つ時間は]の精度を向上させるための追加命令である。これらの追加命令は、主に技術担当教員が、計測・制御教育の単元に入る前に利用することを想定して作成した。

このシステム操作ソフトウェアの開発にあたって、学習の初期には、筆者が作成した追加命令や、技術担当教員が指導する生徒の学習能力や発達段階に合わせて作成したプロシージャを活用しながら、新たなプロシージャを作成することを想定しているが、学習能力の向上に従って、生徒自らがプリミティブを多用したプロシージャを作成できるようになることを、期待している。

表 2-8 動作時間の基本設定に関する追加命令

追加命令	動作説明
動作時間の単位 p1	アクチュエータが動作する時間の単位を変更することができる。初期値は60で約1秒に設定している。 p1：指定できるのは0以上の整数で、60が約1秒にあたる。
動作時間の調整	コンピュータの処理時間に差異がある場合に、動作時間を調節するための命令である。 この追加命令を実行し15秒後にキーボードの任意のキーを叩くと、コマンドセンターに[■■動作時間の調整ができました。■■]と表示し調整作業が完了する。

2.5 開発した実習用教材の評価

開発した実習用教材であるインタフェースとシステム操作ソフトウェアを生徒に利用してもらうには、まず、教員が指導できる実習用教材でなければならないと考えた。そこで、筆者が、2009年度(平成21年度)から2011年度(平成23年度)にかけて技術担当教員(以下、受講者と記述する)を対象に実施した教員免許更新講習(以下、単に講習と記述する)において、実際に指導する立場から容易に生徒が活用できる実習用教材であるかの判断を行なうために、(1)簡単なプログラム作成で開ループ制御と閉ループ制御が体験させられ、目的や条件に応じて、生徒がプログラムの変更を簡単にできること、(2)インタフェースの機能を容易に理解させられることの2点について評価した^[2-12]。この教員免許更新講習の日程と受講者数を表2-9に示す。

講習の目標は、コンピュータを用いた計測・制御の基本的な仕組みを説明できることと、計測・制御を行なうためのプログラム作成ができることとし、表2-10に示す講習内容で行なった。なお、すべての講習は、同一の時間帯と

表 2-9 教員免許更新講習の日程と受講者数

講習日	受講者数
2009年(平成21年)8月27日	5名
2010年(平成22年)8月4日	3名
2011年(平成23年)8月19日	7名
合計	15名

表 2-10 教員免許更新講習の講習内容

講習時間	講習内容
10:00～10:40	座学：計測・制御の定義やシステムの仕組み，システムを構成する各要素の働き等を説明する。
10:40～11:10	実習：システム操作ソフトウェアの操作説明，インタフェースの動作確認，プログラム作成方法の説明等を行なう。
11:10～15:40	実習：アクチュエータを制御する開ループ制御のプログラムを，課題に沿って作成させる。 ※ 12:00～13:00は昼食・休憩時間
15:40～15:50	実習：予め用意したセンサとアクチュエータで計測・制御を行なうプログラムを，課題に沿って変更させる。
15:50～16:20	実習：センサとアクチュエータを利用した閉ループ制御のプログラムを，課題に沿って作成させる。

内容で行なった。

この講習では受講者に対して，プレアンケート(15項目)とポストアンケート(20項目)を実施し，その結果から開発した実習用教材の有用性についての評価を得ることができた。ただし，全受講者数が15名であったので，統計的処理が行なえるサンプル数に至っていないと判断し，カテゴリカルデータとして処理をした。

表 2-11 は，「簡単なプログラム作成で開ループ制御と閉ループ制御が体験させられるか」の設問に対するポストアンケートの回答である。この結果から，開ループ制御のプログラム作成については，生徒が簡単にできると判断したが，閉ループ制御のプログラム作成については，やや難しいと判断したと言える。

表 2-11 生徒のプログラム作成能力についてのアンケート結果

	開ループ制御	閉ループ制御
作成できる	13	7
どちらでもない	2	6
作成できない	0	2

(単位：受講者数)

表 2-12 インタフェースの機能の理解状態のアンケート結果

	プレ	ポスト
理解している	6	15
どちらでもない	4	0
理解していない	5	0

(単位：受講者数)

また、ポストアンケートの別の設問である「目的や条件に応じて、生徒が簡単にプログラムの変更ができるか」に対しては、簡単に変更できるが 13 名で、どちらでもないが 2 名、変更は難しいが 0 名であった。この結果から、プログラムの変更については、生徒が簡単にできると判断したと言える。

「受講者がインタフェースの機能を理解しているか」の設問に対して、プレアンケートでは、理解していると回答した受講者は 6 名であったが、ポストアンケートでは、全受講者が理解したと回答をしている(表 2-12)。

また、プレアンケートでインタフェースの機能を生徒に理解させられないと回答した 8 名中 7 名が、ポストアンケートでは理解させられると回答している(表 2-13)。

これらの結果から、受講者は、開発したインタフェースの機能を自ら理解したことから判断して、生徒にインタフェースの機能を理解させる指導ができると推測したと言える。なお、インタフェースを可視化し物理的に独立させることの効果や、プログラムの転送エラーを生じさせないようにすることの効果については、アンケート調査を行っていないので評価はできない。

表 2-13 インタフェースの機能を生徒に理解させられるかのアンケート結果

	プレ	ポスト
理解させられる	4	14
どちらでもない	3	1
理解させられない	8	0

(単位：受講者数)

表 2-14 計測・制御学習を指導する自信についてのアンケート結果

	プレ	ポスト	7 単位時間での指導の可能性
自信がある	3	11	7
どちらでもない	2	2	4
自信がない	10	2	4

(単位：受講者数)

計測・制御学習を指導した経験がないと、プレアンケートで回答した受講者は 12 名いる。表 2-14 では、この 12 名の内 10 名が計測・制御学習を指導する自信がないと回答している。また、ポストアンケートでは、11 名が計測・制御学習を指導する自信があると回答している。これは、開発した実習用教材を用いた実習で得られた知識が大きな影響を与えた結果である。

なお、7 単位時間程度では指導が行なえないと回答した 4 名は、ポストアンケートで指導する自信があると回答した 11 名の内 2 名と、自信がないと回答した 2 名である。

この評価を基にして、学習者支援の立場から、後述の考察を行なった。

システム操作ソフトウェアの設計は、生徒が簡単にプログラムを作成できることを重視しているので、開ループ制御のプログラム作成は、ねらい通りの効果が得られたが、閉ループ制御のプログラム作成は、再考する余地を残している。また、生徒が簡単にプログラムの変更ができるとの評価を得ることができたのは、記述する追加命令の働きを容易に推測できることや、簡単な構文で記述ができること等が影響していると推測できるので、ねらい通りの効果が得られた可能性がある。

インタフェースの役割について、表 2-10 で示した講習の導入部分である座学で簡単に説明しただけであるから、インタフェースの機能の理解については、開発したインタフェースを用いた実習で得られた知識が、表 2-12 で示したように大きな影響を与えている。このことが表 2-13 の生徒にインタフェースの機能を理解させる指導ができると判断したことに関連する。インタフェースの設計は、生徒がその機能を容易に理解できることを重視しているので、ねらい通りの効果が得られた。また、全受講者がインタフェースの

機能を理解できたことは、インタフェースが可視化し物理的に独立していることが、要因の1つであると思われる。

開発した実習用教材を用いた閉ループ制御学習については、ねらい通りの効果が得られた。しかし、表 2-14 で示したように、開発した実習用教材を用いた計測・制御学習を行なう自信はあっても、7 単位時間程度の授業時数では時間が足りないと感じている受講者が多いので、生徒が閉ループ制御のプログラム作成を行なうための手順を再考することや、例題等の補助教材の開発が必要であることが分かってきた。

第 2 章の参考文献

- [2-1]日本教育工学会編：“教育工学事典”，実教出版（2000）
- [2-2]文部科学省：“中学校学習指導要領：第 2 章 第 8 節 技術・家庭(平成元年 3 月)”，http://www.mext.go.jp/a_menu/shotou/old-cs/1322468.htm
- [2-3]文部省：“情報化の進展に対応した初等中等教育における情報教育の推進等に関する調査研究協力者会議 情報化の進展に対応した教育環境の実施に向けて”，（1998）
- [2-4]文部科学省：“中学校学習指導要領(平成 10 年 12 月)”，国立印刷局（1998）
- [2-5]文部科学省：“中学校学習指導要領(平成 10 年 12 月)解説 技術・家庭編”，東京書籍（1999）
- [2-6]文部科学省：“中学校学習指導要領(平成 20 年 3 月)”，東山書房（2008）
- [2-7]文部科学省：“中学校学習指導要領(平成 20 年 9 月)解説 技術・家庭編”，教育図書（2008）
- [2-8]山本 利一，本村 猛能，本郷 健，“プログラムによる計測・制御に関する教員研修内容の提案”，日本教育情報学会，年会論文集（27），2011 年 8 月，pp.210-213，（2011）

- [2-9]村松 浩幸, 岡田 雅美, 阿久 津一史, 兼折 泰彰, 鈴木 善晴, 長谷川 元洋, “中学校技術・家庭科における制御教材の開発と評価：制御教材の3つの開発課題に対応して”, 日本教育工学会論文誌 29(Suppl), 2006年3月, pp.177-180, (2006)
- [2-10]政宗 賢治: “中学校技術・家庭[技術分野]の学習内容を相互に関連付ける指導の在り方 — プログラムによる計測・制御を題材とした教材開発を通して —”, 広島県立教育センター 研究紀要 第38号, pp99-116 (2011), <http://www.hiroshima-c.ed.jp/web/publish/ki/pdf1/kk38/gijilyutu.pdf>
- [2-11]国立教育政策研究所教育課程研究センター: “音楽等質問紙調査 集計結果”,
<http://www.nier.go.jp/kaihatsu/ongakutou/04000573120007001.pdf>
- [2-12]森石 峰一, 横山 宏, 魚井 宏高, “中学校の技術・家庭科の「プログラムによる計測・制御」で利用する学習教材の開発と試用”, 教育システム情報学会, Vol.29, No.4, 2012年10月, pp.190-200, (2012)
- [2-13]西ヶ谷 浩史, 兼宗 進, 青木 浩幸, 紅林 秀治: “アーム付き自律型移動ロボットを使った授業実践”, 情報処理学会研究報告. コンピュータと教育研究会報告 2008(13), pp.17-23 (2008)
- [2-14]吉田 昌春, 山崎 千恵子, 吉田 竹虎: “自律型ロボット入門教材に関する研究”, 岐阜大学教育学部研究報告. 教育実践研究 9, pp.45-52 (2007)
- [2-15]加藤 幸一, 永野 和男: “新編 新しい技術・家庭技術分野”, 東京書籍, pp218-219 (2007)
- [2-16]間田 泰弘: “技術・家庭 [技術分野]”, 開隆堂出版, pp218-233 (2012)
- [2-17]山岸 正人, 藤田 孝夫: “USB インターフェースを利用した制御学習教材の研究”, 山梨大学総合情報処理センター研究報告 8 (2004),
<http://www.ipc.yamanashi.ac.jp/bul/final04/fujita/index.html>

