

# LEGO Mindstorms を用いたマルチメディア教育の試み

中 嶋 正 夫

## Education of Multimedia with LEGO Mindstorms robot kit.

Masao Nakajima

### Abstract

This study reports on multimedia education with LEGO Mindstorms robot kit. In our department, it is difficult to build the multimedia application system for the lack of knowledge about programming. But using LEGO Mindstorms, students' willingness to study multimedia is increasing. At last students develop a simulation model for traffic safety at intersections.

Since LEGO Mindstorms use, students can practice multimedia and robotics easily.

### 1. はじめに

筆者は、城西国際大学メディア学部において、1、2年生は情報リテラシー教育を、3、4年生は「プロジェクト研究」においてマルチメディア応用システムの教育を主に担当している。

近年のコンピュータは機能の向上とともに、マルチメディアへの対応が高機能化され、誰でもマルチメディアコンテンツの制作が可能となった。このような状況下において、マルチメディア関連の専門教育では、単なるコンテンツ作成のノウハウに留まらずに、さまざまな分野への応用に対する適応力を養う必要がある。

たとえば、一昔前のマルチメディアというと、学習教材やコンピュータゲームなど、画像、映像、音響のデジタルデータとネットワークをソフトウェアによって統合したシステムが主であった。これらは、コンピュータシステムで完結でき、学生の手持ちのパソコンで学習できることから、教育の面からも効率がよい。しかし、学生は幼年期からコンピュータやゲーム機をはじめとした、ディスプレイやキーボード、マウス、ゲームコントローラといったインターフェースに慣れ親しんでいるが、新たなインターフェース機器や、既存のインターフェースの応用的な利用についての関心が弱い傾向があった。

また、長年マルチメディアシステム構築の教育に携わっていて、次のことが明らかになった。それは、プログラミングソフトのような、システムの開発環境を活用する上で、学生の基礎的なアルゴリズムの理解度が低く、システムの開発スキルが、独自のシステムを構築するレベルに達していないことである。

以上のようなことから、容易にアルゴリズムの学習ができ、応用力が身に付くような教育方法を試みた。それは、教育用のロボットのキットとして有名な、LEGO 社の Mindstorms を用いた教育である。Mindstorms はいわゆるレゴパーツを組み合わせることでモデルを作り、モーターや各種センサーを Brick といわれるコントロールユニットに接続して、動作を制御するロボットの教材である。

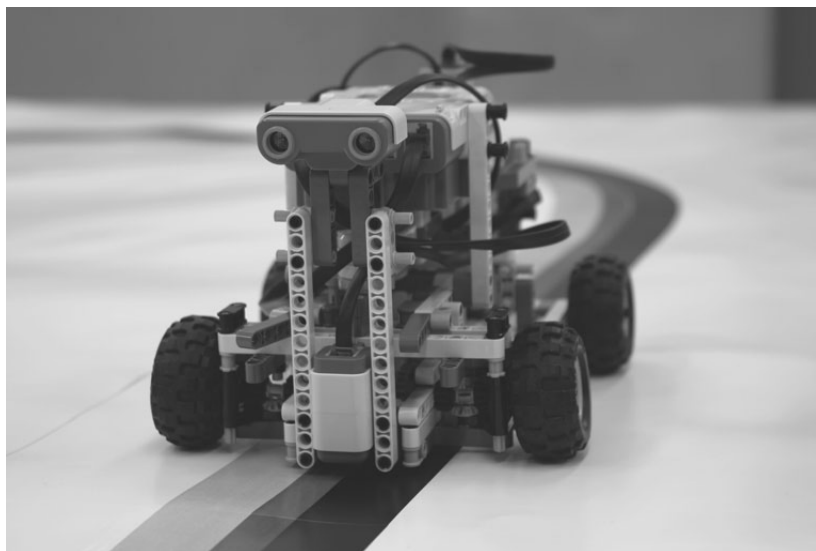


図 1 LEGO Mindstorms

本メディア学部では、実践的なものづくりを授業に取り入れているが、ハードウェアとソフトウェアの両面からのものづくりを体験できることは、教育上にメリットがあると判断した。ハードウェアといっても、基本は多くの学生が幼少時に体験した LEGO 遊びがベースであり、作成したロボットの動作を制御するソフトウェアは、ビジュアルプログラミング環境を備えたものが多くなっている。従って、理系指向でない学生でも容易に理解できるものと思われる。

ここでは、大学のマルチメディア系授業において、LEGO Mindstorms を活用した事例を報告する。まず、LEGO Mindstorms の特徴について述べる。次に、3 年次生後半から 4 年次生にかけての、LEGO Mindstorms を使った教育活動の概要を紹介する。そして、授業で学生が構築した「交差点において交通信号を遵守する車両のシミュレーションモデル」の詳細について述べる。最後に、この 1 年半での教育成果についての考察を報告する。

## 2. LEGO Mindstorms とは

### 2. 1 LEGO Mindstorms の成り立ち

LEGO Mindstorms は、1998 年に LEGO 社とマサチューセッツ工科大学による共同研究によって開発された教育用のロボット作成キットである。対象は小学生から大学生までカバーし、キットに付属のチュートリアルに従って、モデルを作成しサンプルプログラムを実行することで、だれでも簡単

にロボットの製作と動作のプログラミングが可能である。

1998 年当初は **Mindstorms RCX** が発売された。これは、コントローラに 8bit CPU が内蔵されている。各種センサーやモーターはレゴブロックの形状をしており、子供が遊ぶようなブロックを組み立てることで、ロボットを作成する。

2006 年に登場した 2 世代目は **Mindstorms NXT** と呼ばれ、コントローラは 32bit CPU を搭載し、プログラムやデータを保存する **RAM** の容量も拡大された。ロボットの構築に使うパーツは **LEGO** テクニクに使われているものと共通で、主にアーム系のパーツを多用している。

