

計測・制御を目的とするIoT を利用した学習教材の検討

メタデータ	言語: Japanese 出版者: 静岡大学学術院教育学領域 公開日: 2019-12-24 キーワード (Ja): キーワード (En): 作成者: 安達, 悠真, 室伏, 春樹 メールアドレス: 所属:
URL	https://doi.org/10.14945/00026963

計測・制御を目的とする IoT を利用した学習教材の検討

Consideration of Teaching Materials using IoT that aim the Measurement and Control.

安達 悠真¹, 室伏 春樹²

Yuuma ADACHI, Haruki MUROFUSHI

(令和元年 12 月 2 日受理)

ABSTRACT

We have developed measurement and control materials for learning about IoT. The learning of IoT corresponds to the understanding of network utilization and interactivity required in the new Courses of Study. Our teaching materials aim at learning about edge devices, gateways, and servers, which are components of IoT systems. In this paper, the teaching material was developed on the edge device. An Arduino-compatible communication board is used for the edge device, and ECHONET Lite is used for the communication protocol. Thereby, using this, the learner can make the equipment which cooperates with the IoT equipment in the home. Therefore, it is the teaching material that can return the learning content to life.

1. はじめに

情報通信技術が発展し、IoT (Internet of Things)と呼ばれる技術が社会に普及しつつある。IoT とは、身の回りにあるセンサや制御機器を搭載する製品(モノ)が通信機能を持ち、直接インターネットに繋がり、機器同士でそれぞれが取得した情報をやり取りすることで、個人や会社などにとって最適な環境やサービスを実現し、そのモノに元の価値以外の新たな価値を付加するものである[1]。この IoT は AI(Artificial Intelligence : 人工知能)やビッグデータなどとともに社会の産業に大きな革新をもたらす第四次産業革命の技術とされている。内閣府では、「サイバースペース（仮想空間）とフィジカル空間（現実空間）を高度に融合させたシステムにより、経済発展と社会的課題の解決を両立する、人間中心の社会」とした Society5.0 を提唱している[2]。したがって、今後の日本社会では IoT 機器が大幅に増加することが予想される。

総務省によると世界中の IoT デバイス数は 2020 年には約 400 億台にものぼると予想されている[3]。つまり、IoT を活用した Society5.0 の到来が近づいている。よって、IoT が現代の日本で生きる私たちや、これからを生きていく子供たちにとって身近なものとして理解を深めなければならない。

生活で身近な IoT 機器にスマートフォンが挙げられる。スマートフォンは電話としての通話機能に加え、インターネット経由で天気や交通情報を閲覧する機能やスマート家電[4]の遠隔操

¹ 教育学研究科大学院生

² 技術教育系列

作機能などを有している。また、新たな移動通信方式である第五世代移動通信式システム(5G)も発表された。5Gは、「大容量・高速・多数接続・低遅延」が特徴とされており、情報共有がより即時的なものとなるため、Society5.0が目指す仮想空間と現実空間の融合が促進されるものと考えられる。しかし、スマートフォンの中身は高度なシステムであり、IoTの原理を理解するために利用するのは適当ではない。そのため、IoTを学習するための教材の開発が必要であると考えた。

教材の開発にあたり、IoTの定義を八子による「IoTはモノをネットワークに接続し、複数のデータを収集・分析し、分析によって得た示唆や予測モデルのもと、結果を結果や設備（エッジデバイス）にフィードバックして最適化を行う[1]」ものとした。本研究ではこの定義のもと、IoTにおけるデータを収集するセンサ部分と最適化を行うアクチュエータ部分である端末の要素を「エッジデバイス」とする。また、エッジデバイスが収集したデータを集約する要素を「ゲートウェイ」、集約したデータを分析・制御するクラウドサービスの働きをする要素を「サーバ」とし、この3つをIoTの構成要素とする。これを用いて学習者自身がデータの収集から分析など機器の制御や通信の流れについての理解につながる教材の開発を目的に研究を行った。

本論文は全5章からなる。第2章では、本研究の研究背景について述べる。第3章では、本研究で開発するIoTシステムを提案し、教材として開発する範囲を特定する。第4章では、通信に利用するプロトコルのECHONET Liteについて述べる。第5章では、開発したエッジデバイスについて開発した機器やそれを用いた動作検証について述べる。第6章で本研究における考察を述べ、第7章でまとめを述べる。

2. 研究背景

2.1 学習指導要領から見るIoTの学習意義

平成29年3月に中学校学習指導要領(以降、指導要領)が告示された。指導要領の技術・家庭技術分野(以降、技術科)の変更点として、「D 情報の技術」において、従来の計測・制御に加え、「ネットワークを利用した双方向性のあるコンテンツのプログラミングについて取り扱う[6]」旨が示された。ネットワークを利用した双方向性とは、「使用者の働きかけ(入力)によって応答(出力)する機能であり、その一部の処理の過程にコンピュータ間の情報通信が含まれていること[7]」であり、コンテンツとは「デジタル化された文字、音声、静止画、動画などを、人間にとて意味のある情報として表現した内容[7]」である。つまり、指導要領では、何らかのコンテンツを扱うシステムに対し、使用者が入力したデータを、情報通信技術を利用して複数のコンピュータで処理し、利用者にとって意味のある情報として出力される仕組みや手順をプログラミングを通して学習することが求められていると考えられる。そのため、提示するコンテンツが具体的で学習者の課題意識に即していることが必要である(要件①)。

IoTの学習では、エッジデバイスからゲートウェイを介したサーバへの情報通信でデータの分析し、その結果のデータをサーバからゲートウェイを介してエッジデバイスを制御、またはサーバから直接制御するという3つの要素の一連の流れによって環境の最適化を行うことを理解する必要がある。この学習では、エッジデバイスでのデータの収集やアクチュエータの動作を入出力と考え、エッジデバイスからサーバまでのデータ通信をコンピュータによる情報通信と考えると、技術科の指導要領に示された「ネットワークを利用した双方向性のあるコンテンツ」に当てはまると考えられる。そのため、IoTの学習で利用する教材は、計測・制御の学習ができる

て情報通信機能を有することが必要である(要件②)。

さらに、平成 30 年 3 月に高等学校指導要領(以降、高校指導要領)が告示された。高校指導要領では、共通必履修科目として情報 I が新たに設置された[8]。情報 I では、技術科の内容である「D 情報の技術」との連携を重視している。また、情報 I において「問題の発見・解決に向けて情報と情報技術を活用するための知識と技能を身に付け、情報と情報技術を適切かつ効果的に活用するための力を養い、情報社会に主体的に参画するための資質・能力を育成する[9]」ことを狙いとしている[10]。つまり、今回新設された情報 I を含む共通教科情報は技術科からの系統性をもつたため、学習者の生活に還元できる教材が必要である(要件③)。