- [2-18] 嶋田 彰子, 針谷 安男, “技術教育における自律型ロボット教材を用いた計測・制御の学習に関する研究”, 宇都宮大学教育学部教育実践総合センター紀要 26, 2003年4月, pp.145-154, (2003)
- [2-19] 古平 真一郎, 石島 隆志, 坂本 弘志, 高橋 脩太, 宮川 こずえ, 伊藤 秀哲, 針谷 安男, “自律型ロボット教材を用いたプログラム学習に対する効果”, 宇都宮大学教育学部教育実践総合センター紀要 30, 2007年7月, pp.539-548, (2007)
- [2-20] 山本 利一, 齋藤 雅宏, “プログラムによる計測・制御を学習する指導過程の提案: 自動制御模型の製作とプログラムによる制御学習”, 教育情報研究: 日本教育情報学会学会誌 27(1), 2011年7月, pp.25-32, (2011)
- [2-21] 山本 利一, 本村 猛能, 本郷 健, 飯塚 嶺, “生活の中に組み込まれた自動制御の仕組みを学習する教員研修の提案”, 日本教育情報学会, 年会論文集 (28), 2012年8月, pp.118-121, (2012)
- [2-22] Massimo Banzi (著), 船田 巧(翻訳): “Arduinoをはじめよう 第2版”, オライリージャパン, (2012)
- [2-23] GainerBook Labo + くるくる研究室: “+GAINER—PHYSICAL COMPUTING WITH GAINER”, オーム社, (2008)
- [2-24] 森石 峰一, 魚井 宏高, “中学校の技術・家庭科における制御の学習を行うための教材開発”, 教育システム情報学会 第33回全国大会講演論文集, 2008年9月, pp.162-163, (2008)
- [2-25] 森石 峰一, 兼宗 進, “中学校の技術・家庭科における制御の学習を行うための教材開発(2)”, 教育システム情報学会 第34回全国大会講演論文集, 2009年8月, pp.252-253, (2009)
- [2-26] 浅田 寿展: “ブレッドボード版のヒダピオ学習回路”,
<http://hidapio.jp/bb.htm>
- [2-27] 間田 泰弘: “技術・家庭 [技術分野]”, 開隆堂出版, pp233 (2012)
- [2-28] 田中 啓勝, 山本 尚登, 藤山 秀公, “PICを用いたコンピュータ制御教材の研究”, 三重大大学教育学部附属教育実践総合センター紀要 24, 2004年3月, pp. 95-101, (2004)

- [2-29]本郷 健, “教材の差異が学習者の情報処理手順の理解に及ぼす影響”, 日本教育情報学会, 教育情報研究: 日本教育情報学会学会誌 14(4), 1999年3月, pp.43-51, (1999)
- [2-30]森山 潤, “プログラミングの学習過程における課題の制御構造及び言語プロセッサが生徒の思考過程に及ぼす影響”, 日本教育情報学会, 教育情報研究: 日本教育情報学会学会誌 15(3), 1999年12月, pp.11-18, (1999)
- [2-31]林 秀昭, 八高 隆雄, “中学校「技術・家庭科」への制御教育導入のための機構-制御組合せ教材の開発”, 横浜国立大学教育紀要 31, 1991年10月, pp.263-270, (1991)

第3章 大学における計測・制御教育の授業設計及び実践

計測・制御教育は、中学校では全生徒が指導されているものの、高等学校では、工業高校、農業高校、水産高校^[3-1]のみで、全生徒数の10.6%^[3-2]しか指導されていない。このような現状であるので、大学での計測・制御教育は、基礎・基本から指導しなければならないことになる。

大学での教育は、中学校や高等学校の教育環境とは異なり、学習指導要領がなく、文部科学省検定教科書もない。また、教授者は教員免許を持つ必要もない。したがって、大学における授業設計は、教育行政面での制約条件がほとんどなく、教授者の自由裁量で行える部分が多い。

この自由度の高い授業設計環境において、優れた授業設計を行なうためには、教育の目的や目標を明確にし、実績がある指導方法を活用することが効果的である。

筆者は、工学部の一員として、大学で実施されている計測・制御教育に関心を持っているので、大学で計測・制御教育を指導することを望んでいたところ、O大学でHidden Curriculum企画委員会が企画した「光・動き出す・未来」プロジェクト(以下、本プロジェクトと記述する)を担当することになり、大学生を対象とした計測・制御教育を行なう機会を得ることができた。

本プロジェクトは、在学生のキャンパス生活を活性化することを目的とし、コンピュータで制御する対象物を、学生自身が考案し製作することを目標としている。

本プロジェクトの目標を達成させるためには、参加する学生が自由な発想で製作ができるような環境作りが重要であるので、学生の主体的な学習を促し創造性を高めるために有効な指導方法であるProject Based Learning(以下、PBLと略記する)を活用することが望ましいと考えた。

現在では、このPBLを創成科目という用語を用いて、多数の大学が学科単位で実施している。この創成科目の考え方と実施案は、都倉(2000)^[3-3]が報告している。また、武田(2005)^[3-4]は、創成教育の理論と技法を整理する試みを行っている。さらに、渡邊ら(2011)^[3-5]は、宇都宮大学の工学部1年生を対象にした初期導入科目をPBL方式で行い、実施したアンケートの分析から、実習型の必修科目として満足できる結果が得られたと報告してい

る。実践については、2000年(平成12年)頃から現在まで、様々な大学が工夫し、良い成果を上げた報告が多数ある。

3.1 「すだち方式」による計測・制御教育の授業設計

菅井(1987)^[3-6]は、コンピュータ教育において、学習者は、能動的・主体的な学習者観に立つ必要があることを、「学習者中心の学習」の授業モデルを用いて論じている。

この考えはPBLを行なう上で、教員に求められることや、学生に必要な基礎能力についての重要なポイントを示唆している。つまり、PBLを行なう場合に、教員に求められることを具体的にあげると、①教員が自らの意識を、講義者(Lecturer)や指導者(Instructor)から、促進者(Facilitator)や助言者(Adviser)に変革すること、②教員が学生に対して、一方向コミュニケーションを基本にした講義形式から脱却すること、③学生が主体となり、能動的に学ぶことができる学習環境を整備すること、④学生同士が、相互に協力する仕組みを取り入れること等になる^[3-7]。

また、学生に必要な基礎能力は、④PBLの到達目標までの計画を立案できること、⑤PBLの開始から学生自身が設定した到達目標までの構想を、ある程度段階的に考えられる知識、意識(価値観や動機付け)、スキル(技術)等を身に付けていること、⑥PBLの到達目標を、成果物の製作や成果報告の発表等の具体的な方法で、他者に伝えられることである。

しかし、前述の④～⑥までの基礎能力について、教員は自らの経験からその必要性を認識しているが、理解の乏しい学生はその必要性を十分認識していないことがある。このような現状があるので、PBLの効果を着実に高めるためには、学生に求めるこれら3つの基礎能力を高める授業を、PBLを実施する前に行なうべきであると考え、この授業に、PBL Oriented Instruction(以下、POIと略記する)と名づけた。

POIでは、学生の知識、意識、スキルを、向上させなければならない。したがってPOIは、教員主導で進めるが、一方向コミュニケーションを基本にした講義形式に終始するのではなく、机間指導による観察やアンケート等で学習者特性をきめこまかく把握した上で、その状態に即応した指導ができる

ように、必要最小限に授業内容を細分化した講義と、その講義終了後に学生の授業内容に対する理解度を確認する課題に取り組ませることを1ユニットとし、当該ユニットを学生が理解し、次のユニットに進めるように、十分配慮したユニット方式を取り入れた教授方法で行なうことが重要である。また、教員と学生とが意見を交わしながら進める双方向コミュニケーションを基本にした授業形式で行なうことや、学生自身が学習すべき方向を明確に認識し、継続できるように支援をすることも求められる。

この POI は教員主導で進め、学生の持つ基礎能力が PBL を開始できるレベルに達した時点で、教員主体の指導から学生主体の PBL へとつなげる指導方法であることから、動物が巣立ちをする行動になぞらえて、「すだち方式」と名づけた。この「すだち方式」のイメージを図 3-1 に示す。

POI での「すだち」として、教員が学生の意識を刺激すること (stimulate = 「す」)、教員は学生とよく対話をする (dialogue = 「だ」)、教員は

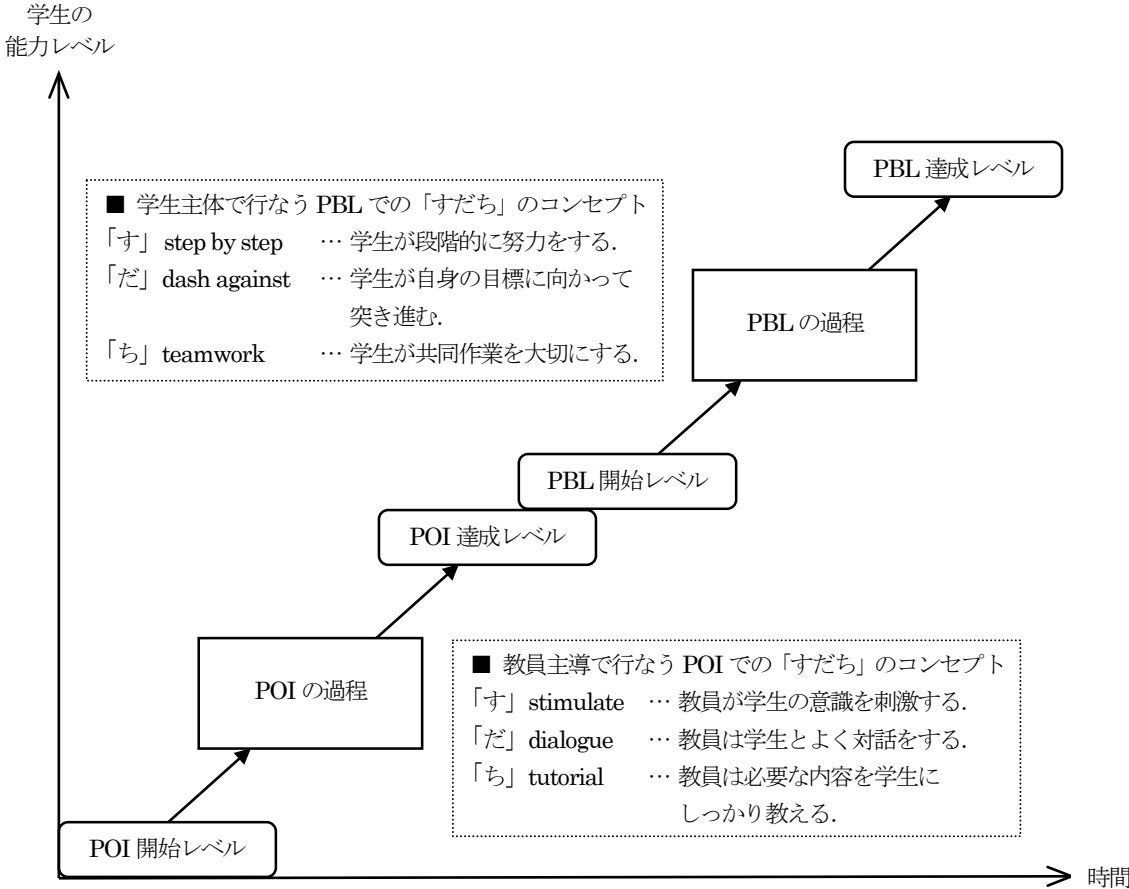


図 3-1 「すだち方式」による学生の能力レベル遷移

必要な内容を学生にしっかり教えること (tutorial = 「ち」) に配慮している。この POI では、学習能力に格差を持った学生が受講しても、POI 達成レベルでは、PBL 活動時に学生に必要な 3 つの基礎能力のベースになる能力を、PBL を開始することができるレベルまで到達するように指導をする。

また、PBL での「すだち」として、学生が段階的に努力をすること (step by step = 「す」)、学生が自身の目標に向かって突き進むこと (dash against = 「だ」)、学生が共同作業を大切にすること (teamwork = 「ち」) ができるように配慮している。PBL 開始レベルでは、POI 達成レベルで学生が身に付けた 3 つの基礎能力をベースにして、PBL の到達目標までの計画を、学生自身が立案できる能力が要求される。この計画に従って進行する PBL の過程では、設定した到達目標までの構想を、ある程度段階的に考えられる能力が要求される。さらに、PBL 到達レベルでは、到達目標を、成果物の製作や成果報告の発表等の具体的な方法で、他者に伝えられる能力が要求される。