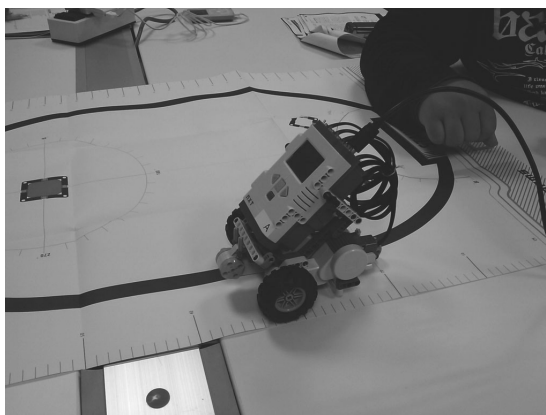


図 2 Mindstorms NXT

次世代機は **Mindstorms NXT 2.0** と呼ばれている。**NXT** と **NXT2.0** の違いは、走行系のパーツの変更や、一部のセンサーのデザインの変更、ソフトウェアのバージョンアップである。2009 年 11 月現在、日本語バージョンは販売されていない。

## 2. 2 LEGO Mindstorms NXT の特徴

今回の教育に用いた **LEGO Mindstorms** は **LEGO Mindstorms NXT** (以降 **NXT** と略す) である。これは、教育用のセットと、ホビー用のセットがある。今回はホビー用のセットを選択した。初代の **RCX** と同様に工具なしで組み立てられる **LEGO** パーツによって、さまざまな形状や動きを表現できる。これらのパーツ以外に、動力を発生するモーター、明るさ、音、タッチスイッチ、超音波による測距の各センサーが同梱されている。モーターは回転角度を測るセンサーを兼用している。これらのモーターやセンサーは、**Brick** と呼ばれるコントローラに接続して、使用する。**Brick** は、プログラムや画像、サウンドなどのデータを内部メモリに取り込むことができる。プログラムやデータは、パーソナルコンピュータを使って、**USB** や **Bluetooth** 経由で転送する。ただし、プログラムを実行するときにはスタンドアロンで動作させることができる。

動作を定義するプログラムの開発環境として、ホビー用セットには同梱のソフトウェアがある。これは、いわゆるビジュアルプログラミング環境で、入門者向けのチュートリアルまで内包しているので、このソフトウェア 1 つだけで、初心者にも学習できるようになっている。モーターの動きの制御

や各種センサーの値の読み込み、繰り返し処理、条件分岐などが、1つ1つ LEGO パーツのようにアイコン化され、それらを LEGO パーツの一種である仮想的なアームに接続するイメージで、プログラムを構築していく。

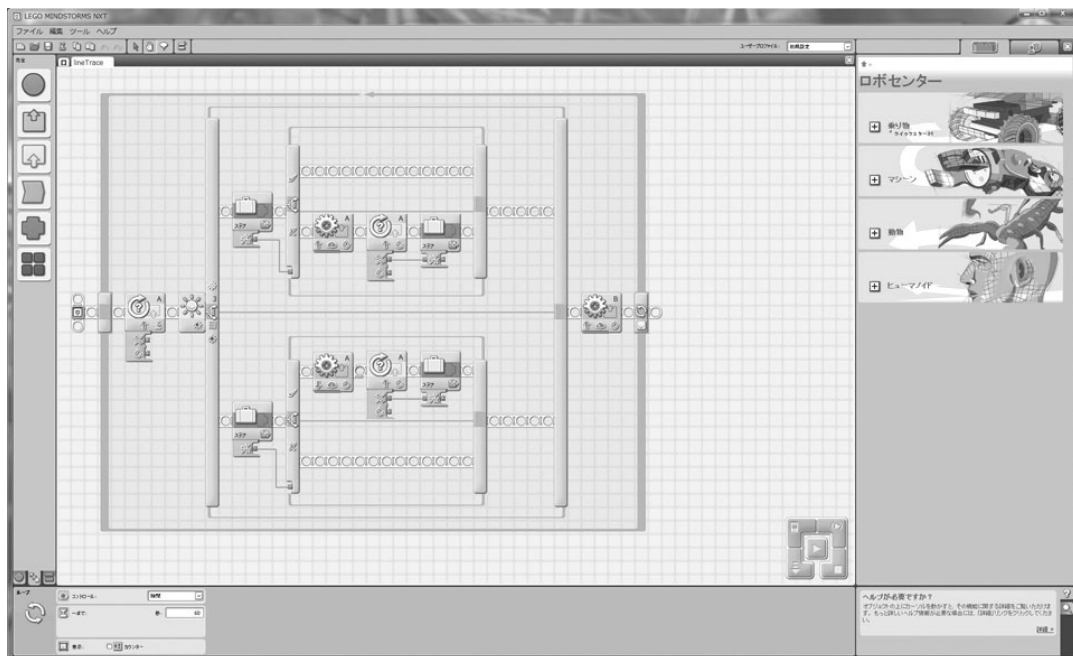


図 3 ホビー用 LEGO Mindstorms NXT に同梱された開発環境

モーターの動きやセンサーの詳細な設定は、別パネルの「プロパティ」で行う。変数や計算のためのパーツも揃っており、センサーの値を変数に代入したり、計算式の引数に使ったりすることができる。



図 4 プロパティパネル

Bricks は、NXT の制御を行うだけでなく、テキストや画像データの表示、各種サウンドの出力、ボタンを使ったユーザからの介入も可能である。

動作をプログラムする開発環境としては、前述のホビー用セットに同梱されているソフトウェアがある。正式には「LEGO Mindstorms NTX」と呼称するが、ここでは、ロボットのキットの名称との混同を避けるために、以下「NXT ソフトウェア」と呼ぶ。また、これ以外に同じく LEGO 社から販売されている ROBOLAB や、C 言語をベースにしたものなど、さまざまなものがある。