2. 2 先行研究

IoT に関する教材に関して調査するため、国立情報学研究所が提供する学術情報データベース CiNii で「IoT 教材」で検索すると 22 件該当した(2019/08/29 現在)。このうち前節で示した 3 つの要件のうち①と②を満たすものとして高橋ら、辻らの研究がある。高橋らは IoT 技術を利用して気温・湿度・気圧の計測データを扱う教材について研究を行っている。高橋らは ESP-WROOM-02 モジュール等を利用し、IoT の基礎的な理解について学習ができ、高校生でも理解できる授業内容と教材を開発している[11]。また、辻らはエンジニアの育成を目的とした IoT におけるセンシング技術を利用した学習教材の開発と実践を行っており、Arduino 互換のマイコン、ZigBee、クラウドサーバなどを組み合わせた気象モニターを大学生向けの講座で実践をし、初学習者にも IoT の概念を理解させ、興味関心を引き出すことができることを報告している[12]。これらの実践は、ESP-WROOM-02 モジュール等の具体物を使用して計測・制御に加え、情報通信も行っており、かつ学習が積極的に取り組んでいることから、いずれも要件①と②を満たしていると考えるが、これらは専用の環境内でのみ運用できるものであり、要件③である学習者の生活に還元できるものではない。

3. IoT 学習教材の提案

2.1 節に示した IoT の学習意義の要件を満たすものとして、ローカルエリアネットワークで利用するプロトコルにスマート家電や HEMS(Home Energy Management System)で標準規格になっている ECHONET Lite[13]を利用する IoT 学習教材を提案する。ECHONET Lite とは、エコネットコンソーシアムが規格化した ECHONET プロトコルの一種である。ECHONET Lite は約 100 種類以上のセンサやアクチュエータが定義されており、Arduino のようなマイコンだけでなく、家電量販店で販売されている製品などにも対応している。そのため、ECHONET Lite を利用することで、生活で実際に利用する IoT 機器のブラックボックス化された情報通信について理解するとともに、学習者自身が製作した機器を家庭の ECHONET Lite 対応機器と接続することで、授業で学習した IoT システムの流れを実感しながら自分の生活に活かすことのできる教材になると考えた。

本研究で提案する IoT 学習教材の概要を図 1 に示す。この教材は IoT の構成要素である「エッジデバイス」「ゲートウェイ」「サーバ」それぞれの役割の違いを学習者が理解しやすいように扱う機器を変える。これにより、システムとしての理解を深めるとともに学習者が機器ごとに分担して製作する協働的な学習も実現できると考えた。

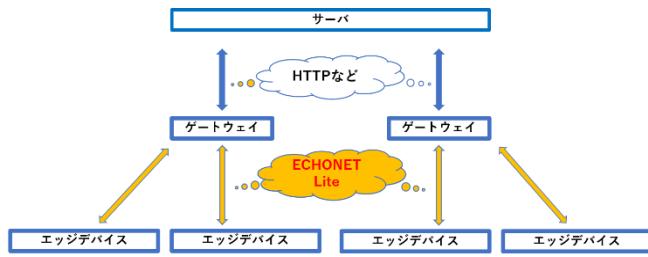


図 1 IoT 学習教材の概念図

実際の開発は計測・制御を行うエッジデバイスの開発を対象とした。これは指導要領における「D 情報の技術」の(3) プログラムによる計測・制御との関連のためである。開発するエッジデバイスは、センサの値を取得することとアクチュエータを制御することを行うことができるアクチュエータ用基板、他の機器に命令を送信したり他の機器から値を取得したりするコントローラ用基板の開発を行う。開発にあたり、①学習者が製作可能であること、②情報通信機能があることを必要な要件と考えた。また、2種類の基板にはそれぞれの働きがあるので、アクチュエータ用基板は、③様々なセンサ・アクチュエータのモジュールを扱うことができること、④計測や制御の内容や動作時間など学習者が任意に動作させることができることを要件に設定した。コントローラ基板では、⑤複数の機器に対して信号が送れること、⑥操作の難度が高すぎないことを要件に設定した。

4. ECHONET Lite の情報通信

4. 1 ECHONET Lite 通信の概要

エッジデバイス間での通信は ECHONET Lite 規格のプロトコルを用いて行う。ECHONET Lite 通信は UDP 方式を用いて IP アドレスを利用したユニキャスト通信や同一ネットワーク内の機器すべてに通信を投げかけるマルチキャスト通信を行う。15 バイト程度のパケットにより構成された電文で HAN(Home Area Network)内で通信をする。ECHONET Lite 通信による電文のプロトコルマップを図 2 に示す。9種類の各データで通信する機器の種類や動作内容などを示し電文を構成する。例えば、図 2 中の ESV(ECHONET Lite サービス)では、パケットの主な種別を表す。ESV は相手のプロパティ値を読み出す Get、相手のプロパティ値へ書き込みを行う(相手を制御する) Set、自身のプロパティ値を通知する INF の 3種類を定義する。

ECHONET Lite では、アクチュエータ、センサ、コントローラなどの機器もネットワーク内で電源を入れたり、ネットワーク内に参加したりしたときに自身がネットワークに参加したことをマルチキャストで他の機器に通知する。この通知する動作を参加通知という。参加通知時のシーケンスを図 3 に、プロトコルマップを図 4 に示す。参加通知では ESV の種類のうち INF で通信を行う。この通信では、送信元の機器の種類を示す SEOJ(送信元オブジェクト)の値を 0x029001、送信先の機器の種類を示す DEOJ(送信先オブジェクト)で 0x0EF001 という値にすることで、一般照明機器からネットワーク内のすべての機器に通知をすることを表す。ESV は、0x74 にして INF を示す。動作内容を示す EPC(ECHONET プロパティ)の値を 0x80 にして動

EHD 2Byte	TID 2Byte		SEOJ 3Byte			DEOJ 3Byte		ESV 1Byte	OPC 1Byte	EPC 1Byte	PDC 1Byte	EDT 1Byte
--------------	--------------	--	---------------	--	--	---------------	--	--------------	--------------	--------------	--------------	--------------

図 2 ECHONET Lite 通信のプロトコルマップ

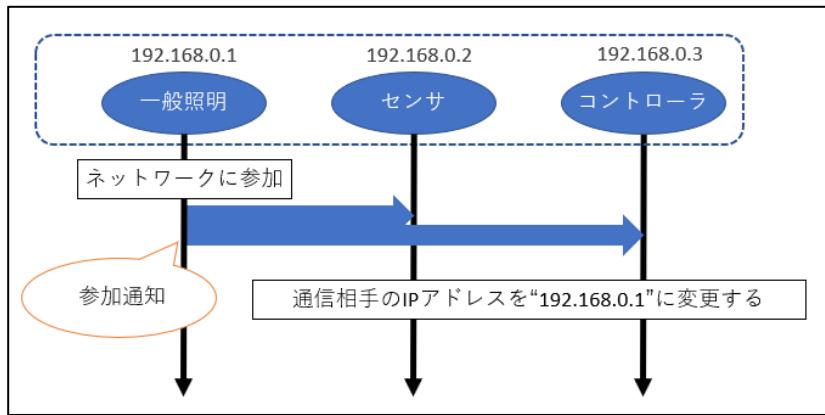


図 4 参加通知のシーケンス図

EHD	TID	SEOJ			DEOJ			ESV	OPC	EPC	PDC	EDT
1081	0000	02	90	01	0E	F0	01	74	01	80	01	31

図 3 参加通知時のプロトコルマップ

作の詳細を示す EDT(ECHONET プロパティ値データ)を 0x30 にすることで機器の動作状態が運転中であることを示している。EPC は機器の種類によって機能が変わるために、指定する値は DEOJ によって異なる。また、EDT の値は EPC に対して同様に依存する。

今回の電文では、動作状態が運転中である照明機器がネットワーク内のすべての機器に向かってネットワークに参加したことを通知する内容になっている。上記で示した値を ECHONET Lite 規格書の APPENDIX で示されている値にすることで様々な機器を制御することができる。