3.2 POI の実践

「すだち方式」で行なう本プロジェクトは、参加希望の学生 14 名 (表 3-1) を対象に、2010 年 (平成 22 年) 8 月から 12 月まで実施した。

表 3-1 参加を希望した学生の所属情報

参加者番号	学部・専攻	学科・分野	学年
1	工学部	電気電子工学科	1
2		電気電子工学科	2
3		応用化学科	2
4		機械工学科	2
5		電子機械工学科	4
6	情報通信工学部	情報工学科	2
7		通信工学科	1
8			3
9, 10			4
11	工学研究科 電子通信工学専攻	通信・ネットワーク工学分野	修士 2
12, 13		光・電子デバイス工学分野	修士 1
14			修士 2

- Q10 あなたの行動パターンを、次の中から選んでください。
①考えがまとまってから行動する方・②考えながら行動する方・③考えずに行動する方
- Q11 グループでの作業について、次の中から選んでください。
①得意な方・②どちらかといえば得意な方・③どちらかといえば苦手な方・④苦手な方
- Q12 グループの中では、どのようなタイプなのかを、次の中から選んでください。
①リーダー型・②アイデア提供型(プロモータータイプ)・③行動分析型(アナリストタイプ)・
④単独行動型・⑤参加協力型(リーダーについて行くタイプ)
- Q13 機械をいじることについて、次の中から選んでください。
①好きな方・②どちらかといえば好きな方・③どちらかといえば嫌いな方・④嫌いな方
- Q14 電気をいじることについて、次の中から選んでください。
①好きな方・②どちらかといえば好きな方・③どちらかといえば嫌いな方・④嫌いな方
- Q15 コンピュータを用いたプログラム作成について、次の中から選んでください。
①得意な方・②どちらかといえば得意な方・③どちらかといえば苦手な方・④苦手な方・
⑤プログラム作成をしたことがない
- Q16 前記の問いで、「得意な方・どちらかといえば得意な方・どちらかといえば苦手な方・苦手な方」と答えた人に対するの質問です。
最も使い慣れたプログラミング言語を、次の中から選んでください。[複数回答が可能です。]
①Basic・②C・③Logo・④HSP・⑤Java・⑥その他 []
- Q17 電子回路の設計について、次の中から選んでください。
①得意な方・②どちらかといえば得意な方・③どちらかといえば苦手な方・④苦手な方・
⑤設計したことがない
- Q18 電子回路の製作について、次の中から選んでください。
①得意な方・②どちらかといえば得意な方・③どちらかといえば苦手な方・④苦手な方・
⑤製作したことがない
- Q19 ハンダ付けについて、次の中から選んでください。
①得意な方・②どちらかといえば得意な方・③どちらかといえば苦手な方・④苦手な方・
⑤ハンダ付けをしたことがない

図3-2 参加予定者の状態を把握するために実施したアンケート内容

本プロジェクトの実施前に、参加者が本プロジェクトの目標である「コンピュータで制御する対象物を、参加者自身が考案し製作すること」を、PBLで達成するために必要な基礎能力である、①PBLの到達目標までの計画を立案できること、②PBLの開始から参加者自身が設定した到達目標までの構想を、ある程度段階的に考えられる知識、意識(価値観や動機付け)、スキル(技術)等を身に付けていること、③PBLの到達目標を、成果物の製作や成果報告の発表等の具体的な方法で他者に伝えられることを、どの程度身に付けているかを把握する必要があるのでアンケートを行なった。このアンケート内容は、参加者の資質や性格に関することや、グループで作業を行なう場合の振る舞い方、電気や機械に対する関心度、電子回路の製作についての関心度等である(図 3-2)。

その中でもグループ作業についての意識に関する回答(表 3-2)と、グループ内の役割についての意識に関する回答(表 3-3)に注目した。また、本プロ

表3-2 グループ作業についての意識

レベル	該当者数
得意な方	1人
どちらかといえば得意な方	5人
どちらかといえば苦手な方	2人
苦手な方	2人
計	10人

表3-3 グループ内の役割についての意識

タイプ	該当者数
リーダー型	0人
アイデア提供型(プロモータータイプ)	0人
行動分析型(アナリストタイプ)	1人
単独行動型	3人
参加協力型(リーダーについて行くタイプ)	6人
計	10人

プロジェクトの目標を達成するためのベースになる能力であるコンピュータを用いたプログラム作成についての知識に関する回答(表 3-4)及び、電子回路の製作についてのスキルに関する回答(表 3-5)に注目した。ただし、参加者 14 名中 10 名の回答であったので、カテゴリカルデータとして処理を行なった。

アンケートの元データから表 3-2 と表 3-3 の関係を調べると、表 3-2 でグループ作業が「得意」と回答した 1 名は、表 3-3 では参加協力型と回答していた。また、表 3-2 でグループ作業が「どちらかといえば得意」と回答した 5 名中 1 名は、表 3-3 では行動分析型と回答し、他の 4 名は参加協力型と回答していた。したがって、グループ作業を得意と回答した参加者であっても、指導力の発揮やアイデア提供を積極的には行ないない難いことが予想できた。

表 3-4、表 3-5 から、プログラム作成についての知識や、電子回路の製作についてのスキルについても、偏りがないことが分かった。さらに、電子回路の製作を行なったことがない者も参加していることが分かった。

表3-4 プログラム作成についての知識

レベル	該当者数
得意な方	1人
どちらかといえば得意な方	4人
どちらかといえば苦手な方	2人
苦手な方	3人
計	10人

表3-5 電子回路の製作についてのスキル

レベル	該当者数
得意な方	1人
どちらかといえば得意な方	3人
どちらかといえば苦手な方	4人
苦手な方	0人
製作したことがない	2人
計	10人

このような状況であるので、PBLを実施する前に参加者に求める3つの基礎能力のベースになる能力を、身に付けさせる授業を行なう必要があると考え、本プロジェクトの開始部分である2010年(平成22年)8月7, 21, 28日の3日間は、「すだち方式」によるPOIを行なうことにした。この授業では、コンピュータで対象物を制御する上で必要なプログラムの作成と、対象物を構成する電子回路の設計と製作の基本的な知識を、参加者に定着させることを目標として、授業内容を細分化して講義し、講義を終えると直ちに、そこで得られた知識を活用する演習や実習を行なった。本研究では、この講義、演習、実習を1つにまとめた教授方法を「ユニット」と呼ぶことにしている。

本プロジェクトで行なった授業を具体的に示すために、通常行なわれている授業と、「すだち方式」によるPOIでの指導方針の比較を表3-6に示す。さらに、学生が行なう学習活動の比較を表3-7に示す。

本プロジェクトを行なう上で指導上 2 つの条件を、①インタフェースは、2.4.1項で述べた **Base configuration unit Ver.2.1b** を使用させること。②参加者自身が比較的使い慣れたプログラミング言語を用いてプログラムを作成させることに設定した。

条件①を設定した理由は、汎用性に富み、シンプルなインタフェースを使用すれば、そのインタフェースが持つ機能を容易に理解できることから、インタフェースに接続する制御対象物を比較的容易に製作出来ると考えたからである。また、2.4.2項で述べたように、**Base configuration unit Ver.2.1b**を制御するコマンドは、DLLを橋渡しにしてプログラミング言語と交信する方法を採用しており、DLLと交信することが可能なプログラミング言語であれば、センサからの計測結果を得る[lmdt_peek]と、アクチュエータを制御する命令を送る[lmdt_poke]の、2つのコマンドのみで**Base configuration unit Ver.2.1b**を制御することができることも決定した理由の1つである。なお、今回使用した**Base configuration unit Ver.2.1b**を、本プロジェクトでは「ちょびっとプラス」と愛称で呼ぶことにした。

条件②を設定した理由は、「ちょびっとプラス」の主要部品であるワンチップマイコンに対しては、C言語(以下、Cと記述する)でプログラムを作成することが一般的であるが、全参加者がCを熟知しているとは考えにくいこ

表 3-6 教員の指導方針で比較

通常行なわれている授業	「すだち方式」による POI
■ 教員の授業意識	
教科書に沿って、できるだけ分かりやすい説明を十分にする。	対話を頻繁に行ない学生の意識を刺激すると共に、ユニット方式を用いて分かりやすく進める。
■ 演習で利用させるプログラミング言語	
教員が予め定めた 1 つのプログラミング言語を利用させる。	利用させるプログラミング言語を予め定めず、個々の学生が使い慣れたプログラミング言語を選択出来るようにする。
■ 講義，演習，実習の連動性	
電子回路の設計を行なう講義，演習，実習では，別々の科目として行なわれ，それらの学習内容は，必ずしも連動や同期がとられていない場合がある。	授業内容を細分化したユニット形式を活用する。個々のユニットでは，課題を解決させる上で必要な講義を行ない，講義を終えると直ちに，そこで得られた知識を活用する演習や実習を行なう。
■ 電子回路の設計目標	
電子回路の設計を行なう講義では，電子回路の設計理論を理解させることを目標としている。	電子回路の設計理論を理解させるだけではなく，設計した電子回路を実際に製作させ，動作チェックを行なわせる実習までを視野に入れている。
■ 学生の理解状態の把握	
予め計画した授業内容を，授業時間内に終了するようにしているが，教員が授業中に学生の理解状態を把握することは稀である。	予め計画した授業内容を，授業時間内に終了させることに固守せず，学生の理解状態を把握しながら授業の進度を調整する。
■ 学習者特性及び学習状態の把握方法	
学習者特性をきめ細かに把握していない。 学習状態を把握する方法は，レポート提出や試験の実施等で行なっている。	学習者特性を知るために，各授業日の開始直後と終了直前にアンケートを行なう。 学習状態を把握する方法は，授業中に発問や机間指導を積極的に行なうことや，学生が演習問題の解答として作成したプログラムを，授業終了後にチェックする等で行なう。

表 3-7 学生の学習活動面で比較

通常行なわれている授業	「すだち方式」による POI
■ 受講する学生の意識	
<p>教員の説明をよく聞き，ノートを取り，教科書の内容を理解する。</p>	<p>教員からの刺激を受け止め，対話を頻繁にし，ユニット方式に適応する。</p>
■ 受講する目的	
<p>多くの学生は，よい成績を取ることや，単位を取得することを目的と考えている。</p>	<p>チーム単位で考案した制御対象物の製作が出来るように，基本的な能力を身に付けることである。</p>
■ 学習時の受講態度	
<p>予め教員によって設計された授業内容に従って，受動的に受講することが多い。</p>	<p>PBL を成功させる知識とスキルを得るために，POI では，能動的に学習をする。</p>
■ 実習時に製作する電子回路	
<p>予め教員によって決められた電子回路を，全ての学生が製作をする。</p>	<p>学生自身が考案するので，全ての学生が同一の電子回路を製作する可能性は少ない。</p>
■ 演習で利用するプログラミング言語	
<p>予め教員によって決められたプログラミング言語で受講するので，そのプログラミング言語の環境に慣れてから課題に取りかかる。</p>	<p>使い慣れたプログラミング言語を選択できるので，授業開始時より課題に集中することができる。</p>
■ 理解できなかった場合の解決方法	
<p>授業終了後に教員に質問するか，自習して理解しなければならない。</p>	<p>授業中であっても，次のユニットの学習に進むまでに，教員に質問をすることができる。</p>
■ 回路設計の教材	
<p>予め教員によってモデル化された電子部品の規格表を教材にして回路設計を行なう。 モデル化された電子部品以外で，実際に動作可能な電子回路を製作することは難しい。</p>	<p>電子部品の特性をテクニカルシート(電子部品の規格表)で確認しながら回路設計を行なう。 実際に動作可能な電子回路を，どの部品で構成するのが望ましいかを判断できるようになる。</p>
■ 電子部品の選択範囲	
<p>予め教員によって指定された電子部品を利用する。</p>	<p>予め教員によって用意された数種類の電子部品から，学生が必要と考える部品を選択し利用する。</p>