図 5 変数アイコン

そして、Microsoft 社は 2006 年に Microsoft Robotics Studio 1.0 をリリースし、現在は Microsoft Robotics Developer Studio 2008<sup>1)</sup> (以下 MRDS と略) に至っている。この環境は、汎用のロボット用の開発環境であるため、NXT だけでなく掃除ロボット Roomba やその他のホビー用ロボットなどの開発にも利用できる特徴を持つ。

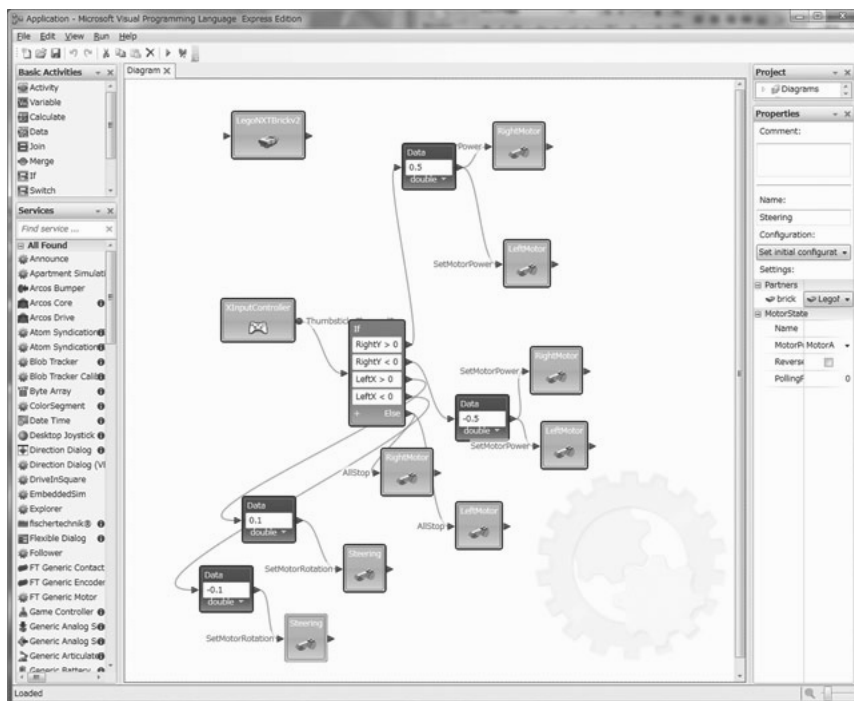


図 6 Microsoft Robotics Developer Studio 2008

### 3. 教育の概要

#### 3. 1 教育の目的とカリキュラムデザイン

この NXT を活用した教育の目的を以下のように設定した。

- (1) マルチメディアを活用した日常の事象をモデル化する考え方や手法を習得する
- (2) ハードウェアとしてのロボットをソフトウェアで制御することの意味を理解する。
- (3) ビジュアルプログラミング環境を用いて、アルゴリズム構築のスキルを養う。

(1)については、今回に限らず本プロジェクト研究の共通の教育目標である。今回の試みでは、前年度の「道路の交差点での事故に関する調査」の研究の延長として位置付けた。前述の調査では、千葉県警察東金署の協力を得て、大学の東金キャンパスがある東金市内の国道 126 号線の事故多発地点の交差点について、事故のデータを元に交差点の状況、車の日常の走行状態などから、事故防止策について考察した。

本研究では、交差点における事故を減らす方策として、交通信号を遵守することがまず重要と考え、機械（車）が自ら信号機の指示を守るシステムを提案する。

そのため、信号遵守をアシストするシステムのシミュレーションモデルの構築を、NXT を用いて行ったのである。動作原理は非常にシンプルであるが、それを実際に動作するようにシステムを作り上げるのが最大の目的である。

(2)は(1)と関連する項目であるが、今回はハードウェアについてのもの作りも行うことにした。筆者のマルチメディア教育においては、その大部分がソフトウェアの開発が主であった。しかし、実社会においては、マルチメディアの応用技術はハードウェアとの連携も重要である。しかし、今まで筆者が行っていたマルチメディア教育では、学生はなかなかハードウェアの構築技術までは学べなかった。それは、機材の購入などコストの問題が大きい。また、対象をロボットに限ってみると、非理工系の学生の場合には、構築スキルはほぼゼロであることも大きな理由である。今回のテーマである、車両ロボットを例に挙げると、専用研究・実験キットの場合は、1 台につき数 10 万円も費用がかかる。高度な制御を研究するのならそれも意義があるが、本教育においては、そこまでの精度や情報処理能力を必要としない。

そこで、今回は NXT という比較的安価な、基礎的な教育やホビー向けのキットを使うことにした。LEGO は子供の時に遊んだ経験がある学生が多くいるので、モデル作成を比較的短時間で済ませられることが期待できる。

(3)に関しては、本学部の学生の多くは、プログラミング言語や開発環境に慣れていない。今までも、Microsoft Visual Studio や Adobe Flash の Action Script などを使い、各種のシステムを学生に作らせてきたが、処理の流れやデータの流れに対する理解度が乏しい。これは、今まで高校の「情報」の授業を含めて、実践的なアルゴリズムの教育を受けていない学生が多いと思われる。また、プログラミング言語に対する苦手意識が強く、自ら学ぶという姿勢に欠けることがしばしばあった。

前述のような開発環境もビジュアルインターフェースや、ガイド機能を充実させ、作業自体の負担は比較的低くなっているが、ソースコードはあくまでもテキストベースである。それに対し、NXT ソフトウェアや MRDS は、プログラム自体がビジュアル化されている。動作手順やデータの流れは、ブロックのつながりや矢印で示すことができ、正にブロックを積み上げる感覚でプログラムを組むことができる。