4.2 ECHONET Lite の通信フロー

ECHONET Lite における通信フローについて i) 照明の ON と OFF の操作に関わる通信と ii) 照明の調光の操作に関わる通信、iii) 照度センサの値の取得に関わる通信の 3 つを例示する。

i) は EDT の値を機器の動作のトリガーとして扱う通信の例、ii) は EDT の値をそのまま出力する通信の例、iii) は取得した値を他の機器に通知する例として示す。

i) 照明の ON と OFF の操作に関わる通信

照明の ON と OFF の操作に関わる通信で使用する電文を図 5 に、通信のシーケンスを図 6 に示す。コントローラから照明機器に対しての制御命令（書き込み要求）であるため、SEOJ を 0x05FF01、DEOJ を 0x029001 に設定し、ESV を 0x60 に、EPC を 0x80 に設定してユニキャスト通信を行う（図 6 中の矢印 1）。点灯の通信の場合は EDT を 0x30 に、消灯の場合は 0x31 に設定し電文を構成する。この電文を受け取った照明機器は、EDT の値に従い動作を行い、状態変化の通知をマルチキャストで送る（図 6 中の矢印 2）。その際 SEOJ と DEOJ が入れ替わり、

	EHD	TID	SEOJ	DEOJ	ESV	OPC	EPC	PDC	EDT
書き込み要求	1081	0000	05FF01	029001	60	01	80	01	30
状態変化通知	1081	0000	029001	05FF01	73	01	80	01	30

図 5 照明の ON と OFF の操作で扱う電文プロトコルマップ

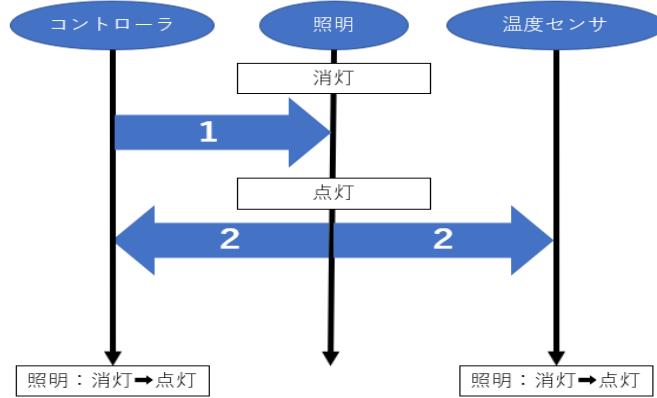


図 6 照明の ON と OFF の操作の通信のシーケンス図

通信の種類も通知に代わるため、ESV の値が 0x73 に変更される。状態変化通知のマルチキャストにより、ネットワーク内にコントローラが複数存在した場合に、各コントローラで照明機器の状態の認識に差異が生じることを防いでいる。また、この ON と OFF に関する通信は、送信先機器が何であってもすべての機器で備わっている機能であるので DEOJ の値に依存しない。

ii) 照明の調光の操作に関わる通信

照明の調光の操作に関わる通信の使用する電文を図 7 に示す。扱う機器が i) と同じであるため、SEOJ と DEOJ の関係性とシーケンスに関しては同様である。また、命令も書き込み要求であるため、ESV の値も同様の値を使う。調光機能は EPC を 0xC0 に指定する。EDT には設定したい明るさの値を 16 進数の 0x00～0xFF の 256 段階で指定する。照明の機器は、EDT の値に応じた出力を行う。また、i) と同様に書き込み要求の通信のうち状態変化の通知を行う。

iii) 温度センサの値の取得に関わる通信

温度センサの値の取得に関わる通信の使用する電文を図 8 に、シーケンスを図 9 に示す。温度センサの EOJ の値は、0x001101 であるため、DEOJ の値を 0x001101 に設定する。i) と ii) などのような制御ではなくパケットの種類が読み取り要求になるため、値を 0x62 に設定しユニキャスト通信をする。読み込み要求では OPC 以下の値に制限がないため、値は任意のものを入れて電文を構成する。読み取り要求を受けた機器は自分が取得したセンサの値を EDT に格納し、読み取り応答としてコントローラにユニキャストで通知する。

	EHD	TID	SEOJ	DEOJ	ESV	OPC	EPC	PDC	EDT
書き込み要求	1081	0000	05FF01	029001	60	01	C0	03	FFFFFF
状態変化通知	1081	0000	029001	05FF01	73	01	C0	03	FFFFFF

図 7 照明の調光の通信で扱う電文のプロトコルマップ

	EHD	TID	SEOJ	DEOJ	ESV	OPC	EPC	PDC	EDT
読み込み要求	1081	0000	05FF01	001101	62	01	E0	02	任意

図 8 温度センサの値の取得の通信で扱う電文のプロトコルマップ

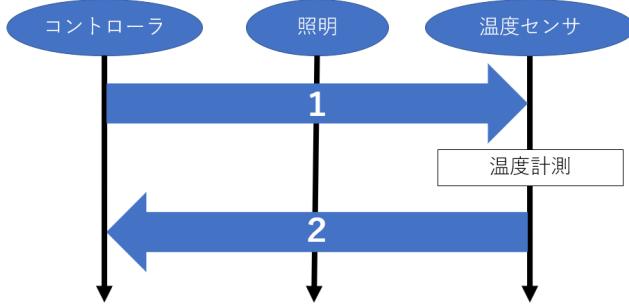


図 9 溫度センサの値の取得の通信のシーケンス図

5. エッジデバイスの開発

5.1 Arduino 互換通信基板の開発

エッジデバイスは、計測によるデータの取得や制御による動作が必要であるため、計測・制御教材を利用する。計測・制御教材は動作プログラムを変更することで、様々なセンサによる計測やモータ、LEDなどのアクチュエータの動作を実現できる。MESH[14]や Linking[15]などのような既製品の計測・制御教材では製作者が設定した動作時間や機能の中で必要なことを取捨選択しなければいけない。そこで技術科をはじめ、初步的な電子工作において広く利用されている Arduino を利用することで学習者が必要だと考えるセンサ値や動作などのさまざまな機能を実現できるようになると考えた。しかし、一般的に利用される Arduino UNO や Arduino MEGA では有線インターフェースしか用意されていない。そこで、エッジデバイスとして高橋らと同様に ESP-WROOM-02 モジュールを利用した Arduino 互換通信基板を製作し、開発を行った。外観を図 10 に、回路図を図 11、部品表を表 1 に示す。

ESP-WROOM-02 は、Arduino と互換性がある。そのため、計測・制御基板として動作が可能であり、加えて Wi-Fi モジュールを搭載しているため、センサ値などの取得データについての情報通信が可能である。3 章で述べたようにこの基板を使いアクチュエータ用基板とコントローラ用基板の開発を行う。Arduino 互換通信基板は、プログラムで動作を変更できるため、1 つの基板で必要に応じてアクチュエータ基板とコントローラ基板にもどちらにも変更が可能になっている。

