

Web 技術を用いた IoT デバイスの画像による目視確認の試み

水矢 亨(企画部新事業戦略グループ)

1. はじめに

最近では、デジタル技術の利活用といえば DX が注目を集めているが、DX 白書 2023⁽¹⁾によれば、IoT は DX で利活用されるデータの獲得手段と位置づけられる。また同白書では、IoT はセンサー・無線/有線通信・IoT ゲートウェイ・IoT プラットフォームなど、複数の構成要素からなる複合的技術であり、その技術的な複雑さが IoT 利用の難しさが IoT 導入・運用上の障害として顕在化する可能性を指摘している⁽¹⁾。

KISTEC では、実際に手を動かして複合的技術である IoT の基本的な部分を体感してもらうための実習を重視しながら、IoT 分野の人材育成に継続的に取り組んできた。その一環として、リモートワークの普及などの情勢をふまえ、令和 4 年度には「Web 技術を活用した IoT 実習のオンライン化」に取り組んだところ、「実機の様子を画像により遠隔からも確認したい」というニーズが明らかになった⁽²⁾。

これに対応するため、Web ブラウザから IoT デバイスの画像による動作状況の目視確認を目的として、Web 技術 (WebSocket) を用いて IoT デバイス側から Web ブラウザに画像を伝送する簡易システムの試作を行った。

2. IoT 分野の人材育成での Web 技術の活用

IoT は前述のように複合的技術であり、その習得のためには IoT デバイスの実機を使った実習が有効である。しかし、IoT デバイスを実際に動かしてみるには、そのための (ソフトウェア、配線や通信等の) 環境構築が必要であり、学習者にとって最初のハードルとなっている。

そこで、KISTEC では予め環境設定を済ませた IoT デバイス (Raspberry Pi) を用意し、Web アプリケーションを利用して遠隔から IoT デバイスを利用する形態により、実習のオンライン化 (図 1) に取り組んできた。実習用のツールとして Web アプリケーション (図 2) を用いることで、受講者側 (Web ブラウザ以外の) ソフトウェアのインストール等の作業が発生しないようにしている。

IoT デバイスが Web システムやクラウドとデータ通信するには、IAF (Industrial Automation Forum) の ia-cloud プロジェクト⁽³⁾が指摘する (図 3) ように、REST API (JSON データ利用) や WebSocket といった Web 技術が必要となる。

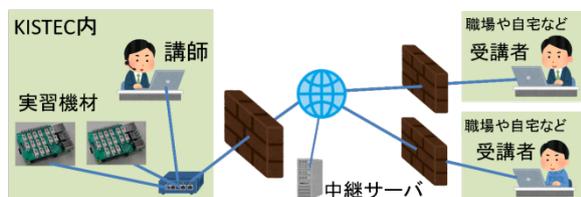


図1 オンラインでのIoT実習のイメージ(2)

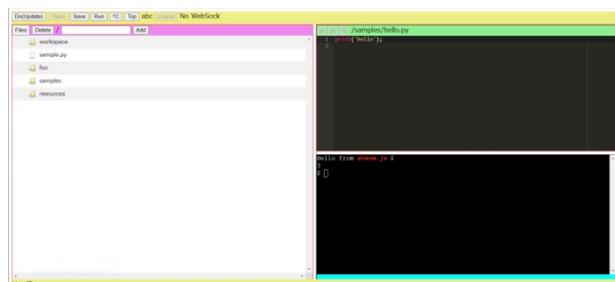


図2 実習用ツール (Web アプリケーション) の画面(2)

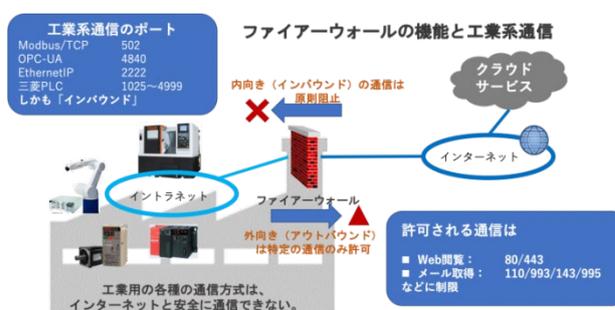


図3 工業系通信とファイアウォールの関係(3)