筆者は以前に Apple 社の HyperCard によるアルゴリズム教育ツール<sup>2)</sup>を開発したことがあるが、基本的なアルゴリズムの構築方法は同じである。これは、今までプログラミングの経験が無い学習者でも、比較的容易に簡単なプログラムを組むことができたので、初心者に対して、NXT ソフトウェア

などのビジュアルプログラミング環境が有効であることは経験済みである。

3 年次後期から本教育を開始した。当初はロボットキットとして NXT を使い、開発環境としては MRDS を使用する予定であった。それは、以下の理由による。

まず、MRDS はロボットのハードウェアが無くても、付属の Visual Simulation Environment（ビジュアルシミュレーション環境、以下 VSE と略す）で、仮想的な動作確認ができる。NXT は比較的安価であるが、1 人につき 1 台ずつ用意することは費用の上から難しかった。また、NXT を学生に長期貸し出すことは管理上問題がある。しかし、MRDS なら、プログラミング環境は Express エディションを使えば無料であり、NXT のハードウェアがなくても、VSE で授業時以外でもプログラムのデバッグが可能である。

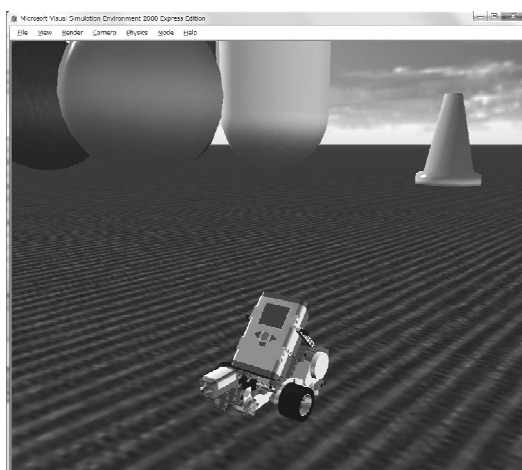


図 7 Visual Simulation Environment

そして、今回の NXT だけでなく、ホビー向け 2 足歩行ロボットや、掃除ロボット Roomba などの動作もプログラムすることができた。これは、システム作りに汎用性が出て、さまざまなプラットホームに応用が効く。

しかし、実際には MRDS の導入を断念した。その理由は、学生所有のノートパソコンでは、グラフィックの処理機能が VSE の動作に必要なスペックを満たしていなかったのである。これは、入学時に大学推奨のパソコンを購入した学生にのみ該当した。他機種を購入した学生については、筆者の所有のノートパソコンを含めて、VSE は動作が確認できた。

もう 1 つの理由は、ロボットとコンピュータを接続するには、Bluetooth が必須であった。学生所有のほとんどのノートパソコンには Bluetooth は内蔵されておらず。学生の中には Bluetooth について、不案内な者も少なくなかった。現在は国内でも Bluetooth レシーバーは安価な物が多く問題ないが、NXT と Bluetooth の相性問題もあり、モデルの制作以外での学生の負担を負わせるのは適切でない判断した。

そこで、前述のように MRDS は使わず、開発環境は NXT ソフトウェアを用いた。

### 3. 2 3 年次後半の活動

3 年次の前期の授業では、大学が所在する千葉県東金市の中心を通る国道 126 号線に多く点在する事故多発地点での交通実態調査を、メディア学部ならではの視点から考察することをテーマとした。たとえば、実際に道路を走行する車の速度を、ビデオカメラを利用して計測したり、信号が黄に変わったときに信号機からどのくらい手前を走っている車が停車するかなどを実地調査した。それに加え、千葉県警察東金署の協力を得て、事故多発地点での事故の事例を調べ、秋季の学園祭に成果を展示発表した。

それらを踏まえ、プロジェクト研究の学生と今後の活動について討議し、4 年次の最終成果として、交差点での事故を減らすための提案を行うことにした。そのスタイルとして、事故を未然に防ぐような交通シミュレーションモデルを構築することにした。このことについては、後に「4.1.シミュレーションモデルの概要」で述べる。

シミュレーションモデルを構築するにあたり、何を使ってどのようにシミュレーションするかは、学生にとっては未知の要素が多くあり、なかなか良いアイデアが出てこなかった。そこで、一部の学生が夏期に参加した、企業のインターンシップのプログラムで経験した LEGO Mindstorms を筆者が提案した。プロジェクト研究の学生の約半数は経験しており、それ以外の学生も関心が高かったので、すぐに教材とした。

3 年次後半は実質 3 ヶ月ほどしかなかったので、キットに同梱されていたチュートリアルを使い、サンプルロボットの組み立ておよび動作のプログラミング実習を行った。このときは、NXT のキットは 2 組だけであったので、11 人いるプロジェクト研究のメンバーを 2 組に分け、それぞれで制作させて動作の正確さを競わせることにした。

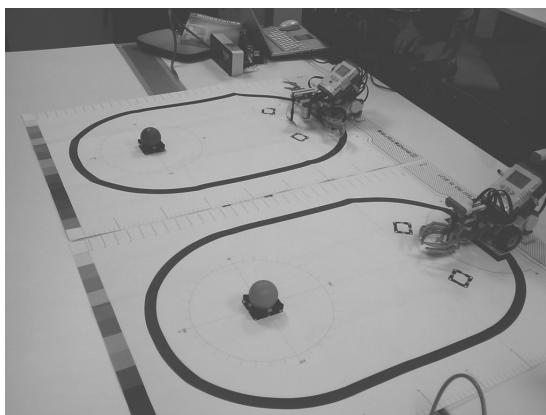


図 8 チーム対抗戦

この実習は、NXT のキットの各パーツの特性の理解と組み立てのスキルを身につけることが主な目的である。幼少時に LEGO で遊んだ経験のある学生でも、LEGO テクニックを組み上げた者はいなく、アームやコネクタ、シャフトといったパーツには馴染みがない。まして、モーターやセンサーの取り付けなどは、経験がないと難しいと判断したからである。