とや、POIでCを基礎から指導する時間がとれないためである。O大学の学生のほとんどが、BASICかCまたはC++のいずれかを、2年生までに学習している。つまり、Cを学習していない学生も参加を希望する可能性があることになる。そこで、多くの学生が参加しやすい条件にするため、プログラムの作成を、Cに限定せずに、C、C++、BASICのいずれかから、個々の参加者が比較的使い慣れたプログラミング言語を選択できるようにした。また、本プロジェクトを始める前に、参加を予定している学生に対して実施したアンケート(図3-2)の設問16である「最も使い慣れたプログラミング言語」の回答結果は、CとC++が7名、BASICが2名、PHPが1名であった。この結果からも、本プロジェクトで利用するプログラミング言語が、O大学で指導されているC、C++、BASICのいずれかから選択できるようにすることが妥当であったと判断できる。

本プロジェクトのPOIでは、コンピュータで対象物を制御する上で必要なプログラムの作成や、制御対象物を構成する電子回路の設計と製作のベースになる知識を、参加者に定着させることを目標として授業を行なった(図3-3、図3-4)。

また、POIの最終日に参加者たちはチームに分かれ、POIで得た知識や経験を活用して、製作する制御対象物を考案し企画書を作成した後に、その内容を発表した。このPOI活動を行なうことで、PBLを実施する際に必要な基礎能力が向上する。なお、本プロジェクト全体のスケジュールを表3-8に示す。



図3-3 POI 授業風景(電子回路の製作)

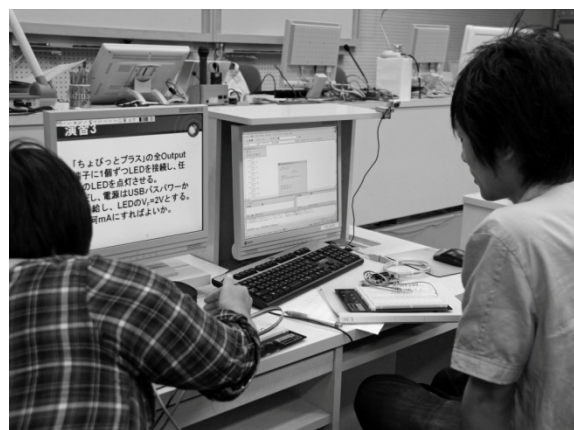


図3-4 POI 授業風景(電子回路の動作テスト)

表 3-8 本プロジェクトのスケジュール

時期	活動内容
(POI 活動) 2010 年(平成 22 年)8 月 7, 21, 28 日	<ul style="list-style-type: none"> ・プログラム作成の基本的な演習 ・電子回路の設計と製作の基本的な学習 ・制御対象物の考案 ・制御対象物の企画書の作成と発表
(PBL 活動) 2010 年(平成 22 年)8 月 29 日 ↓ 2010 年(平成 22 年)12 月 9 日	<ul style="list-style-type: none"> ・制御対象物を製作する計画の立案 ・制御対象物の製作とプログラムの開発 ・教員対象の説明会で作品の発表 ・Hidden Curriculum 企画委員会委員対象のプレゼンテーションで作品の発表

3.3 POI を導入した効果

POI を導入した効果を知るために、各 POI 実施日の開始直後と終了直前にアンケートを行なった。ただし、全回答者が 14 名であったので、統計的処理が行えるサンプル数に至っていないと判断し、カテゴリカルデータとして処理を行った。なお、表 3-9 から表 3-12 までの参加者番号は、表 3-1 と対応させている。

POI の 1 日目に行なった「ちょびっとプラス」を動作させるプログラム作成の基本的な演習の効果を、1 日目と 2 日目のプレアンケートおよび 3 日目のポストアンケートを比較する方法で検証した(表 3-9)。表中の数字は、4.得意な方、3.どちらかといえば得意な方、2.どちらかといえば苦手な方、1.苦手な方と回答したことを表す。また、POI 1 日目のプレアンケートを参考にして、POI 3 日目のポストアンケートとのレベル差を「プレ(1 日目)・ポストのレベル差」の項目として追加した。この項目の値は、+(プラス)はプロ

表 3-9 プログラミングについてのアンケート結果

参加者番号	1	2	4	5	12	13	14
POI 1 日目 プレアンケート	3	4	2	1	3	1	2
POI 2 日目 プレアンケート	3	4	1	3	3	2	3
POI 3 日目 ポストアンケート	3	4	1	2	3	1	3
プレ(1 日目)・ポストのレベル差	0	0	-1	+1	0	0	+1

プログラミング能力が向上していることで、0(ゼロ)はプログラミング能力に変化のないこと、-(マイナス)はプログラミング能力が低下していることを表している。なお、14名中7名の参加者データが有効であった。

この結果から、POIの1日目に行なったプログラミング演習では、受講前にプログラミングが苦手であると回答し、受講後はどちらかといえば得意であると回答している参加者番号5, 14がいるが、全体のプログラミング能力に関するレベル差については、ほとんど変化がなかったことが分かった。つまり、プログラミングに苦手意識を持っている参加者に対して、十分な支援が出来ていなかったことになる。

これは、POIでプログラミング言語を基礎から指導する時間がとれないことから、プログラミング言語の基礎的な説明や、与えた課題を理解させプログラムを最初から作成させる手法を採らず、事前に課題に沿って作成したサンプルプログラムを理解させ、課題の目標に達するように、そのサンプルプログラムを修正させる演習を多く体験させることで、徐々にプログラミング能力を向上させられると考え、ユニット方式の教授方法を採らなかったことが要因である。

このような方法を採った背景には、O大学の学生のほとんどが、BASICかCまたはC++のいずれかを、2年生までに学習しているので、個々の参加者が比較的使い慣れたプログラミング言語を選択出来るようにすれば、プログラミングについて苦手意識を持っている参加者であっても徐々にプログラミング能力を向上させられると考えていたことがある。

POIの3日目に、インタフェースの定格を超えた複数個のLEDの点滅方法をユニット方式で教えた。この授業日の前後に行ったアンケートの項目「インタフェースの定格を超えた複数個のLEDの点滅方法を説明できるか」に対する回答を表3-10に示す。表中の数字は、4.説明できる方、3.どちらかといえば説明できる方、2.どちらかといえば説明できない方、1.説明できない方と回答したことを表す。また、プレアンケートとポストアンケートとのレベル差を「プレ・ポストのレベル差」の項目として追加した。この項目の値は、+(プラス)は説明能力が向上していることで、0(ゼロ)は説明能力に変化のないことを表している。なお、14名中12名の参加者データが有効であった。

表 3-10 複数個の LED を点灯する方法の理解度のアンケート結果

参加者番号	1	3	4	5	6	8	9	10	11	12	13	14
POI 3 日目 プレ	2	3	1	2	4	1	1	1	1	4	1	1
POI 3 日目 ポスト	2	4	1	3	4	3	2	3	3	4	3	4
プレ・ポストのレベル差	0	+1	0	+1	0	+2	+1	+2	+2	0	+2	+3

この結果から、本ユニットの受講前は、説明できる参加者が 3 名で、説明できない参加者が 9 名であったが、受講後は、説明できる参加者が 6 名増え、説明できない参加者が 6 名減ったことが分かった。特に、受講前に説明できないと回答した参加者番号 5, 8, 10, 11, 13, 14 は、受講後は説明できると顕著な変化を示している。

POI の 3 日目に行ったアンケートの項目「電子回路の設計は得意か」に対する回答を表 3-11 に示す。表中の数字は、5.得意な方、4.どちらかといえば得意な方、3.どちらかといえば苦手な方、2.苦手な方、1.設計をしたことがないと回答したことを表す。また、プレアンケートとポストアンケートとのレベル差を「プレ・ポストのレベル差」の項目として追加した。この項目の値は、+ (プラス) は電子回路の設計能力が向上していることで、0 (ゼロ) は電子回路の設計能力に変化のないこと、- (マイナス) は電子回路の設計能力が低下したことを表している。なお、参加者 14 名中 12 名のデータが有効であった。

この結果から、受講前は、電子回路の設計が得意な参加者が 2 名、得意ではない参加者が 6 名で、設計をしたことがない参加者が 4 名であったが、受講後は、得意な参加者が 4 名増えたことが分かった。特に、受講前に得意ではないと回答した参加者番号 9, 10 と、設計をしたことがないと回答した参加者番号 11, 12 は、受講後は得意と顕著な変化を示している。

表 3-11 電子回路の設計についてのアンケート結果

参加者番号	1	3	4	5	6	8	9	10	11	12	13	14
POI 3 日目 プレ	2	4	2	3	4	2	2	3	1	1	1	1
POI 3 日目 ポスト	2	5	2	2	5	2	5	5	4	5	2	2
プレ・ポストのレベル差	0	+1	0	-1	+1	0	+3	+2	+3	+4	+1	+1

表 3-12 電子回路の製作についてのアンケート結果

参加者番号	1	3	4	5	6	8	9	10	11	12	13	14
POI 3 日目 プレ	2	4	2	3	5	2	3	4	2	2	1	1
POI 3 日目 ポスト	2	5	2	2	5	2	5	5	4	5	2	2
プレ・ポストのレベル差	0	+1	0	-1	0	0	+2	+1	+2	+3	+1	+1

POIの3日目に行ったアンケートの項目「電子回路の製作は得意か」に対する回答を表3-12に示す。表中の数字は、5.得意な方、4.どちらかといえば得意な方、3.どちらかといえば苦手な方、2.苦手な方、1.製作をしたことがないと回答したことを表す。また、プレアンケートとポストアンケートとのレベル差を「プレ・ポストのレベル差」の項目として追加した。この項目の値は、+（プラス）は電子回路の製作能力が向上していることで、0（ゼロ）は電子回路の製作能力に変化のないこと、-（マイナス）は電子回路の製作能力が低下したことを表している。なお、参加者14名中12名のデータが有効であった。

この結果から、受講前は、電子回路の製作が得意な参加者が3名、得意ではない参加者が7名で、製作をしたことがない参加者が2名であったが、受講後は、得意な参加者が3名増え、得意ではない参加者が1名減ったことが分かった。特に、受講前に得意ではないと回答した参加者番号9, 11, 12は、受講後は得意と顕著な変化を示している。

プログラム作成能力を向上させる教授方法に再考の余地を残すが、ユニット方式を取り入れた「すだち方式」によるPOIについては、インタフェースの定格を超えた複数個のLEDの点灯方法の説明(表3-10)、電子回路の設計(表3-11)及び電子回路の製作(表3-12)の結果から、一定の効果が得られたものと判断している。

また、参加者がPBLの開始時から到達目標までの計画を立案し遂行する際や、到達目標を成果報告の発表等で他者に伝える際に必要な基礎能力をPOIによって向上したことで、製作する制御対象物の企画書を全チームが作成することができ、POI最終日の発表に至る原動力になったと考えている。

3.4 PBLの実践

PBLでは、制御対象物を完成するまでの計画を、POIで作成した企画書を元に立案させてから、製作に取りかかるようにした^[3-8]。

PBL実施の際は、参加者の主体性を尊重し、教員の介在を極力少なくすることを重視して、特定の教室では行なわなかった。参加者らはPBL活動を実施しやすい場所として、所属する研究室、クラブの部室、課外活動としても作りが出来る施設である「自由工房」等を選んだ。

参加者のやる気を継続させるために、製作している制御対象物の進捗状況を報告させる機会を設ける必要があると考え、定期的に集合させることにした。この報告会で、参加者は問題点を整理する機会を得ることや、他チームの発表を聞き、自身のチームと他チームの進捗状況とを比較することによって、製作意欲を喚起できる。さらに、チームで改善できない問題が発生した場合に、教員や他の参加者からの助言を受けることも期待できる。これらによって、「プログラムが考えたような動作をしないし、改善方法が分からない。」や「バーサライタを回転させると、大きく振動する問題が発生している。」等の製作中に直面する諸問題に対して、他者の協力を得ながら、参加者が主体的に取り組むことができた。

2010年(平成22年)12月9日には、製作した総数5点の制御対象物を展示し、主にO大学の工学系と情報系の教員対象に説明会を行なった。教員ら38名は、制御対象物を全て見終わった後、評価し投票した。投票は、3項目の設問に対して、各々1作品を選出する方式で行なった(表3-13)。なお、表中の数は得票数を表すが、「技術的に優れている作品はどれか」と「アイデアが優れている作品はどれか」の項目に関しては、3名が無回答であった。