WebSocket は FRC 6455 (2011 年)⁽⁴⁾で標準化されており、Web ブラウザとサーバ間での双方向通信を可能とする。

前述の IoT 実習用ツールにおいても、Web ブラウザとサーバの双方向通信が実現できる WebSocket を利用している。WebSocket 通信により、中継サーバを介して受講者が各自に割り当てられたリモートの実機にアクセスし、プログラムの実行や停止などの操作をすることができる。また、プログラムの標準出力など文字情報による確認も可能であったが、LED の点灯など「実機の様子を画像により遠隔からも確認したい」との要望があり、今後の機能追加が求められている。

3. WebSocket による画像伝送の実装と評価

IoT デバイス側から Web ブラウザに画像を伝送する簡易システムの概略を図 4 に示す。Web ブラウザ、中継サーバ、IoT デバイス (Raspberry Pi 4B) が主な構成要素である。また、Raspberry Pi 4B (と USB カメラ) でキャプチャーした画像を、WebSocket 通信で中継サーバを経由して Web ブラウザに送信するのが、本簡易システムの基本機能である。

IoT 実習用ツールへの活用を想定していることもあり、画像は JPEG 形式のバイナリデータを Base64 でアスキー文字からなる文字列に変換したうえで JSON の値に含めて WebSocket で伝送した。

Raspberry Pi 4B (Python は 3.7.3) では、USB 接続した

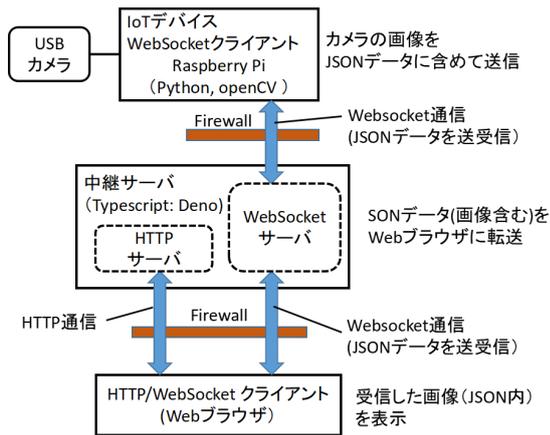


図4 簡易システムの概略

Web カメラの画像キャプチャーに openCV (4.1.0.25)を、WebSocket 通信には websocket-client (1.6.1)を用いた。中継サーバは WebSocket サーバと Web サーバを兼ねており、Web ブラウザに表示する HTML ファイルも送信する。中継サーバには Deno(1.32.3)を利用し、Deno が標準で提供する HTTP と WebSocket の通信機能を用いた。Web アプリケーションの構築には Hono(3.4.3)を用いた。Web ブラウザ側 (Core i5-8265, メモリ 16GB で Windows 10 の PC 上の Chrome)では、Web 標準である WebSocket API を用いた。

簡易システムの評価のため、160×120(QQVGA), 320×240(QVGA), 640×480(VGA)の各サイズの画像を送信間隔 (送信処理間の停止時間) は 100[ms], 200[ms]で 300 回送信を行い、IoT デバイス側で Web ブラウザからの応答を受信するまでの時間を応答時間として計測した。図 5 は、受信した QVGA 画像を Web ブラウザで表示した例である。

遅延時間の計測結果を図 6 及び図 7 に示す。なお図中のエラーバーは標準偏差を示している。送信間隔 100ms の場合は、各サイズで平均応答時間が 100ms を超えており、応答の受信前に次の画像を送信するケースが多いと考えられる。送信間隔が 200ms の場合は、QQVGA と QVGA では平均応答時間が送信間隔の 200ms を下回っており、応答受信後に次の画像を送信しているケースが多いと考えられる。なお、JPEG データを Base64 での変換で得られた文字列は、QQVGA で約 13000 文字、QVGA で約 33000 文字、VGA で約 92000 文字であった。