この実習では、車輪で走行するロボットや多足歩行ロボットなどを組み立てては作り直すことを繰り返し、NXT ソフトウェアでさまざまなセンサーを制御するサンプルプログラムを作ることで、自然とスキルを向上させることを目指した。これは、もともと NXT に学生たちが大いに興味を持っていたことから、自らすずんで学んでいくだろうと予想したからである。筆者はプロジェクト研究の活動毎に毎回テーマを提示し、最低限のアドバイスをするだけで、完成後のグループ対抗競技を行うまでは、ほとんど学生に任せきりとした。



図 9 Mindstorms NXT の組み立て風景

最初の数回の実習授業では、学生は問題なくチュートリアルをこなしていき、筆者が当初望んでいたように、順調に課題をこなしていたが、途中で 2 グループとも正確に動作させることに苦労した点があった。それは、ライントレースである。ライントレースの課題で、NXT キットに同梱されている練習コース上の、黒い太線で描かれたオーバルコースを光センサーで読み取って線に従って走行するものである。

NXT の組み立ては、慣れてきて問題なくセンサーの取り付けも終わったのだが、サンプルプログラム通りに NXT ソフトウェアでプログラムを組んでも、正しく動作しなかったのである。ライントレースに使われるロボットのモデルは、通称トライポッド（3 輪車型ロボット）と呼ばれており、駆動輪の左右のタイヤにモーターが直結されている構造である。この左右のモーターの回転を変化させることで直進、左右の転回などを実現する。

光センサーで反射する光の強さを読み取る。センサーが線上にあれば反射する光は暗くなり、線から外れれば明るくなる。たとえば線の右端を読み取ることになると、センサーの値が明るければ左方向へ、つまり右モーターを動かし、暗ければ右方向へ、つまり左モーターを動かせばよい。これを繰り返すことによって、ジグザグ運動をしながら、線上をロボットが走行する。

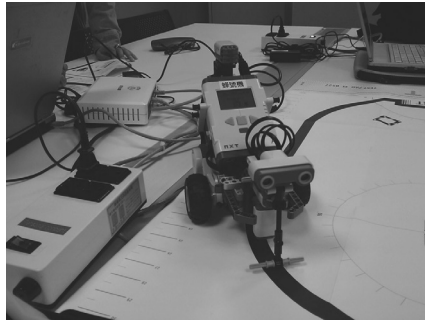


図 10 ライントレースするトライポッド

原理は単純なのだが、モーターの回転量を増やしすぎると、次に光センサーを読み取るまでに線を跨いでしまい、コースアウトするか、トライポッドが一周して元に戻るような動作を示す。学生たちは最初、コースアウトすることに動揺するが、すぐに光センサーの明るさの閾値の設定が不適切だったと判断し、値を修正するようになる。ただ、それでも動作が改善されないと思考が停止してしまった。

これは未だプログラムの繰り返し処理の概念の知識が定着していなかったのである。そこで、繰り返しの処理で、次の光センサーの読み取り処理までに、トライポッドがどのくらい動いているかを観察させた。このアドバイスにより、モーターの移動距離に適切な値があることに気がつき、モーター部分のプログラム修正を行うようになった。これで、ライントレースはうまく動作したのだが、次の課題が発生した。それは、2 グループで競争するので、正確さとともに動作速度も必要であるということである。これにより、正確な動作とスピードという、相反する要素の妥協点を見つけることの必要性を体得したことになった。

このように、チュートリアルだけではうまく動作しないときの対処法や、競わせることで動作速度というファクターの存在を自然と身につけることができた。そして、何よりグループ対抗とすることで、本プロジェクト活動のモチベーションの維持にも役立った。

### 3. 3 4 年次前半の活動

3 年次後半時には、4 年次で制作するシミュレーションモデルについて、具体的なイメージを持っていなかったため、明確な目標を定めさせた。それは、交差点のモデルを作成し、そこに NXT によって作成した車を走らせ、交差点での事故を未然に防ぐような、交通安全サポートシステムのシミュレーションを構築し、提案することであった。

これは、交差点の信号機も NXT で作成し、信号機と車を Bluetooth で通信を行うことで、自動車の走行を制御するものである。どのような事故を想定するかは後回しにして、前期では、オリジナルの自動車を制作することと、それを NXT ソフトウェアで制御することを目標とした。

実際の車両の構造を知っている学生が、ほとんどいなかったために、ステアリング機構やディファレンシャルギアの役割についての簡単な説明を行った。NXT のホビー用のキットにはこのディファレンシャルギアは同梱されていないが、LEGO テクニック系のパーツとして市販されているので、必要

なギアのセットを別途用意した。また、信号機の動作も NXT で行うことを考えていたので、それ用に Brick 本体と、RCX 用のライトパーツ（電球）、RCX 用センサーを接続するための変換ケーブルも用意した。

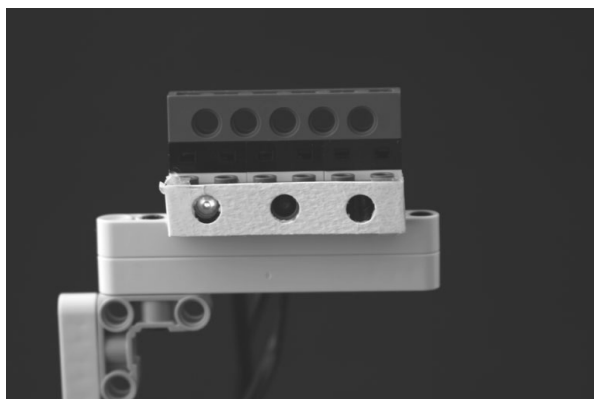


図 11 LEGO パーツを使った信号機モデル

車両の制作事例をいくつか見せながら、車両の主要パーツの組み立てのノウハウや、センサーの取り付け位置などを検討した。

その後、3 年次の時と同様に 2 つのグループに分け、車両制作に取りかかった。1 つのグループは、車の構造に詳しい学生がいたために、その学生が中心として試作機を 1 月ほどで製作した。一方、もう 1 つのグループは、前者とは対照的になかなかオリジナル車体を組み立てることができなかった。この件に関しては、LEGO への関心の度合いと対象のモデルに関する知識の有無が大きく関係しているものと思われる。また、グループ作業の場合には、リーダー的役割をする学生の力量も影響している。