表3-13 展示説明会後の投票結果

質問項目	作品1	作品2	作品3	作品4	作品5	計
技術的に優れている作品はどれか	14	12	3	2	4	35
アイデアが優れている作品はどれか	10	11	7	4	3	35
一番気に入った作品はどれか	14	7	9	6	2	38

(数値は得票数)

その後、Hidden Curriculum 企画委員会委員を対象にした「学生によるプレゼンテーション」が行なわれ、教員の投票結果とプレゼンテーション内容を加味した審査を経て、参加 5 チームは授賞式に臨んだ。

その結果、作品 1 の「E³ ～8×8×8 個の LED を用いた立体ディスプレイ～」(図 3-5)が Hidden Curriculum 大賞を受賞した。この作品は、単なる平面的な電光掲示板ではなく、512 個の LED を立体的に配置する立体ディスプレイにしたところに特徴がある。また、作品 2 の「回転型バーサイライタ」(図 3-6)が学長特別賞と友電会賞(友電会は同窓会の名称で、友電会賞は友電会

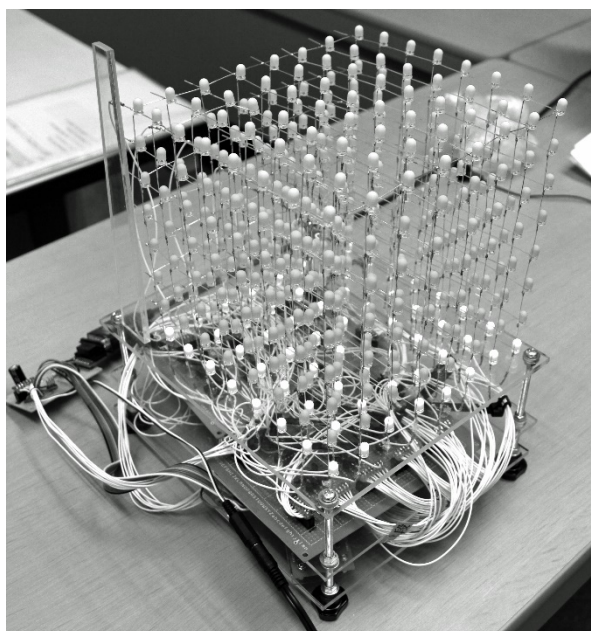


図 3-5 8×8×8 個の LED を用いた立体ディスプレイ
(作品 1)

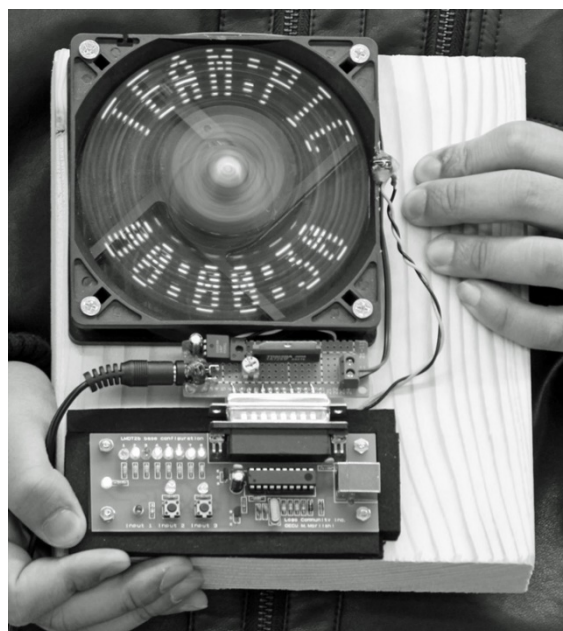


図 3-6 回転型バーサイライタ
(作品 2)

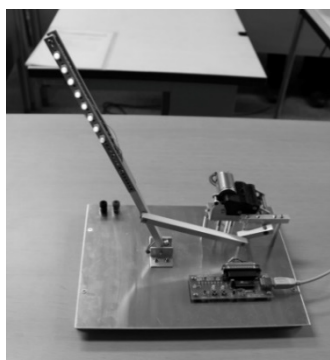


図 3-7 バーサイライタ
(作品 3)

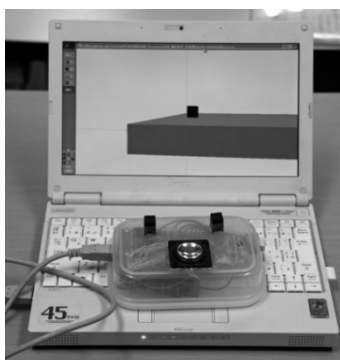


図 3-8 ちょびつとダッシュ
(作品 4)

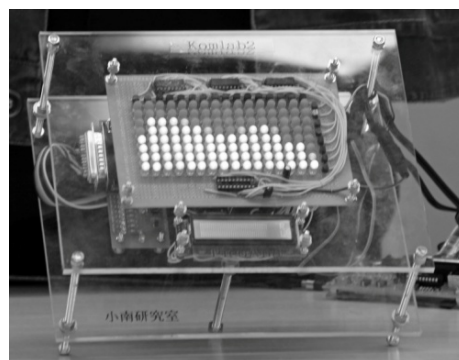


図 3-9 PicoAna
(作品 5)

の団体賞)を受賞した。一般的なバーサイタは、棒状に配置した複数個の光を点滅させ、それを手で振ることによって、目の残像効果を利用し、文字や絵が見られるようにしている。この作品は、バーサイタをモータで回転させ、時刻等の文字表示を固定して見せているところに特徴がある。

他には、棒状に配置した 8 個の LED を、モータを用いてメトロノームのように左右に振り、表示文字作成支援ソフトで作成した 4 文字までを LED の点滅によって表示可能にした作品 3 の「バーサイタ」(図 3-7)が技術賞を受賞した。また、「ちょびっとプラス」をインタフェースとして利用せず、ゲームを楽しむためのコントローラにした作品 4 の「ちょびっとダッシュ」(図 3-8)がアイデア賞を受賞した。さらに、マイク等から入力した音声信号を、フーリエ変換などで 16 個の周波数成分に分解し、16×8 個の LED を用いて表示する作品 5 の「PicoAna ～スペクトルアナライザ & ボリュームメータ～」(図 3-9)が企画委員会奨励賞を受賞した。

3.5 PBL 活動の成果

PBL 活動の例として、学長特別賞と友電会賞を受賞した「回転型バーサイタ」と、Hidden Curriculum 大賞受賞の「E³ ～8×8×8 個の LED を用いた立体ディスプレイ～」を製作したチームを取り上げる。

この 2 チームが、POI の最終日に発表した企画書の内容(以下、企画書と記述する)と、進捗状況報告会で行なった口答調査(以下、口答調査と記述する)で明らかになった到達目標までの経緯をまとめた内容から、PBL を行なう上で学生に必要な基礎能力について考察する。

回転型バーサイタを製作したチームの企画書には、製作物の概要を「目の残像効果を利用したバーサイタをモータで回転させる。表示は、実用性を重視して下部を時刻に、上部をチーム名にする」としていた。また、使用する主な電子部品を、PIC16F88(ワンチップマイコン)を 1 個、TA7291P(DC モータ用フルブリッジドライバ IC)を 1 個、セラミック発振子を 1 個、LED を 10 個、フォトトランジスタを 1 個、コンピュータ用冷却ファンを 1 個、D-SUB コネクタを 1 個等と記述していた。

この企画書の内容から、このチームは PBL 活動を開始する時には、既に作

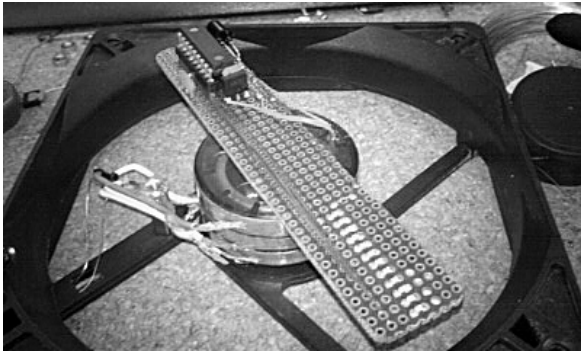


図3-10 回転型バーサライタの試作1号機

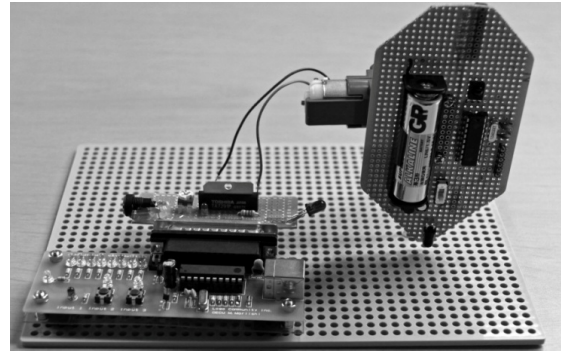


図3-11 回転型バーサライタの試作2号機

成する制御対象物の完成イメージを，正確に把握していたことが分かる．

口答調査では，リーダーが高校生の時に，一般的なバーサライタを製作したが，全く文字に見えず断念した経験があり，この経験から，一般的なバーサライタの製作に再挑戦し，完成後にコンピュータ用冷却ファンを改造して，バーサライタを回転させる計画にしたことが分かった．

この計画に従って，一般的なバーサライタを完成した後に，試作1号機(図3-10)を製作し，バーサライタを，コンピュータ用冷却ファンを改造して回転させた．しかし，バーサライタに外部から給電するブラシ部分の接触が不安定であったので，時々表示しない問題が発生した．

試作1号機の経験から，回転するバーサライタに動作電源として乾電池を実装する方法に変更し，模型用のモータで回転させる試作2号機を作成した(図3-11)．

試作2号機は，回転部分の重量バランスが悪く，しかも，モータの取り付け強度が不十分であったので，回転させると大きく振動する問題が発生した．

これら2台を試作した知見から，回転部分の重量バランスに配慮し，回転するバーサライタに，動作電源として乾電池より軽量のボタン電池を実装する方法に変更した．また，駆動部分は，コンピュータ用冷却ファンを改造する方法を採用し，図3-6で示した回転型バーサライタを目標通り完成させた．

このチームは企画書を元に，ある程度の計画を立案し，試行錯誤を行ないながら段階的に修正を加え，目標に到達する方法を採っている．結果，PBLの到達目標である回転型バーサライタを製作し，教員対象の説明会とHidden Curriculum 企画委員会委員対象のプレゼンテーションを成功させているの

で、PBLを行なう上で学生に必要な基礎能力が向上したと言える。

一方の「E³ ～8×8×8個のLEDを用いた立体ディスプレイ～」(以下、完成品と記述する)を製作したチームの企画書には、製作物の概要を「LEDを立体的に組み立て、シフトレジスタやSPI通信を用いて明滅させる。8×8×8個のLEDをキューブ状に配置し、3次元的な波を打たせるなど視覚的に楽しめるようにする」としていた。また、使用する主な電子部品を、AVR(ワンチップマイコン)を1個、シフトレジスタを8個、LEDを512個、抵抗を512個、コンデンサを3個等と記述していた。

この企画書の内容から、このチームはPBL活動を開始する時には、既に作成する制御対象物の完成イメージを、正確に把握していたことが分かる。

口答調査では、2010年(平成22年)11月末までに、完成品を完成させるために、製作工程の計画と作業分担を明確にすることから始めたことが分かった。しかし、512個のLEDを明滅できる完成品を、製作するためにかかる時間や工数を推測することは経験不足から困難であったので、4×4×4個(64個)のLEDを用いた立体ディスプレイ(以下、試作品と記述する)を試作し、そこで得られた知見から製作工程の計画と作業分担を作成することにしていた。