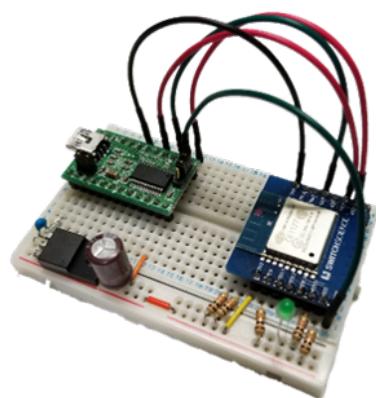


図 10 Arduino 互換通信基板の外観

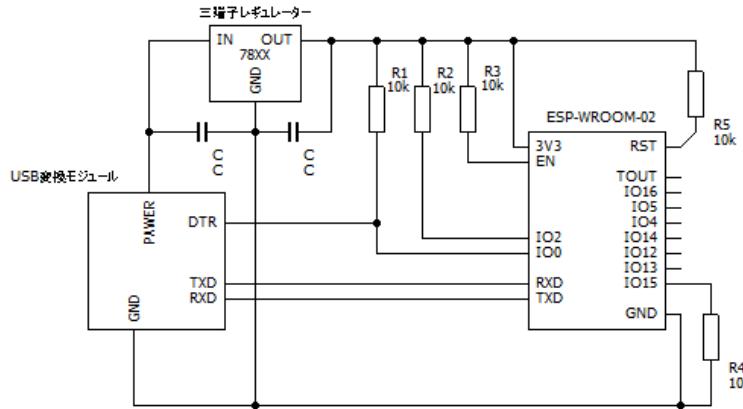


図 11 Arduino 互換通信基板の回路図

表 1 Arduino 互換通信基板部品一覧

部品名	型番	個数
制御基板	ESP-WROOM-02 モジュール	1 個
USB 変換モジュール	FT232RL	1 個
三端子レギュレーター	BA033CC0T	1 個
セラミックコンデンサ	0.1 μ F	1 個
炭素皮膜抵抗	10k Ω	5 個
ジャンパー線	オス-オス 10cm	5 本
ブレッドボード	SAD-101	1 個

Arduino 互換通信基板では 4/5/6/12/13/14/15 番の 7 ピンで PWM 出力とディジタル入出力が可能である。加えて A0(TOUT) ピンの 1 ピンでのみ最大 1V までのアナログ入力が可能である。この基板を中心に 4 章の i) ～ iii) を実行するためのアクチュエータ用基板およびコントローラ用基板を設計、開発した。

5.1.1 コントローラ用基板

アクチュエータ用基板を制御するためのコントローラ基板の外観を図 12、回路図を図 13、部品表を表 2 に示す。

タクトスイッチと可変抵抗を組み合わせて操作をする仕様にした。通信待機状態時に可変抵抗の値を常に可変抵抗の値を読み続け、タクトスイッチを押された時の抵抗値によって動作内容を変更することができる。読み取る値が 0-255 まで範囲があり、タクトスイッチを 3 つ使用しているため、理論上制御したいセンサ、アクチュエータの種類を 768 種類に分け、操作することが可能になる。そのため、学習者が必要と考える操作の数の上限はないに等しいと考える。

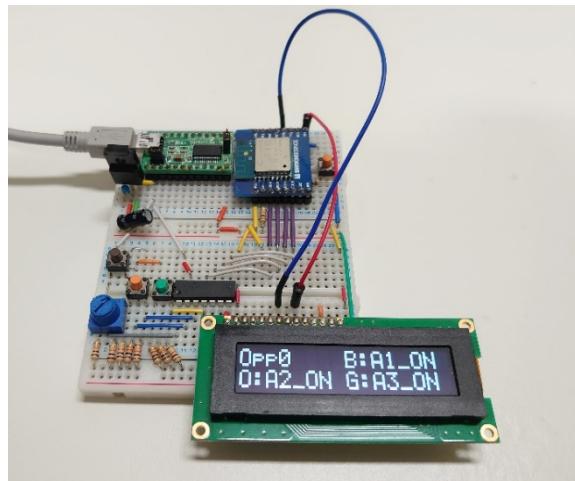


図 12 コントローラ用基板の外観

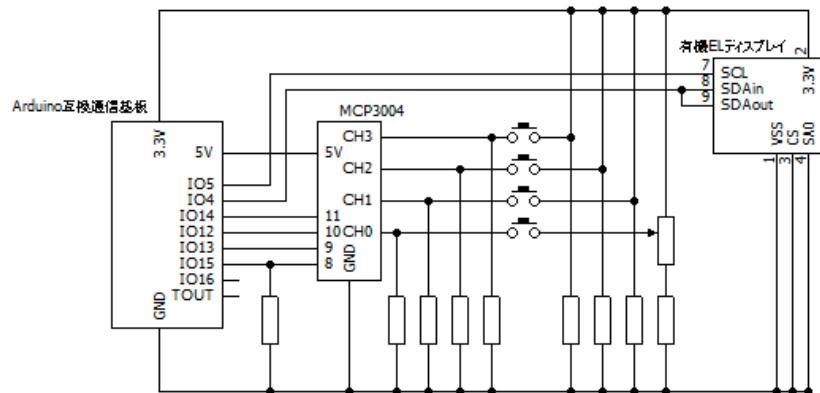


図 13 コントローラ用基板の回路図

表 2 コントローラ用基板の部品表

部品名	型番	個数
Arduino 互換通信基板		1 個
有機 EL キャラクタディスプレイモジュール	SO1602AWWB-UC-WB-U	1 個
10bit 4ch AD コンバータ	MCP3004-I/P	1 個
タクトスイッチ	TVDP01-6.5	3 個
単回転型サーメットトリマ	GE063P1KB103	1 個
炭素皮膜抵抗	10kΩ	4 個
	10Ω	4 個
ジャンパー線	オス-オス 10cm	19 本
ブレッドボード	SAD-101	1 個

5.1.2 アクチュエータ用基板

コントローラからの信号を受け、照明やソリッドステートリレーなどのアクチュエータや温度や照度など様々なセンサを動作させることができるアクチュエータ用基板を製作した。iii)を行うための温度センサを扱う際の外観を図 14、回路図を図 15、部品表を表 3 に示す。

この基板では、アクチュエータを PWM やデジタル出力により操作するほか、様々なセンサを取り扱うことを想定している。そのため最大 1V 入力の TOUT ピンが 1 ピンのみでは、汎用性や機能に不足が生じると考えた。そこで、12/13/14/15 番ピンに Microchip Technology 社製の AD コンバータ MCP3004[16]を繋ぎ SPI 通信を行うことで、最大 5V まで扱うことのできるアナログピンを 4 本追加した。これにより 12~15 番ピン以外のデジタル入力できるピンを含めると、合計 8 本のピンによるセンサ値の取得が可能である。この際 MCP3004 に接続しているピンは、出力ができないため、当該のピンでアクチュエータを利用するときは MCP3004 を取り外して利用する必要がある。LED を動作させている様子を図 16 に示す。