ただし、グループ内でも 1 人のスキルが他と比べて甚だしく優れている場合には、その学生が 1 人で作業を行いがちで、他のメンバーが積極的に加わるができなくなる場面もしばしば見られた。

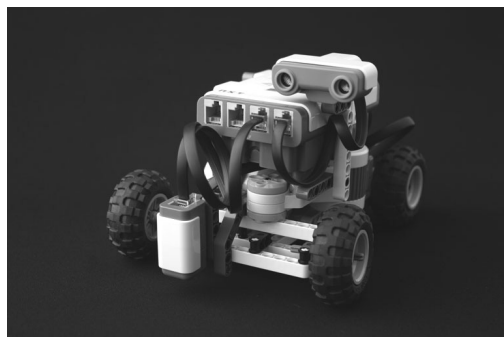


図 13 筆者が制作した車両

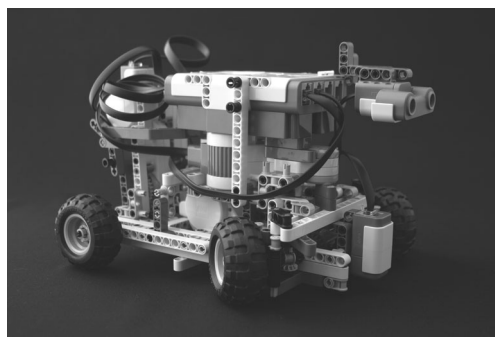


図 12 学生が制作した車両

オリジナル車体の制作ができなかったグループについては、筆者が設計した車を利用することにし

た。ただし、組み立て手順を撮影したビデオを見せながら、実際に学生に作らせ、NXT のキットの組み立てのノウハウを、できるだけ習得できるように考慮した。

オリジナル車体を制作したグループは、あまりにも車両の構造を複雑にしたため、パーツの数が多くなり、重量、車体サイズとも過剰な設計となっていた。これについても、筆者が組み立てたオリジナル車体を見せ、車体の簡略化と軽量化を促した。

これらの作業によって最終的にシミュレーションで使う車両を 4 台制作した。うち、1 台は学生オリジナル車両で、後の 3 台は筆者が設計したものを学生が組み立てたものである。これに加えて、NXT の Brick を使った信号機を 2 台制作した。これは、プログラムによって電球 3 つを点灯・消灯させるもので、同時に Bluetooth で車両にも指示を送るためのものである。

車両が完成した後、NXT ソフトウェアで車を自由に動かすプログラムの制作に取りかかった。この目的は、車のステアリングの切れ角の特性や、速度の制御が意図した通りにできるかを確認すると同時に、NXT ソフトウェアで自由にプログラムを組むことのトレーニングであった。

課題は簡単で、8 の字のコースを設け、チェックポイントを曲がり、スタート地点に正確に戻ることを課した。このときは、まだライントレースで動かすのではなく、ステアリングの切れ角や速度の調整だけでやらせたのであるが、1 時間も行っていると、試行錯誤を繰り返し、課題をこなせるようになった。

ただ、ここで学生は 1 つの問題に気がつき始めた。それは、電池が消耗していくとモーターの出力が変化し、同じプログラムでも走行軌跡が変わってしまうことである。つまり、これによって学生は、センサーなどによって何らかの制御が必要であることを自ら発見したのである。

次に予備実験として、Bluetooth による車両間の通信を行った。これは、Bluetooth を使う前のペアリングの作業や、通信の手順をあらかじめ慣れしておくのが目的である。普段学生が使っている情報機器では、Bluetooth を活用する機会がほとんどない。そのため、Bluetooth という言葉自体を知らない学生が多い。そこで、Bluetooth の仕組みから活用方法まで一通りの説明を行ってから実験を開始した。

実験はシンプルで、2 台の車両を向かい合わせに走らせ、1 台をマスター、もう 1 台をスレーブとして、ある距離まで近づくとマスター側の距離センサーが反応し、Bluetooth を経由してスレーブ側にメッセージを送る。その信号を受けたスレーブ側の車両が迂回して回避行動をとるものである。これらのプログラムをマスター、スレーブの車両にそれぞれ転送する。

Bluetooth のペアリングやメッセージ授受は容易にプログラミングが行えたが、車両をまっすぐに走らせることが、なかなか容易ではなかった。車両を直進させることは単純なのだが、2 台の車両を向かい合わせに、フリーで同一直線上を走行させるが難しいのである。車両をスタートさせるときは、回避行動をさせるために車両間を最低でも 2m ほど離す必要がある。しかし、走行している途中で、2 台の車両の位置がずれて、なかなか正面からぶつかる状態を設定するのが困難だったわけである。

そこで、交差点のシミュレーションモデルのデモンストレーションでは以前のチュートリアルで行ったライントレースのアルゴリズムを用いて、決められたコースを走らせることにした。

これで、学園祭の展示発表で行うデモンストレーションの準備に取り組むことができた。

### 3. 4 4 年次後半の活動

3.3.で制作した車両と信号機を使い、学園祭の展示発表用のデモンストレーションをするためのプログラムとコース作りを行った。

コースについては、図 14 のような 2 つのオーバル状のループが 2 カ所で交差するものを作成した。この交差した一方の場所に信号機のモデルを置き、作成したシミュレーションモデルのデモンストレーションを行うこととした。