この試作品の製作は、製作工程の計画と作業分担を決定するだけではなく、コンピュータからの制御命令をできるだけ少なくすることも目標にしている。完成品は、512個のLEDを明滅するスイッチ回路が必要になるが、この512個のスイッチ回路を直接制御する設計は現実的ではない。試作品の考察から、72個のスイッチ回路で512個のLEDを明滅できることが分かった。さらに、コンピュータからの制御命令や明滅処理データの同期を、3本の通信線を用いて制御することで、1個から512個までのLEDを自由に明滅できることが分かった。

完成品の製作で苦心した点は、LEDの明滅をどの方向からでも確認できるように、LEDの端子2本を半田でつなぎ合わせて、きれいな立方体に見えるようにすることである。しかし、複数のメンバーが半田付けを行った結果ずれが生じ、きれいな立方体にならなかった。このような経験から、各LEDを正確な位置に半田付けが出来るようにするための治具を製作したこ

とと、作業方法のマニュアルを作成したことで、複数のメンバーが半田付けを行った場合でも、正確な位置に半田付けが出来るようになった。

このチームは企画書を元にしながらも、最初から完成品の設計と製作に着手するのではなく、完成品の簡易版である試作品を製作し、そこで得られた知見から、目標までの計画を立案している。また、複数のメンバーが作業を行うことで、LEDの半田付けが正確な位置に出来ないことが判明した時点で、治具の製作や作業方法のマニュアルを作成し、作業の改善を行なっている。これらの経緯を経て、PBLの到達目標である完成品を製作し、教員対象の説明会とHidden Curriculum企画委員会委員対象のプレゼンテーションを成功させているので、PBLを行なう上で学生に必要な基礎能力が向上したと言える。

学生らは、同年11月末の1週間は徹夜で製作と最終調整を行なったが、「この時間がとても楽しかった」と発言していたのが印象的であった。これは、教員の介在を極力少なくし、学生自身が設定した目標を自ら遂行するPBL方式の成果であり、学生自身が学ぶ楽しさを実感した活動であったことを意味している。

第3章の参考文献

[3-1]文部科学省：“高等学校学習指導要領(平成21年3月)”，東山書房
(2009)

[3-2]総務省 統計局：“高等学校(全日制・定時制)の学科別学校数及び生徒数(本科)”，<http://www.e-stat.go.jp/SG1/estat/List.do?bid=000001049724&cycode=0>

[3-3]都倉信樹：“創成科目の考え方と実施案”，工学教育，工学・工業教育研究講演会講演論文集 平成12年度，pp.275-278，(2000)

[3-4]武田邦彦：“工学系における創成教育の理論”，工学教育，53-1，pp.27-34，(2005)

[3-5]渡邊信一，高木淳二，長谷川 光司，入江 晃亘，杉山 均，横田 和隆：“学科横断的 PBL 形式科目「創成工学実践」における授業評価”，工学教育，59-2，pp.47-51，(2011)

- [3-6]菅井 勝雄：“コンピュータ教育のための授業モデルの理論的研究 - パラダイム論の観点から -”，日本教育情報学会，教育情報研究 日本教育情報学会学会誌 第3巻 第1号，pp.16-22, (1987)
- [3-7]森石 峰一，宇治 典貞，横山 宏，魚井 宏高，“PBLの成功を目的とした「すだち方式」による準備授業の設計と実践”，日本教育情報学会，第28巻，第3号，2013年3月，pp.15-22, (2013)
- [3-8]森石 峰一，都倉 信樹，横山 宏，魚井 宏高，“基礎指導を組み込んだPBL方式でのもの作り教育の試み”，日本工学教育協会，「工学教育」11月号，pp.22-28, (2013)

第4章 結論

我が国の計測・制御教育を発展させるために、中学校で2012年度(平成24年度)から、必修項目として完全実施されたばかりの「プログラムによる計測・制御」を技術担当教員が指導できるように、指導計画と指導目的を達成するために必要な実習用教材の提案を行なった。その中で、インタフェースと計測・制御システムを動作させるソフトウェアの開発について述べた。また、工学系の大学における計測・制御教育については、学生が主体的に学びやすい環境を整備することを目標にした授業設計と実践について述べた。これらの有用性については、後述の結果を得た。

筆者は、教職科目を指導する立場から、中学校の計測・制御教育で活用する実習用教材を開発しなければならないと考え、インタフェースとシステム操作ソフトウェアの開発を行なった。インタフェースは、物理的に独立し可視化させ、計測・制御システムを構成する各要素間の情報を伝達するための役割を担っていると、容易に理解できるようにすることを重視して開発を行なった。また、システム操作ソフトウェアは、中学校に導入実績のある **Logo Writer** をベースにし、生徒が理解できるように配慮した計測・制御実習用の追加命令を、予めプロシージャで作成した。

この実習用教材の有用性については、開発したインタフェースを使用することで、生徒にインタフェースの機能を容易に理解させられることが、2.5節で述べた評価結果から分かった。また、開発したシステム操作ソフトウェアの閉ループ制御のプログラムを作成する環境については、再考する余地を残しているが、開ループ制御のプログラムを作成する環境については、有用性が認められた。

したがって、必修項目になった「プログラムによる計測・制御」の指導経験が乏しい技術担当教員であっても、本研究で提案した指導計画と、開発した実習用教材を用いることで、不安や戸惑いを最小限にして指導を行なうことができると考えている。

筆者は、教職科目を指導する教員であるだけでなく、工学部の一員として、大学で実施されている計測・制御教育にも関心を持っている。

このような立場の筆者であるが、O大学の **Hidden Curriculum** 企画委員会

が企画し、「コンピュータで制御する対象物を、学生自身が考案し製作すること」を目標にした「光・動き出す・未来」プロジェクトを担当することで、大学生を対象とした計測・制御教育を行なう機会を得ることができた。

「光・動き出す・未来」プロジェクトの目標を達成させるためには、参加する学生が自由な発想で製作ができるような環境作りが重要であることから、PBLを活用することにした。

また、PBLの効果を着実に高めるために、学生に求める基礎能力である①PBLの到達目標までの計画を立案できること、②PBLの開始から学生自身が設定した到達目標までの構想を、ある程度段階的に考えられる知識、意識（価値観や動機付け）、スキル（技術）等を身に付けていること、③PBLの到達目標を、成果物の製作や成果報告の発表等の具体的な方法で、他者に伝える能力を向上させる授業であるPOIを、PBLを実施する前に行なうべきであると考え、理論づけを行なった。そして、POIからPBLへと系統的に接続していく指導方法を「すだち方式」と名づけた。

そこで、「光・動き出す・未来」プロジェクトを、POIからPBLへとつなげる「すだち方式」の指導方法で進めることにした。

前述の3つの基礎能力に差のある学生が参加した「光・動き出す・未来」プロジェクトの実施であったが、全チームがプロジェクトの目標であるコンピュータで制御する対象物を考案し製作することができた。また、作品の発表は、教員を対象にした説明会や、Hidden Curriculum 企画委員会委員を対象にしたプレゼンテーションの場で行われた。これは、POIの最終日に作成した企画書において、全チームが製作する制御対象物の完成イメージを把握していたことや、学生が主体的にPBLの目標に到達するまでの計画を作成し、その計画に従って遂行することができたから可能になったことである。したがって、学生は、POIで学習した3つの基礎能力を、PBLで主体的に活用できたと言える。また、POIについては、3.3節で述べたアンケート結果から学生の理解度を向上させることについての有用性が認められているので、POIをPBLの実施前に行ない、POIからPBLへと系統的に接続していく「すだち方式」の指導方法も有用性が認められたことになる。

4.1 今後の課題

開発した技術・家庭科用実習教材については、2013年(平成25年)12月より、京都府八幡市立男山東中学校の授業で利用してもらい、その有用性について検証する予定である。

授業を受ける生徒は、中学3年生(30人学級)の3クラスである。また、授業時数は、5単位時間になる予定である。

この授業の実施に向けて、男山東中学校の技術担当教員と、次のような打ち合わせを進めている。

- ①5単位時間の指導計画を作成する。
- ②プログラムを作成する際に使用する計測・制御実習用の追加命令を、中学生が直感的に理解できるような名前に再検討する。
- ③システム操作ソフトウェアのマニュアルを、中学生でも理解できるように改訂する。
- ④インタフェースやシステム操作ソフトウェア以外の教材を検討する。

この授業で得られた結果を分析し、さらなる改善を行ないたいと考えている。

「すだち方式」については、本研究では、工学系の大学での計測・制御教育をPOIからPBLへと系統的に接続していく「すだち方式」の指導方法で実施したが、この「すだち方式」を、工学系の大学でのもの作り教育全般に広げるための研究を進めたいと考えている。また、工学系の大学で行なう教育に限定せず様々な教育内容で、この「すだち方式」を活用できるように改善したいとも考えている。

さらに、ユニット方式を取り入れた「すだち方式」によるPOIの効果を高めるための研究も進めたいと考えている。

謝辞

本研究を進めていく上で、本当に多くの方々のお世話になり、様々なご指導、ご助言をいただきました。

特に、本研究の機会を賜り、研究を進める上で多大なるご指導をいただいた大阪電気通信大学大学院 総合情報学研究科教授の魚井宏高先生に心より感謝申し上げます。

本研究を進めるに当たり、多くのご指導、アドバイスをいただいた大阪電気通信大学大学院 総合情報学研究科教授の登尾啓史先生に心より感謝申し上げます。

本研究を進めるに当たり、多くのご指導、アドバイスをいただいた大阪電気通信大学大学院 総合情報学研究科教授の上善恒雄先生に心より感謝申し上げます。

また、大阪電気通信大学 総合情報学部の横山宏准教授には多くのご助言とご助力をいただきましたことを心より感謝申し上げます。

さらに、本研究に賛同して、**Logo Writer** のソースコードを提供していただいた(株)ロゴコミュニティの橋本 泰典さんに心より感謝申し上げます。

最後に、大阪電気通信大学名誉教授の石桁正士先生には、大変貴重な助言をいただきました。心より感謝申し上げます。

■ 付録 システム操作ソフトウェアのプログラム

(1) 生徒がプログラムを作成する環境 (ファイル名: プログラム作成用ファイル.LGW)

■ これより下で、手順を作成してください。

■ これより下は、指示があるまで変更しないでください。

■ 下記は、コマンドセンターにコピーして利用できる追加命令の例です。

動作チェック

光センサチェック

センサを使う

センサを使う手順のコピー

■ 下記は、各センサが ON になった時に、動作するプログラムを手順内に追加することができます。

追加したプログラムは、コマンドセンターで追加命令[センサを使う]を実行すると動作します。

※ 手順名の変更または削除をしないでください。

てじゅんは センサ 1 と 2 と 3 全てが OFF の動作

全出力を切れ

おわり

てじゅんは センサ 1 だけが ON の動作

出力 1 1

出力 2 0

出力 3 0

おわり

てじゅんは センサ 2 だけが ON の動作

出力 1 0

出力 2 1

出力 3 0

おわり

てじゅんは センサ 1 と 2 が ON の動作

出力 1 1

出力 2 1

出力 3 0

おわり

てじゅんは センサ 3 だけが ON の動作

出力 1 0

出力 2 0

出力 3 1

おわり

てじゅんは センサ1と3がONの動作

出力 1 1

出力 2 0

出力 3 1

おわり

てじゅんは センサ2と3がONの動作

出力 1 0

出力 2 1

出力 3 1

おわり

てじゅんは センサ1と2と3全てがONの動作

出力 1 1

出力 2 1

出力 3 1

おわり

■ 下記の手順は変更しないでください。

■ 2012 Copyright Minekazu MORIISHI Osaka Electro-Communication University

to startup

gettools "LMDT_2_main_routine.LGW

lmdt_on

Set_J_icons

End

(2) システム操作ソフトウェア本体のプログラム (ファイル名 : LMDT_2_main_routine.LGW)