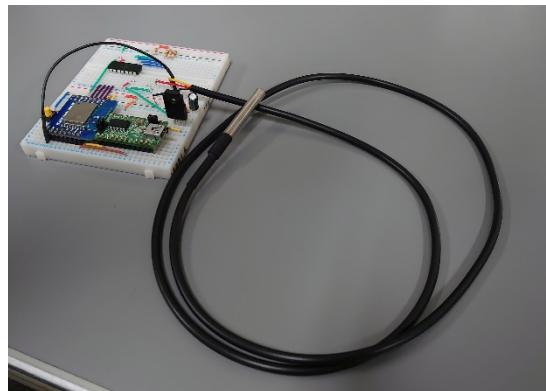


図 15 溫度センサを接続したアクチュエータ用基板の外観

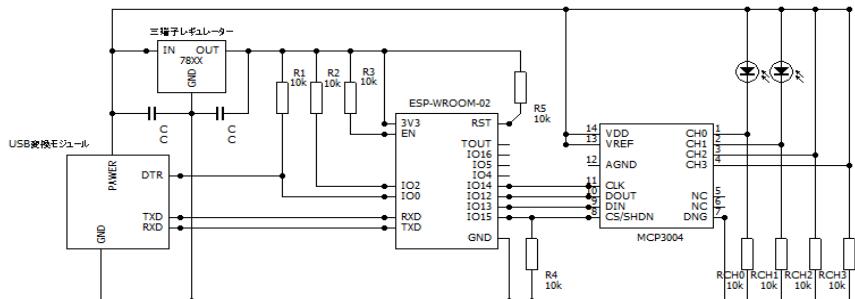


図 14 溫度センサを接続したアクチュエータ用基板の回路図

表 3 アクチュエータ用基板部品表

部品名	型番	個数
Arduino 互換通信基板		1 個
10bit 4ch AD コンバータ	MCP3004-I/P	1 個
温度センサ	DS18B20	1 個
炭素被膜抵抗	10kΩ	4 個
	10Ω	4 個
ジャンパー線	オス-オス 10cm	19 本
ブレッドボード	SAD-101	1 個

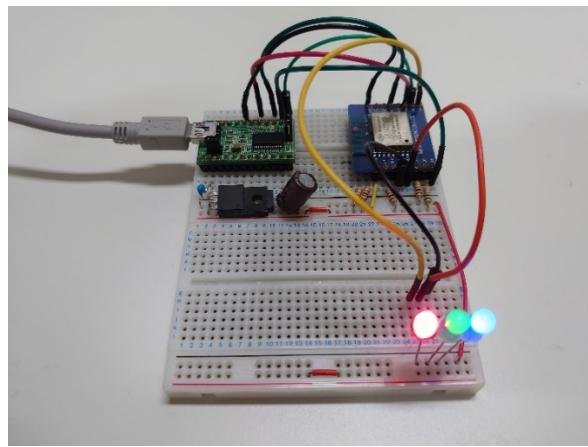


図 16 アクチュエータの動作時の様子

5. 2 プログラム

Arduino 互換通信基板を ECHONET Lite 対応にするためのプログラムを Arduino IDE で作成した。フローチャートを図 17 に示す。本研究のプログラムは大きく 2 部分に分かれており、他の機器からパケットを受け取った時に動作する部分(図 17 の A)とパケットを受け取るまでの待機している間に動作する部分(図 17 の B)に分かれている。図 17A では、まず受信したパケットを ECHONET Lite 電文に対応した「rBuffer [i] : (i は変数)」に格納する。i の値で電文の種類を指定する。電文のデータを対応する値を表 4 に示す。例えば、受け取ったパケットが 4.1 節で扱っている参加通知のものならば “rBuffer[10]=0x74” となる。

4. 2 節で挙げた i) ～iii) の例を実行するプログラムの作成をした。アクチュエータ用基板(機器①)、MCP3004 をつけたアクチュエータ用基板(機器②)、コントローラ用基板を用意し、機器①には i) と ii) のプログラムを書き込み、機器②には、iii) のプログラム、コントローラ用基板には、これらに対応するプログラムをそれぞれ書き込んだ。それぞれの機器に書き込むプログラムについて記述する。

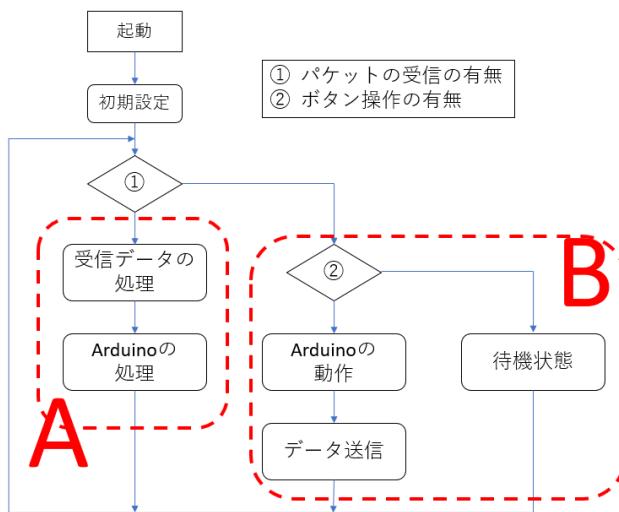


図 17 制作するプログラムのフローチャート

表 4 EL 電文と Buffer の対応表

電文	EHD		TID		SEOJ			DEOJ			ESV	OPC	EPC	PDC	EDT
sBuffer[]	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
rBuffer[]															14

まず、コントローラ用基板のプログラムについて記述する。5.1.1 節で示したようにコントローラ用基板のタクトスイッチと可変抵抗の操作をトリガーに行う。本研究では、可変抵抗から読み取る 0~1023 の値を 3 つの範囲に分けて、その範囲内で押されたボタンの色によって動作を分けた。ソースコードを表 5 に示す。可変抵抗の値によって変数 vv の値を変化させ、コントローラ用基板のスイッチを押したときの vv の値によって送る命令を変更する。この操作を図 15 の B で行う。 $vv=1$ のときには i) に関する命令を送り、 $vv=2$ のときに ii) に関する命令、 $vv=3$ のときに iii) に関する命令を送る。黒スイッチの操作のソースコードを表 6、緑スイッチの操作のソースコードを表 7 に示す。

$vv=1$ と 2 の操作では、書き込み要求を行うために ESV を 0x60 にし、 $vv=3$ では読み込み要求を行うために ESV を 0x62 に指定した。 $vv=2$ では、RGB3 色の照明に対しての命令を想定しているため、EDT を 3 バイトに増やして RGB それぞれに対して異なる値の制御を行う。これに伴い、EDT のバイト数を指定する PDC(sBuffer[13]) の値も 0x03 になっている。

アクチュエータ基板への書き込み要求と読み込み要求ではユニキャスト通信を行うため、通信先の指定が必要である。ECHONET Lite ではユニキャスト通信での通信先の指定を IP アドレスで行うが、Arduino 互換通信基板ではその指定を行うことが困難である。そこで、通信相手を特定するため、自由記述として使用が可能な TID の値を指定した。機器①の TID を 0x0001 とし、機器②の TID を 0x0002 とし、表 6 の 5、16、29 行目、表 7 の 5、16 行目でそれぞれの指定を行っている。

次に機器①のプログラムについて記述する。ソースコードを表 8 に示す。コントローラ用基板から送られたパケットを受け取り、TID が 0x0001 であることを switch 文で確認したら命令を実行する。機器①には R,G,B それぞれの LED を搭載し、操作を行う。コントローラ用基板からは機器①に対して EPC が 0x80 の電文か 0xC0 の電文が送信される。6 行目から 13 行目では EPC が 0x80 の場合の動作を示し、15-20 行目では EPC が 0xC0 の場合の動作を示している。16-18 行目と 23-25 行目で 0xC0 の場合に rBuffer[14]~[16] のそれぞれの値を直接 R,G,B の LED が output することを示している。機器①ではこれらを出力した後 13 行目の命令により状態変化通知を行う。