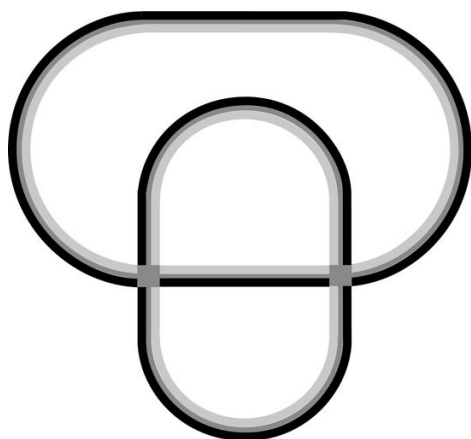


図 14 デモ用コース図

車両のプログラムは、黒いライン上を走らせるためのライントレースの処理と、信号機からのメッセージを受け車両の進行を制御する処理からなる。信号機は交差する 2 つの道にそれぞれ設置した。1 つの信号機(Master)のプログラムでは、もう 1 つの信号機(Slave)と、2 台の車両(Car-1、Car-2)に Bluetooth でメッセージを送る。そして、同時に 3 つのライトのパーツの点灯を制御する。信号機(Master)は、NXT パーツのタッチセンサーを使い、センサーを押す毎に通常の信号機と同様に、青→黄→赤→青→・・・とライトの点灯が変化する。それと同時に、Bluetooth によってメッセージを送信する。

メッセージを受けた別の信号機は別のプログラムでライトの点灯を変える。また、2 つの車両は前述のような共通のプログラムで動作し、進行方向の信号が赤の場合は、距離センサーにより、信号機の手前で停止し、信号が青であればそのまま交差点を通過したり、動き出したりする。

これら NXT で作成したモデルと信号機をデモコースで運行した。最初はライントレースする車両がこまめにジグザグに走るためにスムーズに走行できなかった。これは、1 本の線上を走行するために、1 つの光センサーで黒い線と白い紙との反射光の強弱で線の端を読み取り、常に線の左右どちらかの縁に沿って動くためである。学生はこのライントレースをスムーズに行うための改良が困難であったため、次のような提案を行った。

それは、線に 3 種類の濃淡を付け、中心の線の幅を広げることで、走行時の進路の修正回数を減らし、1 つの光センサーで中心の線から左右どちらにずれているのかを効率的に感知できるようにした。これによって、ライントレースの走行プログラムを簡略化しつつ、スムーズな走行を実現できた。

これによってデモ走行と信号機のプログラムが図 15~18 のように完成し、学園祭において展示を行った。ライントレースの走行に関しては、まれにコースを逸脱するような場面が見られたが、所期の目標であった、交差点における信号を遵守するプログラムはほぼ実現できた。

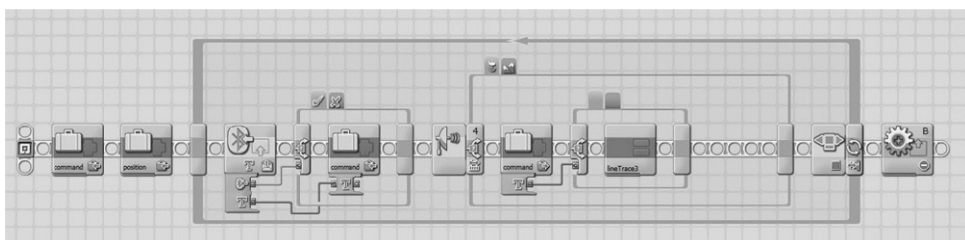


図 15 車両の信号遵守プログラム部

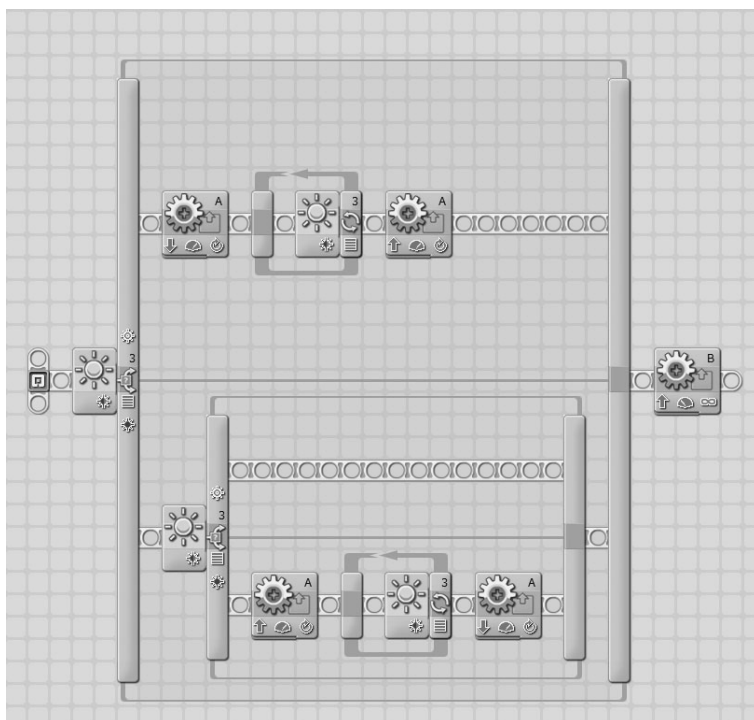


図 16 車両のライトレース部

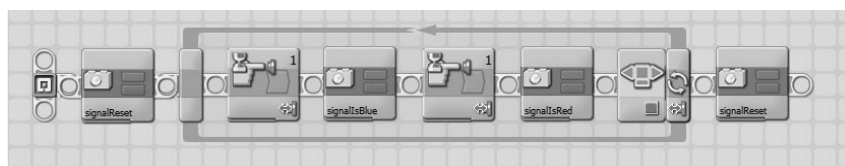


図 17 信号機(Master) 部 (一部省略)