■ 2012 Copyright Minekazu MORIISHI Osaka Electro-Communication University

■ 下記は、コマンドセンターにコピーして利用できる追加命令の一部です。

lmdt_on
動作時間の調整
動作チェック
光センサチェック
出力 1 1 出力 2 1
8ビットでは 192
モータ 2 1 モータ 3 1
全出力を切れ

■ 追加命令[センサを使う]の実行時に、動作するプログラムを追加する手順内をコマンドセンターにコピーする。

```
to copy_sensor_procedure
  type char 13
  show "|■ 下記は、各センサが ON になった時に、動作するプログラムを手順内に追加することができます。|
  show "| 追加したプログラムは、コマンドセンターで追加命令[センサを使う]を実行すると動作します。|
  show "| ※ 手順名の変更または削除をしないでください。| type char 13
  show "|てじゅんは センサ 1 と 2 と 3 全てが OFF の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 1 だけが ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 2 だけが ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 1 と 2 が ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 3 だけが ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 1 と 3 が ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 2 と 3 が ON の動作| type char 13 show "おわり type char 13
  show "|てじゅんは センサ 1 と 2 と 3 全てが ON の動作| type char 13 show "おわり
end
```

■ Base configuration unit の追加命令を日本語で使用するための手順

■ 開ループ制御の追加命令

```
to 出力 :出力番号 :入り切り決定
  lmdt_out :出力番号 :入り切り決定
end

to 全出力を切れ
  lmdt_out_all_off
end

to 8ビットでは :hyouka
  lmdt_8bit :hyouka
end
```

```
to 8ビットでは :hyouka
  lmdt_8bit :hyouka
end
```

```
to 待つ時間は :秒数
  lmdt_motion_time :秒数
end
```

■ 動作時間の基本設定に関する追加命令

```
to 動作時間の単位 :jikan
  lmdt_time_unit :jikan
end
```

```
to 動作時間の調整
  lmdt_time_adjustment
end
```

■ 開ループ制御の追加命令 (Motor control unit 関係)

```
to モータ :モータ番号 :動作決定
  lmdt_motor :モータ番号 :動作決定
end
```

```
to 回転角度は :角度
  lmdt_turn :角度
end
```

■ Motor control unit で動作させる制御対象物の回転角度を決定する追加命令

```
to 右モータは :set_motor
  lmdt_right_motor_set :set_motor
end
```

```
to 左モータは :set_motor
  lmdt_left_motor_set :set_motor
end
```

```
to 一回転の測定
  lmdt_360-degree_roll
end
```

■ 閉ループ制御の追加命令

```
to センサを使う
  lmdt_sensor
end
```

```
to 計測
  lmdt_measurement
  make "計測結果 :result
end
```

■ 動作チェック用の追加命令

```
to 動作チェック
  lmdt_out_auto_run
end
```

```
to 光センサチェック
  lmdt_photo_check
end
```

■ 初期設定に関する追加命令

```
to 開始
  lmdt_on
end
```

```
to 終了
  lmdt_off
end
```

■ アイコン設定に関する追加命令

```
to 日本語アイコン
  Set_J_Icons
end
```

```
to 英語アイコン
  Set_E_Icons
end
```

■ アイコン設定に関する追加命令

```
to センサを使う手順のコピー
  copy_sensor_procedure
end
```

■ アイコンの設定

to Set_J_Icons

```
seticon 1 "出力 [ type "|出力 | ]  
seticon 2 "全切 [ type "全出力を切れ ]  
seticon 3 "待て [ type "|待つ時間は | ]  
seticon 4 "チェック [ type "光センサチェック ]  
seticon 5 "モータ [ type "|モータ | ]  
seticon 6 "角度 [ type "|回転角度は | ]  
seticon 7 "回転 [ type "一回転の測定 ]  
seticon 8 "ビット [ type "|8ビットでは | ]  
seticon 9 "センサ [ show "センサを使う ]  
seticon 10 "手順 [ show "|てじゅんは | type char 13 show "おわり ]
```

end

to Set_E_Icons

```
seticon 1 "out [ type "|lmdt_out | ]  
seticon 2 "ALof [ type "lmdt_out_all_off ]  
seticon 3 "wait [ type "|lmdt_motion_time | ]  
seticon 4 "Pchk [ type "lmdt_photo_check ]  
seticon 5 "|(M)| [ type "|lmdt_motor | ]  
seticon 6 "turn [ type "|lmdt_turn | ]  
seticon 7 "roll [ type "lmdt_360-degree_roll ]  
seticon 8 "8bit [ type "|lmdt_8bit | ]  
seticon 9 "snsr [ show "lmdt_sensor ]  
seticon 10 "手順 [ show "|to | type char 13 show "end ]
```

end

■ Base configuration unit を初期化するための追加命令

to lmdt_on

```
lmdt_init  
lmdt_wait 0  
initialization_variable  
make "motion_unit 60  
make "time_adjustment 0.64  
make "angle_unit 1  
make "MMtemp_right_motor 2  
make "MMtemp_left_motor 1  
set_input_check
```

end

```

to initialization_variable
  make "output_check 0
  make "motor_check 0
  make "MM_data 0
  make "MM_data_1 0 make "MMtemp_data_1 0
  make "MM_data_2 0 make "MMtemp_data_2 0
  make "MM_data_3 0 make "MMtemp_data_3 0
  make "MM_data_4 0 make "MMtemp_data_4 0
  make "MM_data_5 0 make "MMtemp_data_5 0
  make "MM_data_6 0 make "MMtemp_data_6 0
  make "MM_data_7 0 make "MMtemp_data_7 0
  make "MM_data_8 0 make "MMtemp_data_8 0
  make "moving 0
  make "MM_bitwise_data 0
  make "result 0
end

```

```

to set_input_check
  hideturtle
  loadicon 19 "operation_check.ICO
  tell [ 1 2 3 ] hideturtle penup setsh 19 setcolor 0
  tell 1 setpos [ -100 0 ]
  tell 2 setpos [ 0 0 ]
  tell 3 setpos [ 100 0 ]
  tell [ 1 2 3 ] showturtle
end

```

■ Base configuration unit の操作を終了し、安全に取り外すことができる状態にするための追加命令

```

to lmdt_off
  initialization_variable
  lmdt_exit
end

```

■ Base configuration unit の Output1 から Output8 までを順番に 0.5 秒間 ON にするための追加命令 (主に Output の動作チェックに使用)

```

to lmdt_out_auto_run
  make "output_no 0
  repeat 8 [ lmdt_poke power 2 :output_no wait 3 make "output_no :output_no + 1 ]
  wait 3
  lmdt_out_all_off
end

```

■ Base configuration unit の Input1 に接続している光センサの ON/OFF チェックのための追加命令

```
to lmdt_photo_check
  tell 1
  in_check
  if :result = 1 [ setcolor 2 output_1 1 ]
  if :result = 0 [ setcolor 0 lmdt_out_all_off ]
  if key? [ setcolor 0 lmdt_out_all_off show "| 光センサのチェックを終了!! | stop ]
  lmdt_photo_check
end
```

■ 開ループ制御の追加命令

```
to lmdt_out :out_no :on_off
  if :output_check = 0 [ initialization_variable make "output_check 1 ]
  if :out_no = 1 [ output_1 :on_off stop ]
  if :out_no = 2 [ output_2 :on_off stop ]
  if :out_no = 3 [ output_3 :on_off stop ]
  if :out_no = 4 [ output_4 :on_off stop ]
  if :out_no = 5 [ output_5 :on_off stop ]
  if :out_no = 6 [ output_6 :on_off stop ]
  if :out_no = 7 [ output_7 :on_off stop ]
  if :out_no = 8 [ output_8 :on_off stop ]
  output_coverage_check :out_no
end
```

```
to output_1 :on_off
  output_error_check 1 :on_off
  if and :on_off = 1 :MMtemp_data_1 = 0 [ make "MM_data_1 1 make "MMtemp_data_1 1 output_control ]
  if and :on_off = 0 :MMtemp_data_1 = 1 [ make "MM_data 1 make "MMtemp_data_1 0 stop_processing ]
  make "MM_data_1 0
end
```

```
to output_2 :on_off
  output_error_check 2 :on_off
  if and :on_off = 1 :MMtemp_data_2 = 0 [ make "MM_data_2 2 make "MMtemp_data_2 1 output_control ]
  if and :on_off = 0 :MMtemp_data_2 = 1 [ make "MM_data 2 make "MMtemp_data_2 0 stop_processing ]
  make "MM_data_2 0
end
```

```
to output_3 :on_off
  output_error_check 3 :on_off
  if and :on_off = 1 :MMtemp_data_3 = 0 [ make "MM_data_3 4 make "MMtemp_data_3 1 output_control ]
  if and :on_off = 0 :MMtemp_data_3 = 1 [ make "MM_data 4 make "MMtemp_data_3 0 stop_processing ]
  make "MM_data_3 0
end
```

```

to output_4 :on_off
  output_error_check 4 :on_off
  if and :on_off = 1 :MMtemp_data_4 = 0 [ make "MM_data_4 8 make "MMtemp_data_4 1 output_control ]
  if and :on_off = 0 :MMtemp_data_4 = 1 [ make "MM_data 8 make "MMtemp_data_4 0 stop_processing ]
  make "MM_data_4 0
end

to output_5 :on_off
  output_error_check 5 :on_off
  if and :on_off = 1 :MMtemp_data_5 = 0 [ make "MM_data_5 16 make "MMtemp_data_5 1 output_control ]
  if and :on_off = 0 :MMtemp_data_5 = 1 [ make "MM_data 16 make "MMtemp_data_5 0 stop_processing ]
  make "MM_data_5 0
end

to output_6 :on_off
  output_error_check 6 :on_off
  if and :on_off = 1 :MMtemp_data_6 = 0 [ make "MM_data_6 32 make "MMtemp_data_6 1 output_control ]
  if and :on_off = 0 :MMtemp_data_6 = 1 [ make "MM_data 32 make "MMtemp_data_6 0 stop_processing ]
  make "MM_data_6 0
end

to output_7 :on_off
  output_error_check 7 :on_off
  if and :on_off = 1 :MMtemp_data_7 = 0 [ make "MM_data_7 64 make "MMtemp_data_7 1 output_control ]
  if and :on_off = 0 :MMtemp_data_7 = 1 [ make "MM_data 64 make "MMtemp_data_7 0 stop_processing ]
  make "MM_data_7 0
end

to output_8 :on_off
  output_error_check 8 :on_off
  if and :on_off = 1 :MMtemp_data_8 = 0 [ make "MM_data_8 128 make "MMtemp_data_8 1 output_control ]
  if and :on_off = 0 :MMtemp_data_8 = 1 [ make "MM_data 128 make "MMtemp_data_8 0 stop_processing ]
  make "MM_data_8 0
end

to output_control
  make "MM_data :MM_data_1 + :MM_data_2 + :MM_data_3 + :MM_data_4 + :MM_data_5 + :MM_data_6
+ :MM_data_7 + :MM_data_8
  make "MM_bitwise_data :MM_bitwise_data + :MM_data
  lmdt_poke :MM_bitwise_data
end

to stop_processing
  make "MM_bitwise_data :MM_bitwise_data - :MM_data
  make "MM_data 0
  lmdt_poke :MM_bitwise_data
end

```



```

to output_error_check :output_error_no :on_off
  if ( or :on_off = 0 :on_off = 1 ) [ stop ]
    error_msg :on_off
    show ( word "|出力 |:output_error_no|" の動作は、ON であれば 1 を、OFF であれば 0 のどちらかを入力
    してください。 | )
    stopall
  end
end

```

```

to output_coverage_check :no
  if ( or :no = 1 :no = 2 :no = 3 :no = 4 :no = 5 :no = 6 :no = 7 :no = 8 ) [ stop ]
    error_msg :no
    show "|出力番号の設定には、1 ~ 8 の整数を入力してください。 |
    stopall
  end
end

```

■ 全ての Output を OFF にするための追加命令

```

to lmdt_out_all_off
  lmdt_poke 0
  initialization_variable
end