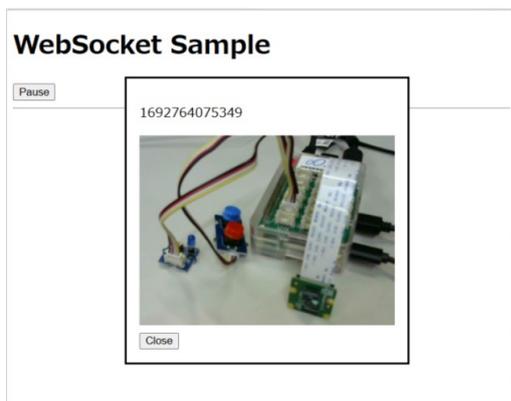


図5 受信した画像(320×240)のブラウザでの表示

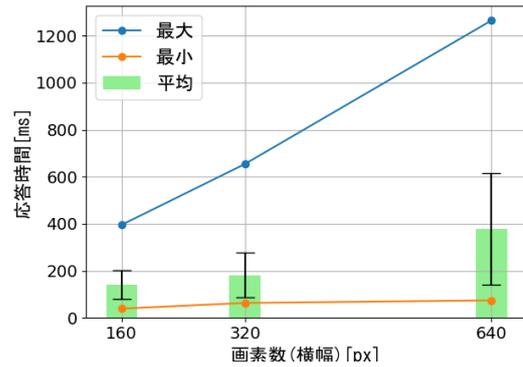


図6 送信間隔 100ms の場合の応答時間

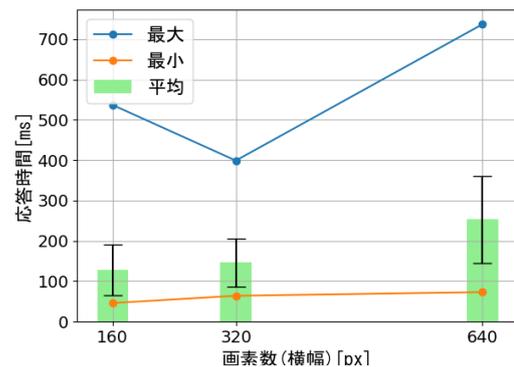


図7 送信間隔 200ms の場合の応答時間

4. 考察及び今後の課題

図 6 及び図 7 に示した結果から、QVGA(320×240)の画像であれば、5FPS 程度の送信が可能であることが分かった。5FPS であれば、LED の 0.5 秒間隔と 1 秒間隔の点滅の違いを認識でき、IoT 実習の用途で利用可能と考えられる。ただし、実際に目視で確認したところ画像の表示には 1 秒程度の遅延が生じており、Web ブラウザでの描画時間も注意する必要がある。また、実用の際には WebSocket と TLS による暗号化と合わせて利用することが想定され、その処理時間も考慮する必要がある。

今後は、IoT 実習用ツールに WebSocket による画像伝送機能を追加する計画である。IoT 実習用ツールに追加実装し、Raspberry Pi と Web ブラウザの組が複数ある場合の画像伝送の評価を行うことは実用上も重要と考えられる。

【謝辞】「Web 技術を活用した IoT 実習のオンライン化の取組み」は公益財団法人 JKA の補助を受けて実施しました。

【参考文献】

1. 情報処理推進機構：DX 白書 2023(2023).
2. 神奈川県立産業技術総合研究所：「Web 技術を活用した IoT 実習のオンライン化の取組み」, オンライン IoT 実習を含む IoT 技術の人材育成フォーラム事業成果報告書 (2023).
3. ia-cloud プロジェクト：ia-cloud の概要
<https://ia-cloud.com/about> (2023.08.18 閲覧)
4. IETF: RFC 6455 “The WebSocket Protocol”(2011).