次に機器②のプログラムについて記述する。ソースコードを表 9 に示す。機器②と同様にコントローラ用基板から送られてきたパケットについて TID を確認する。受信後は 4 行目に示しているセンサにより計測を行う。計測した値は EDT に格納して電文に乗せるのだが、温度は小数点以下のデータも必要なため、計測した結果を整数部と小数部に分け、別データとして送信をする。(表 9 の 12-16 行目)

表 5 スイッチ操作のソースコード

```

1. int val = getSpiValue(0) ;
2. int bl = getSpiValue(1);
3. int og = getSpiValue(2);
4. int gr = getSpiValue(3);
5. int vv = 0;
6. if ( val < 340) {
7.   vv = 1;
8. } else if (val < 630) {
9.   vv = 2;
10. } else (val <= 1024) {
11.   vv = 3
12. }

```

表 6 黒スイッチの動作のソースコード

```

1. if (bl > 400) {
2.   switch (vv) {
3.     case 1:
4.       sBuffer[3] = 0x01;
5.       sBuffer[7] = 0x02;
6.       sBuffer[8] = 0x90;
7.       sBuffer[10] = 0x60;
8.       sBuffer[11] = 0x01;
9.       sBuffer[12] = 0x80;
10.      sBuffer[13] = 0x01;
11.      sBuffer[14] = 0x30;
12.      sendPacketSize = 15;
13.      break;
14.    case 2:
15.      sBuffer[3] = 0x01;
16.      sBuffer[7] = 0x02;
17.      sBuffer[8] = 0x90;
18.      sBuffer[10] = 0x60;
19.      sBuffer[12] = 0xC0;
20.      sBuffer[13] = 0x03;
21.      sBuffer[14] = 0x54;
22.      sBuffer[15] = 0xaa;
23.      sBuffer[16] = 0xff;
24.      sendPacketSize = 17
25.      break;
26.    case 3:
27.      sBuffer[3] = 0x02;
28.      sBuffer[7] = 0x00;
29.      sBuffer[8] = 0x0D;
30.      sBuffer[10] = 0x62;
31.      sBuffer[12] = 0xE0;
32.      sBuffer[13] = 0x02;
33.      sendPacketSize = 16;
34.      break;
35.    }
36.  }
37. }

```

表 7 緑スイッチの動作のソースコード

1. if (gr > 400) {	15. case 2:
2. switch (vv) {	16. sBuffer[3] = 0x01;
3.	17. sBuffer[7] = 0x02;
4. case 1:	18. sBuffer[8] = 0x90;
5. sBuffer[3] = 0x01;	19. sBuffer[10] = 0x60;
6. sBuffer[7] = 0x02;	20. sBuffer[12] = 0xC0;
7. sBuffer[8] = 0x90;	21. sBuffer[13] = 0x03;
8. sBuffer[10] = 0x60;	22. sBuffer[14] = 0xff;
9. sBuffer[11] = 0x01;	23. sBuffer[15] = 0xff;
10. sBuffer[12] = 0x80;	24. sBuffer[16] = 0xff;
11. sBuffer[13] = 0x01;	25. sendPacketSize = 17
12. sBuffer[14] = 0x31;	26. break;
13. sendPacketSize = 15;	27. }
14. break;	

表 8 機器①のソースコード

1. switch(rBuffer[3]) {	21. break;
2. case 0x01:	22. }
3. switch (rBuffer[ESV]) {	23. if (devStatus == 1) {
4. case SET_I:	24. analogWrite(lightRpin, RdimStatus);
5. case SET_C:	25. analogWrite(lightGpin, GdimStatus);
6. switch (rBuffer[EPC]) {	26. analogWrite(lightBpin, BdimStatus);
7. case 0x80:	27. sBuffer[14] = 0x30;
8. if (rBuffer[EDT] == 0x30) {	28. }
9. devStatus = 1;	29. else {
10. }	30. digitalWrite(lightRpin, LOW);
11. else if (rBuffer[EDT] == 0x31) {	31. digitalWrite(lightGpin, LOW);
12. devStatus = 0;	32. digitalWrite(lightBpin, LOW);
13. }	33. sBuffer[14] = 0x31;
14. echo.sendMulti(sBuffer, sendPacketSize);	34. }
15. break;	35.
16. case 0xC0:	36. sendPacketSize = 15;
17. RdimStatus = rBuffer[14];	37. break;
18. GdimStatus = rBuffer[15];	
19. BdimStatus = rBuffer[16];	
20. echo.sendMulti(sBuffer, sendPacketSize);	

表 9 機器②のソースコード

1. switch(rBuffer[3]) {	10. void temp_sensor() {
2. case 0x02:	11. sensors.requestTemperatures();
3. switch (rBuffer[ESV]) {	12. float tmp = sensors.getTempCByIndex(0);
4. case 0x62:	13. int tempI = sensors.getTempCByIndex(0);
5. temp_sensor();	14. int tempD = (tmp - tempI) * 100;
6. echo.sendMulti(sBuffer, sendPacketSize);	15. sBuffer[13] = 0x02;
7. break;	16. sBuffer[14] = tempI, HEX;
8. }	17. sBuffer[15] = tempD, HEX;
9. break;	18. sendPacketSize = 16;
	19. }

5. 3 動作検証

製作した基板が ECHONET Lite を利用して動作するかを検証した。コントローラ用基板 1 つ、アクチュエータ用基板 3 つ(機器 A、機器 B、機器 C)を用意し通信を行うか試験した。機器 B は、開発した基板以外との相互接続性を検証するため、市販されている ESP-WROOM-02 開発基板[17]を使用した。ユニキャスト通信の動作についても調べるため、アクチュエータ用基板の機器 A、B に LED を搭載し、機器 C は温度センサを搭載した。これら 3 つの機器を使い、正常に動作するかを確かめた。

この動作をコントローラ用基板から機器 A に向けて i の通信をした場合とコントローラ用基板から機器 C に向けて iii の通信をした場合の検証を行った。なお、i と ii は動作と通信の形式が同じため検証は i の場合のみ行う。今回の検証における正常な状態を以下に示したものとする。

- i) の通信を行うとき、
 - (ア) 機器 A の LED が点灯する
 - (イ) 機器 A の LED のみが変化し、機器 B の LED が反応しない
- iii) の通信を行うとき、上記の条件を満たした場合に得られると想定される結果を以下に示す。
 - (ウ) コントローラ用基板に温度が表示される
 - (エ) 取得した値が正確である

まず、アクチュエータの通信について、図 18 に命令前、図 19 に命令後の写真を示す。図 18において写真上部にある機器が機器 B で写真下部の左がコントローラ用基板、右が機器 A である。この教材は起動時 3 色の LED が全灯している状態になる。今回の検証では、機器 A に対し、図 18 のコントローラ用基板に表示されている 0x0080FF という値を送信し、青に対して 0x00、赤に対して 0x80、緑に対して 0xFF を送信し変化させる。図 19 の写真では、機器 A の点灯の仕方が変化し、機器 B は変化していない様子が分かる。よって、検証の条件であるアトイを満たしている。