```

■ アクチュエータを 8bit で制御するための追加命令

```

to lmdt_8bit :hyouka
  if ( or :hyouka < 0 :hyouka > 255 ) [ error_msg :hyouka show "|0 ~ 255 の整数を入力してください。 | stopall ]
  initialization_variable
  make "output_check 1
  make "MMtemp_control_no 7
  make "MM_bitwise_data :hyouka
  lmdt_poke :MM_bitwise_data
  repeat 8 [ make "temp_target power 2 :MMtemp_control_no
    make "hyouka :hyouka - :temp_target
    ifelse :hyouka < 0 [ make "hyouka :hyouka + :temp_target ] [ output_seting ]
    make "MMtemp_control_no :MMtemp_control_no - 1
  ]
end

```

```

to output_seting
  if :MMtemp_control_no = 7 [ make "MMtemp_data_8 1 stop ]
  if :MMtemp_control_no = 6 [ make "MMtemp_data_7 1 stop ]
  if :MMtemp_control_no = 5 [ make "MMtemp_data_6 1 stop ]
  if :MMtemp_control_no = 4 [ make "MMtemp_data_5 1 stop ]
  if :MMtemp_control_no = 3 [ make "MMtemp_data_4 1 stop ]
  if :MMtemp_control_no = 2 [ make "MMtemp_data_3 1 stop ]
  if :MMtemp_control_no = 1 [ make "MMtemp_data_2 1 stop ]
  if :MMtemp_control_no = 0 [ make "MMtemp_data_1 1 stop ]
end

```

■ アクチュエータの動作時間を決定する追加命令

```
to lmdt_motion_time :atai
  MMtemp_motion_time round ( :motion_unit * :time_adjustment * :atai )
end
```

```
to MMtemp_motion_time :moving
  repeat :moving [ wait 1 if key? [ all_process_stop ] ]
end
```

■ アクチュエータの動作時間の単位を決定する追加命令 ※初期値は 60 で約 1 秒に設定

```
to lmdt_time_unit :jikan
  if :motion_unit = :jikan [ play "FUE show ( word "| | :jikan "| は既に設定された値です。| ) stopall ]
  if ( or :jikan = 0 :jikan < 0 ) [ error_msg :jikan show "正の数を入力してください。 stopall ]
  play "FUE
  type ( word "| | :motion_unit "| だった値を | )
  make "motion_unit :jikan
  show word :motion_unit " | に変更しました。|
  show " | ※ 初期値は 60 で約 1 秒に設定しています。|
end
```

■ アクチュエータの動作時間の単位を調整する追加命令

```
to lmdt_time_adjustment
  make "time_adjustment 0
  play "FUE
  show " | 動作時間の調整を行いません。15 秒後にスペースキーをたたいてください。|
  lmdt_poke 255
  tell [ 1 2 3 ] setcolor 2
  lmdt_time_counter
  tell [ 1 2 3 ] setcolor 0
  lmdt_poke 0
  make "time_adjustment :time_adjustment / 900
  play "FUE
  show " | ■■ 動作時間の調整ができました。■■|
end
```

```
to lmdt_time_counter
  wait 1
  make "time_adjustment :time_adjustment + 1
  if key? [ stop ]
  lmdt_time_counter
end
```

■ 開ループ制御の追加命令 (Motor control unit 関係)

```
to lmdt_turn :angle
  MMtemp_motion_time round ( :angle_unit * :angle )
end

to lmdt_motor :motor_no :select
  if :motor_check = 0 [ initialization_variable make "motor_check 1 ]
  if :motor_no = 1 [ motor_1 :select stop ]
  if :motor_no = 2 [ motor_2 :select stop ]
  if :motor_no = 3 [ motor_3 :select stop ]
  if :motor_no = 4 [ motor_4 :select stop ]
  motor_coverage_check :motor_no
end

to motor_1 :select
  motor_error_check 1 :select
  if and :select = 1 :MMtemp_data_1 <> 1 [ sub_motor_1 1 stop ]
  if and :select = 0 :MMtemp_data_1 <> 0 [ make "MM_data :MMtemp_data_1 make "MMtemp_data_1 0
stop_processing stop ]
  if and :select = -1 :MMtemp_data_1 <> 2 [ sub_motor_1 2 stop ]
end

to sub_motor_1 :select
  make "MM_data_1 :select
  make "MM_bitwise_data :MM_bitwise_data - :MMtemp_data_1
  make "MMtemp_data_1 :select
  output_control
  make "MM_data_1 0
end

to motor_2 :select
  motor_error_check 2 :select
  if and :select = 1 :MMtemp_data_2 <> 4 [ sub_motor_2 4 stop ]
  if and :select = 0 :MMtemp_data_2 <> 0 [ make "MM_data :MMtemp_data_2 make "MMtemp_data_2 0
stop_processing stop ]
  if and :select = -1 :MMtemp_data_2 <> 8 [ sub_motor_2 8 stop ]
end

to sub_motor_2 :select
  make "MM_data_2 :select
  make "MM_bitwise_data :MM_bitwise_data - :MMtemp_data_2
  make "MMtemp_data_2 :select
  output_control
  make "MM_data_2 0
end
```

```

to motor_3 :select
  motor_error_check 3 :select
  if and :select = 1 :MMtemp_data_3 <> 16 [ sub_motor_3 16 stop ]
  if and :select = 0 :MMtemp_data_3 <> 0 [ make "MM_data :MMtemp_data_3 make "MMtemp_data_3 0
stop_processing stop ]
  if and :select = -1 :MMtemp_data_3 <> 32 [ sub_motor_3 32 stop ]
end

to sub_motor_3 :select
  make "MM_data_3 :select
  make "MM_bitwise_data :MM_bitwise_data - :MMtemp_data_3
  make "MMtemp_data_3 :select
  output_control
  make "MM_data_3 0
end

to motor_4 :select
  motor_error_check 4 :select
  if and :select = 1 :MMtemp_data_4 <> 1 [ sub_motor_4 64 stop ]
  if and :select = 0 :MMtemp_data_4 <> 0 [ make "MM_data :MMtemp_data_4 make "MMtemp_data_4 0
stop_processing stop ]
  if and :select = -1 :MMtemp_data_4 <> 2 [ sub_motor_4 128 stop ]
end

to sub_motor_4 :select
  make "MM_data_4 :select
  make "MM_bitwise_data :MM_bitwise_data - :MMtemp_data_4
  make "MMtemp_data_4 :select
  output_control
  make "MM_data_4 0
end

to motor_error_check :motor_error_no :select
  if ( or :select = 1 :select = 0 :select = -1 ) [ stop ]
  error_msg :select
  show ( word "|モータ | :motor_error_no | の設定は、正回転であれば 1 を、停止であれば 0 を、逆回転で
あれば -1 を入力してください。 | )
  stopall
end

```

■ Motor control unit で動作させる制御対象物の回転角度を決定する追加命令

```
to lmdt_right_motor_set :set_motor
  play "FUE
  if :MMtemp_right_motor = :set_motor [ show ( word " | :set_motor" | 番は既に右側のモータとして設定されて
います。 | ) stopall ]
  if :MMtemp_left_motor = :set_motor [ show ( word " | :set_motor" | 番は左側のモータとして設定されてい
ます。 | ) stopall ]
  motor_coverage_check :set_motor
  type ( word " | :MMtemp_right_motor" | 番のモータを右側に設定していましたが、 | )
  make "MMtemp_right_motor :set_motor
  show word :MMtemp_right_motor | 番のモータに変更しました。 |
end

to lmdt_left_motor_set :set_motor
  play "FUE
  if :MMtemp_left_motor = :set_motor [ show ( word " | :set_motor" | 番は既に左側のモータとして設定され
ています。 | ) stopall ]
  if :MMtemp_right_motor = :set_motor [ show ( word " | :set_motor" | 番は右側のモータとして設定されてい
ます。 | ) stopall ]
  motor_coverage_check :set_motor
  type ( word " | :MMtemp_left_motor" | 番のモータを左側に設定していましたが、 | )
  make "MMtemp_left_motor :set_motor
  show word :MMtemp_left_motor | 番のモータに変更しました。 |
end

to lmdt_360-degree_roll
  make "angle_unit 0
  play "FUE
  show " | 回転角度の決定を行いません。制御対象物が 360 度回転すればスペースキーをたたいてください。
|
  lmdt_motor :MMtemp_right_motor -1
  lmdt_motor :MMtemp_left_motor 1
  lmdt_angle_measurement
  lmdt_motor :MMtemp_right_motor 0
  lmdt_motor :MMtemp_left_motor 0
  make "angle_unit :angle_unit / 360
  play "FUE
  show " | ■■ 一回転の測定終了 ■■ |
end

to lmdt_angle_measurement
  wait 1
  make "angle_unit :angle_unit + 1
  if key? [ stop ]
  lmdt_angle_measurement
end
```

```

to motor_coverage_check :no
  if ( or :no = 1 :no = 2 :no = 3 :no = 4 ) [ stop ]
    error_msg :no
    show "|モータ番号の設定には、1 ~ 4 の整数を入力してください。|"
    stopall
  end
end

```

■ 開ループ制御共通の入力エラー表示

```

to error_msg :msg
  play "FUE"
  type ( word "| ■■ 入力エラー ■■ → 入力された |:msg| は設定できません。| )
end

```

■ 全ての作業を終了するための追加命令

```

to all_process_stop
  tell [ 1 2 3 ] setcolor 0
  lmdt_out_all_off
  show "| 作業の終了!!|"
  stopall
end

```

■ 閉ループ制御の追加命令

```

to lmdt_sensor
  sensor_process
  lmdt_sensor
end

```

```

to sensor_process
  if key? [ all_process_stop ]
    in_check
    make "MMtemp_result :result
    if :result = 0 [ tell [ 1 2 3 ] setcolor 0 suspend_in0 stop ]
    if :result = 1 [ tell 1 setcolor 2 tell [ 2 3 ] setcolor 0 suspend_in1 stop ]
    if :result = 2 [ tell 2 setcolor 2 tell [ 1 3 ] setcolor 0 suspend_in2 stop ]
    if :result = 3 [ tell [ 1 2 ] setcolor 2 tell 3 setcolor 0 suspend_in3 stop ]
    if :result = 4 [ tell 3 setcolor 2 tell [ 1 2 ] setcolor 0 suspend_in4 stop ]
    if :result = 5 [ tell [ 1 3 ] setcolor 2 tell 2 setcolor 0 suspend_in5 stop ]
    if :result = 6 [ tell [ 2 3 ] setcolor 2 tell 1 setcolor 0 suspend_in6 stop ]
    if :result = 7 [ tell [ 1 2 3 ] setcolor 2 suspend_in7 stop ]
  end
end

```

```

to in_check
  make "result lmdt_peek
end

```

```
to suspend_in0
  センサ 1 と 2 と 3 全てが OFF の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in0
end

to suspend_in1
  センサ 1 だけが ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in1
end

to suspend_in2
  センサ 2 だけが ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in2
end

to suspend_in3
  センサ 1 と 2 が ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in3
end

to suspend_in4
  センサ 3 だけが ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in4
end

to suspend_in5
  センサ 1 と 3 が ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in5
end

to suspend_in6
  センサ 2 と 3 が ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in6
end
```

```
to suspend_in7
  センサ 1 と 2 と 3 全てが ON の動作
  in_check
  if or :result <> :MMtemp_result key? [ stop ]
  suspend_in7
end
```

```
to lmdt_measurement
  in_check
  if :result = 0 [ tell [ 1 2 3 ] setcolor 0 ]
  if :result = 1 [ tell 1 setcolor 2 tell [ 2 3 ] setcolor 0 ]
  if :result = 2 [ tell 2 setcolor 2 tell [ 1 3 ] setcolor 0 ]
  if :result = 3 [ tell [ 1 2 ] setcolor 2 tell 3 setcolor 0 ]
  if :result = 4 [ tell 3 setcolor 2 tell [ 1 2 ] setcolor 0 ]
  if :result = 5 [ tell [ 1 3 ] setcolor 2 tell 2 setcolor 0 ]
  if :result = 6 [ tell [ 2 3 ] setcolor 2 tell 1 setcolor 0 ]
  if :result = 7 [ tell [ 1 2 3 ] setcolor 2 ]
end
```