

IoTシステムにおけるゲートウェイのソフトウェアテスト自動化とその効果検証 ～AWS IoT Greengrassを用いた対向装置環境構築の簡易化～

○大塚 貴丈 尾上 圭介 梶山 剛 浮田 陽介

パナソニック株式会社 エレクトリックワークス社 ソリューション開発本部

はじめに・背景

「クラウド」－「IoTゲートウェイ」－「センサ、制御対象機器」で構成されるIoTシステムでは、バージョンアップに応じてソフトウェアテストを繰り返しておりテストの工数が大きい。その中でもIoTゲートウェイの結合テストでは、センサや制御対象機器等の配下機器が多いため、それらの通信を模擬する対向装置の開発や環境構築、対向装置設定を含めたテスト実行の工数が課題となる。本稿ではAWS IoT Greengrassを用いて対向装置環境構築の簡易化を行い、各対向装置設定を含めたテスト自動化試作と効果検証を行った結果を紹介する。

概要

IoTシステムにおけるIoTゲートウェイの結合テストの内、通信のフォーマット（型・範囲など）を確認するフォーマットテストや判断ロジックによるデータ出力内容が期待値と一致しているか確認するロジックテストの自動化実現に向けた検討を行った。AWS IoT Greengrass^[1]による対向装置ソフトの配置によって対向装置環境構築の工数を削減し、AWSクラウドサービス^[2]を用いた自動化システムにより対向装置の通信応答内容変更を含めたテスト自動実行環境を構築することによりテスト全体工数の削減を図った。自動化有無による作業時間を比較することによりテスト工数削減効果を検証し、自動化適用可能なテスト項目では60%程度工数削減できることを確認した。

問題提起

IoTシステムにおけるIoTゲートウェイは、配下に繋がる機器の追加等のバージョンアップに応じてソフトウェアテストを繰り返す必要があり工数がかかる。そのため、ソフトウェアテストを自動化することでテスト工数が削減でき、商品リリースサイクルも早められると考えられる。

ここでは、IoTシステムは「クラウド」－「IoTゲートウェイ」－「センサ、制御対象機器」という構成を想定する。センサから発信されたデータはIoTゲートウェイで受信して定期的にクラウドに送信され、クラウドから送信されたコマンドをIoTゲートウェイが制御対象機器のコマンドに変換して制御するという一般的な構成であるものとする。

個々の問題に対する課題

■課題①

IoTゲートウェイのソフトウェアテスト（結合テスト）では対向装置を用いたテストが必要であり環境構築に時間がかかる。

■課題②

対向装置は通信対象のセンサや制御対象機器等のサブシステム毎に多種用意する必要があり、開発工数がかかる。

■課題③

対向装置の設定を含む複雑な実行手順をリグレッションテスト等で何度も実行が必要となる。

個々の課題に対する対策

■課題①に対する対策

- 対向装置ソフトをモジュール化して配置することで対向装置環境構築を簡易にする
センサの通信模擬や制御対象機器の通信応答模擬等を行う対向装置として機能するソフトウェアを作成し、IoTゲートウェイと同じハードウェア等の各通信機能を有するハードウェア上で対向装置ソフトウェアもモジュールとして配置することで、対向装置毎に環境構築する工数を削減する。
- AWS IoT Greengrassによりモジュール配置を簡易化する
AWS IoT Greengrassはエッジデバイスのソフトウェアを構築、配置、管理するためのクラウドサービスである。本サービスにより、クラウドに保存した各ソフトウェアモジュールを任意のエッジデバイスに配置可能となる。本サービスを活用し、テストに必要な対向装置のソフトウェアモジュールを選択して対向装置デバイスに簡単に配置可能となる。

■課題②に対する対策

- AWS IoT Greengrassにより実行環境を共通化することで対向装置ソフト開発にセンサ・制御対象機器との通信ソフトを流用し、開発工数を削減する
AWS IoT Greengrassにより実行環境をIoTゲートウェイと対向装置環境とで共通化し、対向装置ソフト開発時にIoTゲートウェイにおける各機器との通信機能の実装部分を流用することで、各機器の通信模擬を行う対向装置の開発工数を削減する。

■課題③に対する対策

- クラウドサービスを使ったテスト自動化システムを構築し、テストを自動実行する
テスト自動化システムとして、テスト実行に必要な制御コマンド送信を行う「テストシナリオ実行機

能」，テストの確認対象となるセンサデータや制御応答・結果を収集する「通信内容収集機能」，収集したデータを期待結果と比較してOK/NGを判定して集約する「テスト判定・結果集約機能」が必要となる。これらはAWSクラウドサービスのサーバレス関数実行サービスであるLambdaやデータベースサービスのDynamoDBを用いて実現した。

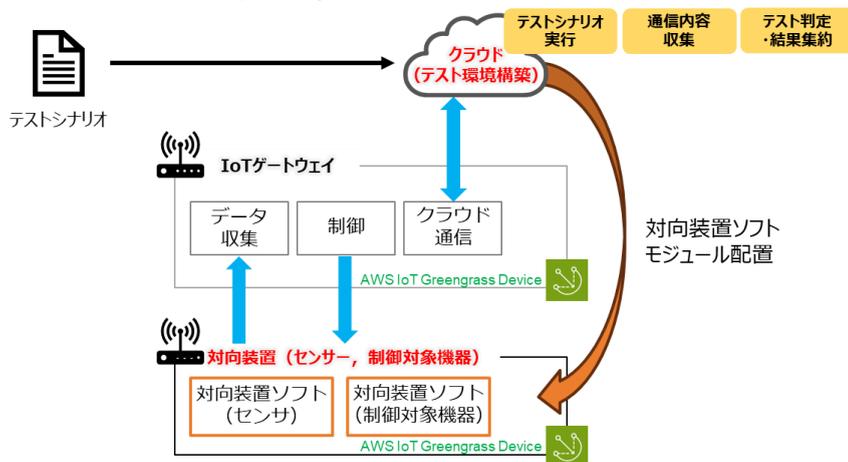


図1 ソフトウェアテスト自動化システム概要

結果・まとめ

従来のテスト自動化しない場合と自動化した場合でのテスト実施時間を比較することにより、工数の削減効果を検証した。提案するテスト自動化システムにより、対向装置環境構築作業やテスト実行、データ比較、結果集計の作業工数を削減できる。ただし、初回実施時はテスト結果比較に必要な期待結果である正解データの作成作業が必要となるが、正解データ作成作業は一度作成しておけば回帰テストやリグレッションテスト等の同じテストを複数繰り返す際は不要となる。そのため、回帰テスト等の実行時にテスト工数削減効果が大きくなると考えられる。この検証の結果、初回テスト、回帰テストを合わせて実施時間を57%削減することができることを確認した。

今後の課題としては、シーケンシャルなテスト手順が必要なテストへの拡張や、対向装置に配置するソフトの拡充（＝対向装置の通信種類の対応拡充）による汎用性の向上が挙げられる。

参考文献

- [1] Amazon Web Services, Inc., “AWS（アマゾン ウェブ サービス）とは?,” [オンライン]. Available: <https://aws.amazon.com/jp/what-is-aws/>. [アクセス日: 14 5 2024].
- [2] Amazon Web Services, Inc., “AWS IoT Greengrass とは,” [オンライン]. Available: https://docs.aws.amazon.com/ja_jp/greengrass/v2/developerguide/what-is-iot-greengrass.html. [アクセス日: 14 5 2024].