次に、センサの通信について記述する。動作の様子を図 20 に示す。写真左がコントローラ用

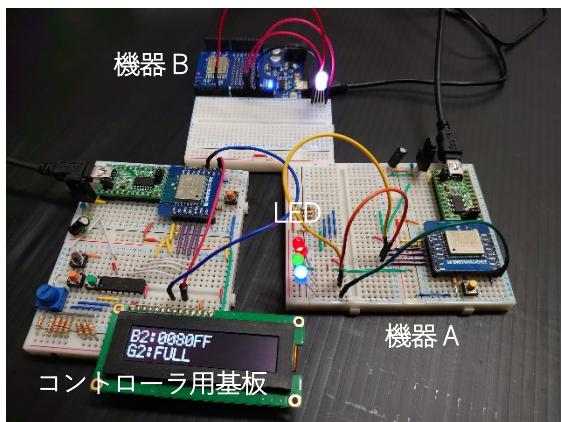


図 18 送信前の様子

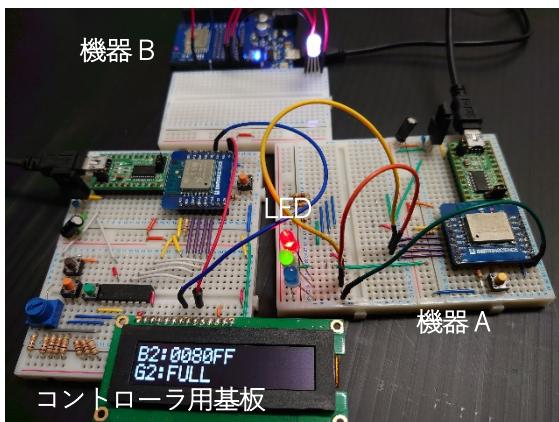


図 19 送信後の様子

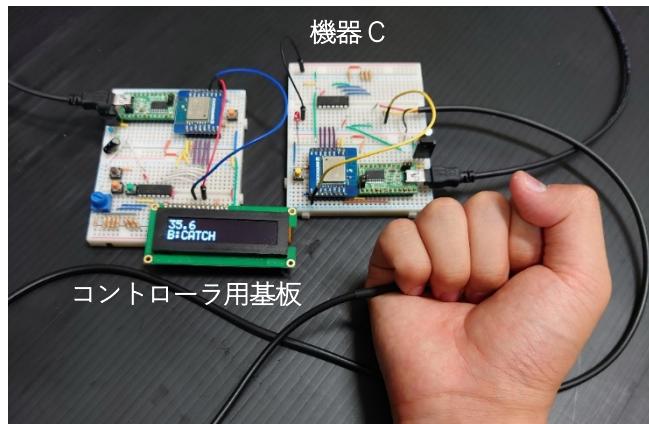


図 20 センサ取得の検証の様子

基板で右が温度センサを搭載した機器 C である。コントローラ用基板からの要求に対し、アクチュエータ用基板がセンサから取得した値をパケットに載せ、コントローラ用基板で表示している。コントローラ用基板の「35.6」はセンサから送信されてきた値である。手の温度を計測しているため、近い値が計測できていると判断できる。よって検証の条件のウとエを満たしている。

6. 考察

6. 1 本教材で想定する学習

本教材の学習としては、5.2 節に示すプログラムを学習者に例示し、コントローラ用基板からアクチュエータ用基板への 1 連の動作を構想し、機器の動作についてのプログラミングを行うことを想定している。学習者は例示する ECHONET Lite 電文の値について変更を施すだけでなく、受け取ったパケットの値に対応するセンサ・アクチュエータを Arduino 互換通信基板に接続し、計測・制御を行うプログラムを制作する。つまり、Arduino 互換通信基板の動作内容を指定する値の指定や識別を行う ECHONET Lite 電文に関する部分のプログラミングと、受け取った ECHONET Lite 電文の値に対して、実際にセンサやアクチュエータが動作するための Arduino 互換通信基板の部分に関するプログラミングのそれぞれを学習者は行う。

学習者は、表 6 の 5-12、16-24、29-34 行目、表 7 の 5-12、16-24 行目、表 9 の 15-17 行目のような実行する命令や状態変化の通知のための送信するパケットの値の指定と表 6 の 13、25、35

行目、表 7 の 13、25 行目のような通信のトリガーのタイミングについてプログラミングをする。また、表 8 の 1 行目、表 9 の 1 行目のような Switch 文や For 文の分岐命令を用いた通信相手先機器の指定、表 8 の 4、5、7、16 行目、表 9 の 4 行目のような受信したパケットの動作内容を指定する値の識別、表 8 の 8-13、17-19、24-26、30-32 行目、表 9 の 5 行目のような Arduino 互換通信基板の動作の 3 つについてもプログラミングを行う。本教材のプログラミングは、ECHONET Lite 通信の部分のパケットの値を用いた分岐命令と Arduino 互換通信基板の動作の部分の順次命令を利用しているため、フローチャートやシーケンス図等を利用し、通信の流れについて考えることで Arduino IDE を利用できる中学生であればプログラミングが可能であると考える。

しかし、プログラム作成時に 1 連の動作の中に ECHONET Lite 通信についてのプログラムと Arduino 互換通信基板の動作についてのプログラムが入り混じり、それぞれの動作が分かりにくいという課題がある。そこで、それぞれの動作について、ライブラリやサブルーチンとして関数化し、製作するフローチャートやシーケンス図等の流れにできるだけ近いかたちで命令をまとめられるようにすることでプログラミングの難易度が下がると考える。

6. 2 学習指導要領における考察

今回の指導要領の大きな変更として「D 情報の技術」の(2)の「ネットワークを利用した双方向性のあるコンテンツのプログラミングによる問題の解決」が挙げられる。2. 1 節でも述べたように IoT を教材として扱うことでこの変更の内容を満たすことができると考える。また、ネットワークに関しても指導要領解説では「利用するネットワークは、インターネットに限らず -中略- 特定の場所だけで通信できるネットワークも考えられる[7]」としていることから、HAN 内でしか利用できない ECHONET Lite でも Wi-Fi ルータを起動し、通信できれば、この内容を満たす学習が可能であると考える。また、(2)のアについても「情報ネットワークの構成、サーバやルータ等の働きや、パケット通信」などの理解についても記述されており、ECHONET Lite の通信方式について理解することで書き込み要求での制御や読み込み要求での計測の通信で値の変換について学ぶことが可能であると考える。この学習ができることで「D 情報の技術」(1)アについて、指導要領解説で示されている「センサなどの入力から、アクチュエータ等の出力装置までの信号の伝達経路や変換の方法」や「構成する要素や装置を結合して機能させるシステム化の方法等[18]」についての理解につながると考える。

さらに、本研究では、基板として Arduino 互換通信基板を使用している。この基板に 5 章で示しているようなプログラミングを行うことで、計測・制御のプログラミングと ECHONET Lite の通信を行うことが可能な教材になった。これにより、指導要領解説の D 情報の技術の内容に示されているネットワークの利用及び双方向性の追加の規定の追加と計測・制御システムの構想についても満たしたと考える[19]。ネットワークの利用及び双方向性の追加に関しては上記に示したとおりである。計測・制御システムの構想に関しては基板によりアナログ・デジタル信号でのセンサ値の獲得や LED の出力などを行うことができる。その値を ECHONET Lite の電文に載せ、基板間で通信を行い情報の共有が行えること、また、送受信に関するプログラミングを行うことで計測・制御システムの仕組みについての学習ができるため、(3) アの「計測・制御システムの仕組みを理解し、安全・適切なプログラムの制作、動作の確認及びデバッグ等ができること」を満たす教材であると考える。