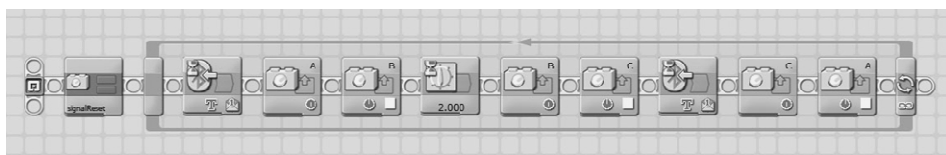


図 18 信号機(Slave) 部 (一部省略)

## 4. 教育成果

前述のように、交差点で信号を遵守する車両のモデルが完成し、学園祭において走行デモと展示を行った。見学者の中には、NXT で作成した車両の動きに興味を持っている者が少なからずいた。ただし、プロジェクト研究の学生全員がシステムを完全に理解しているとは限らなかったため、見学者への対応が不十分であり、満足な研究発表ができたとは言い難かった。

ただし、何も無いところから 1 つのシステムを構築したことへの達成感と、より高度なシステムの構築への向上心は養われたと感じる。



図 19 学園祭でのデモその 1

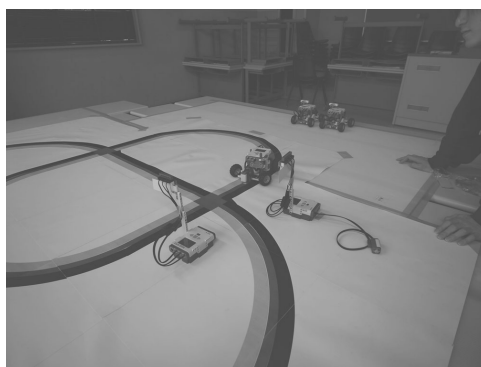


図 20 学園祭でのデモその 2

一昔と違い、学生にとってロボットの存在は身近なものとなった。また、ロボットに対する関心の高さは予想以上のものであり、学習のモチベーションを維持することに一役かっていると思われる。学生は、ハードウェア（ロボット）とソフトウェアの連携した知識の習得や、マルチメディアの領域に対する広い視野と知識でたとえば「交通事故を減らす」といった身近な問題を捉えられるようになったことで、マルチメディア教育において、NXT の導入は有効な手段であると考えられる。

ただし、問題点もいくつか表面化した。たとえば、4 年次の前半の活動では、1 つのグループでは、オリジナル車両（ロボット）を制作することができなかった。これは、学生が子供期に慣れ親しんだ LEGO と違い、ほとんど組み立てたことがない LEGO テクニックのパーツがメインだったことが挙げられる。また、可動部品の組み立てや駆動系の制作などはほとんどの学生にとってスキルが無い状態だったのも一因と思われる。

そして、ロボットを単純に動かすだけなら、3 年次のチュートリアルでカバーできたが、センサーや Bluetooth を活用したプログラムの開発は、なかなか馴染めなかったようである。今まで受けていた大学の講義には出てきていない内容であるために、理解や応用の手法に戸惑いがみられた。学生によっては、いろいろと試行錯誤をする者も見受けられたが、ほとんどの学生は思考が止まり、なす術なしの状態であった。基本の学習から応用へという一般的な学習手法ではなく、別の教育法を考え適用する必要があると思われる。このことについては、今回はこれ以上言及しない。

## 5. おわりに

今回は、マルチメディア教育において、教育用ロボット LEGO Mindstorms NXT を導入し、その意義と成果について述べた。現在のロボットのキットは、理工系の学生でなくても容易に組み立てることができ、各種センサーを活用することによって、マルチメディア応用システムの1つのモデルを構築することができた。

ロボットの制御をするためにプログラミング環境は、ビジュアルプログラミングを利用することができ、学生への負担を軽減しつつ、動作アルゴリズムの構築を容易に行えることが確認できた。

しかし、すべての学生が筆者の思惑通り、システムの構築ができたわけでない。基本学習からオリジナルモデルの構築に移行する上で、ついて行けない学生も半数出てしまった。学生に直接聞いてみるところ、ロボットへの関心は高いのだが、オリジナルロボットの制作だけでなく、プログラミングまでこなすことへの負担と苦手意識が多かれ少なかれあったようである。

もう1つの問題点は、筆者のプロジェクト研究では、5人程度の2つのグループで本活動を行っているが、どうしても組み立てやプログラミングが得意な学生に、ほかの学生が頼りがちになり、システムの品質は個人の力量に依存してしまったことがある。これについては、グループワークにおける分業のテクニックなどを、別の機会に教えておくべきであった。2つのグループにした理由は、単に用意した NXT の数が2セットということだった。

また、チュートリアル時点から、積極的にワークに参加する学生と参加しない学生に分かれてしまった。プロジェクトメンバーの学生の総意でこのテーマを取り上げたのだが、学生らも理想と現実のギャップに、徐々にモチベーションが下がる学生が結果的に出てしまった。進捗状況が芳しくないグループや学生に対して、補講的な授業を行いフォローしたのだが、理解度を満足できるレベルまで引き上げることができなかったことが反省点である。

今後は、今回の事例における問題点を改善し、交差点での右折対直進の事故のパターンに対応するシミュレーションモデルを制作したり、より学生が学習意欲を高められるような魅力あるマルチメディア教育を実現したい。

## 【参考文献】

- 1) <http://www.microsoft.com/robotics/>
- 2) 中嶋 正夫 他：“ビジュアルプログラミング環境と問題解決アプローチの定量評価システム”，Navigware フォーラム 1997 年論文集，p.113-121 1997.
- 3) 大庭 慎一郎：“入門 LEGO MINDSTORMS NXT”，ソフトバンククリエイティブ（株）.
- 4) 五十川 芳仁：“ブロックで作るキカイの本 走る・曲がる・車のしくみ”，（株）毎日コミュニケーションズ