このように「D 情報の技術」において指導要領の内容について記述を満たす項目が多く含まれることから、幅広く学習できる教材になっていると考える。また、HEMS の標準プロトコルになっている ECHONET Lite を扱い、スマートハウスのシステムや HEMS について扱い、各家庭での消費エネルギーや家電製品の制御にも触れることで「D 情報の技術」の内容だけでなく、「C エネルギー変換の技術」の(2)イについても触れることができ、学習指導要領において汎用性の高い教材であると言える。

6. 3 学習の3本の柱における考察

学習指導要領では、子供たちが求められる資質・能力として「知識・技能」「思考力・判断力・表現力等」「学びに向かう力・人間性等」が3つの柱として整理されている。教育課程を通してこの資質・能力を身に付け、生涯にわたり能動的に学び続けることができる人材を育成できることが目指すため、主体的・対話的で深い学びの実現に向けた授業の改善が求められている[20]。この主体的・対話的で深い学びの実現のために、奈須は「有意味学習」「オーセンティックな(真正の、本物の)学習」「明示的な指導」の三つの原理を取り入れることが重要であるとしている[21]。

有意味学習は、子供が生活の中や今までの学習の中で獲得した既存の知識について扱い、不正確で断片的な知識について洗練、統合していくという生徒自身の生活に密接した知識を獲得する学習であるとしている。本研究の教材は、ECHONET Lite を利用することで生徒の家庭内にある機器と接続できる教材になっている。学校で学習した内容を実生活に還元し、自身の生活について見直すどこができる可能性のあることから有意味学習につながる教材であると考える。

オーセンティックな学習は、具体的なシチュエーションを多く含み、本物の社会実践の参画の可能性を秘めた学習である。本教材では、様々なセンサ、アクチュエータを搭載可能な基板を利用しているため、生活のあらゆる計測、制御を行うことができ、学習した内容をほとんど変更することができなく、生徒の生活にそのまま生かすことができる。そのため、より本番に近い学習を行い、学習内容をそのまま生かすことができるオーセンティックな教材であると考える。

明示的な指導では、学習内容にバラバラに存在する要素を汎用的で有用性の高い教材を使って生徒たちに統合的概念にさせる指導を目指すべきとされている。つまり、汎用的で有用的な可能性を持つ教材で具体的な例示を示し、生徒に変更や改良を加えさせ、その教材の学習を深めさせる指導を行うことであると考える。本教材では、Arduino 互換通信基板を ECHONET Lite 通信でデータのやり取りをさせる例示を示すことで、取り扱う電文に変更を加えたり、Arduino 互換通信基板の制御のプログラムの変更を行ったりすることができる。そして、IoT のシステムにはデータの通信の部分と具体的に動作する機械の部分があり、これらの要素が連動することでシステムとして機能することができるという IoT に対する統合的概念の形成を促すことが可能である。

以上のことから本教材は、奈須の示す3つの原理を有する教材であり、学習指導要領の示す主体的・対話的で深い学びを実現可能な教材であると考えた。

7. 終わりに

本研究では、平成29年に告示された学習指導要領技術科の「D 情報の技術」における教材

として、IoT を学習するための教材の提案を行った。提案する教材の要素を IoT の定義にないエッジデバイス、ゲートウェイ、サーバとし、学習者が 3 つの要素について理解しやすくするため、それぞれの要素の役割を担う機器を使い分けて学習を行うことを提案し、本論文ではエッジデバイスについて開発を行った。

エッジデバイスの機器に Arduino 互換通信基板を利用し、エッジデバイスの通信に HEMS の標準プロトコルである ECHONET Lite を利用して、LED の照明の制御やセンサ値の取得ができる教材を開発した。

この教材を利用することについて学習指導要領解説に記載されている内容について考察し、「D 情報の技術」の(2)(3)についての内容を取り扱うことのできる教材であると考えた。また、「D 情報の技術」の内容だけでなく、「C エネルギー変換の技術」についての内容も触れることのできる汎用性の高い教材であり、実際の社会で使われているプロトコルを扱うことで有用性も兼ね備わっている教材であることから、学習指導要領が示す「主体的・対話的で深い学び」の実現につながる教材であると考えた。

参考文献

- [1] 八子知礼 : IoT の基本・仕組み・重要事項が全部わかる教科書, p.2, ,SB クリエイティブ, 2017
- [2] 内閣府 :『Society 5.0』, https://www8.cao.go.jp/cstp/society5_0/index.html (最終アクセス日 : 2019/08/08)
- [3] 総務省 : IoT デバイスの急速な普及, <http://www.soumu.go.jp/johotsusintoeki/whitepaper/ja/h30/html/nd111200.html> (最終アクセス日 : 2019/09/02)
- [4] 下島健彦 : IoT 開発スタートブック ESP32 でクラウドに繋げる電子工作を始めよう, p13, 技術評論社, 2019
- [5] Panasonic : スマート家電, <https://panasonic.jp/pss/> (最終アクセス日 : 2019/09/02)
- [6] 文部科学省 : 中学校学習指導要領(平成 29 年告示), pp.132-136, 2018
- [7] 文部科学省 : 中学校学習指導要領(平成 29 年告示)解説 技術・家庭編, p53, 2018
- [8] 文部科学省 : 高等学校学習指導要領(平成 30 年告示)解説 情報編, p20, 2018
- [9] 文部科学省 : 高等学校学習指導要領(平成 30 年告示)解説 情報編, p22, 2018
- [10] 文部科学省 : 高等学校学習指導要領(平成 30 年告示), p.255, 2018
- [11] 高橋等・永田奈央美 : IoT 技術の活用を目指したプログラミング教材の研究, 静岡産業大学情報学部研究紀要 21 号, pp.217-226, 2019
- [12] 辻明典・川上博 : 次世代エンジニアを育成する IoT 学習教材の開発と実践, 徳島大学大学開放実践センター紀要, 第 26 卷, pp.19-26, 2017
- [13] ECHONET CONSORTIUM : エコーネットとは? <https://echonet.jp/about/> (最終アクセス日 : 2019/09/03)
- [14] SONY corporation : MESH, <http://meshprj.com/jp/> (最終アクセス日 : 2019/09/09)
- [15] Project Linking : Project Linking, <https://linkingiot.com/index.html> (最終アクセス日 : 2019/09/09)
- [16] 秋月電子通称 : 10bit 4ch AD コンバータ MCP3004-I/P, <http://akizukidenshi.com/catalog/g/gI-11987/> (最終アクセス日 : 2019/10/30)

- [17] SWITCH SCIENCE : ESPr® One (Arduino Uno 同一形状 ESP-WROOM-02 開発ボード) ,
<https://www.switch-science.com/catalog/2620/> (最終アクセス日 : 2019/11/15)
- [18] 文部科学省 : 中学校学習指導要領(平成 29 年告示)解説 技術・家庭編, pp.50-51, 2018
- [19] 文部科学省 : 中学校学習指導要領(平成 29 年告示)解説 技術・家庭編, pp.49, 2018
- [20] 文部科学省 : 中学校学習指導要領(平成 29 年告示)解説 技術・家庭編, pp.3-4, 2018
- [21] 奈須正裕 : 「資質・能力」と学びのメカニズム, pp.141-207, 東洋館出版社, 2017