REALIZATION OF PHYSICAL ENVIRONMENT WITH DIGITAL TWIN AND ANALYSIS

Project report submitted

in partial fulfillment of the requirement for the degree of

Bachelor of Technology

By

Golla Venkata Sai Bhargav(18BEC013)

Gurijala Sai Harsha(18BEC016)

Kukkala Bharath(18BEC023)

Peddoju Sathvik(18BEC038)

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DHARWAD

CERTIFICATE

It is certified that the work contained in the project report titled "Realization of physical environment with Digital Twin and analysis", by "Golla Venkata Sai Bhargav (18BEC013)", "Gurijala Sai Harsha (18BEC016)", "Kukkala Bharath (18BEC023)" and "Peddoju Sathvik (18BEC038)" has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

Dr. Jagadeesha R Bhat
Department of Electronics and Communication Engineering
(May, 2022)

Declaration

We declare that this written submission represents my ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Golla Venkata Sai Bhargav
(18BEC013)

Gurijala Sai Harsha
(18BEC016)

Kukkala Bharath
(18BEC023)

Peddoju Sathvik
(18BEC038)

Approval Sheet

This project report entitled **Realization of physical environment with Digital Twin and analysis** by Golla Venkata Sai Bhargav, Gurijala Sai Harsha, Kukkala Bharath, Peddoju Sathvik is approved for the degree of Bachelor of Technology in Electronics and Communication Engineering.

Supervisor
Dr. Jagadeesha R Bhat

Head of the Department
Dr. K T Deepak

Examiners

DATE: **May, 2022**

PLACE: **Dharwad, India**

Acknowledgement

We would like to express our sincere gratitude to our supervisor, Dr. Jagadeesha R Bhat, for his guidance, immense patience, insightful comments and also for giving us this wonderful opportunity to work on this project which helped us in learning and exploring many new applications and technologies.

Date: May, 2022

Place: Dharwad

Golla Venkata Sai Bhargav, 18BEC013
Gurijala Sai Harsha, 18BEC016
Kukkala Bharath, 18BEC023
Peddoju Sathvik, 18BEC038
Department of ECE
IIIT Dharwad

ABSTRACT

Digital twin is the most advanced and evolving technological solution for complex engineering models. This project gives you a basic understanding on Digital twin and explains to you how this technology works. This is a problem solver for most of the complex engineering models these days even NASA uses this technology for their complex designs. Nowadays many of the industries are adopting this application which significantly increases their success rate, effective time management and cost reduction. This project gives a detailed procedure to how to build digital twin and how to connect physical and virtual systems using Digital twin applications. Our design consists of two parts one is a physical system built on a controller NODEMCU and the other is Digital twin which has been created on AZURE IOT Platform. In this project we aim to create a Digital Twin of a WiFi controlled Car which sends sensor datas of the temperature and humidity of the surrounding environment and also the tilt angles of the car which can be viewed from the Azure Digital Twin platform.

Content

1. INTRODUCTION

The term '**Digital twin**' was first used and applied by Michael Grieves in 2002.He proposed the digital twin as the conceptual model underlying product lifecycle management (PLM). Then onwards It was adopted by NASA in their projects as it increased their success rate of projects. Later on it was most reliable and used concepts in industry level for complex designs such as bridges, aircrafts through which we can expose our system to the harsh environment and get feedback from Digital twin which saves us a lot of money and hard labor of construction.

Digital twin is a virtual replica of a physical object. This virtual twin exists only as software rendered by computing power. For the most part, this definition is reasonably adequate. Having a digital replica of a physical object can significantly improve on one or more of the following processes, design, simulation, planning, building, operating, maintaining, optimizing, and disposal.

1.1 DEFINITION

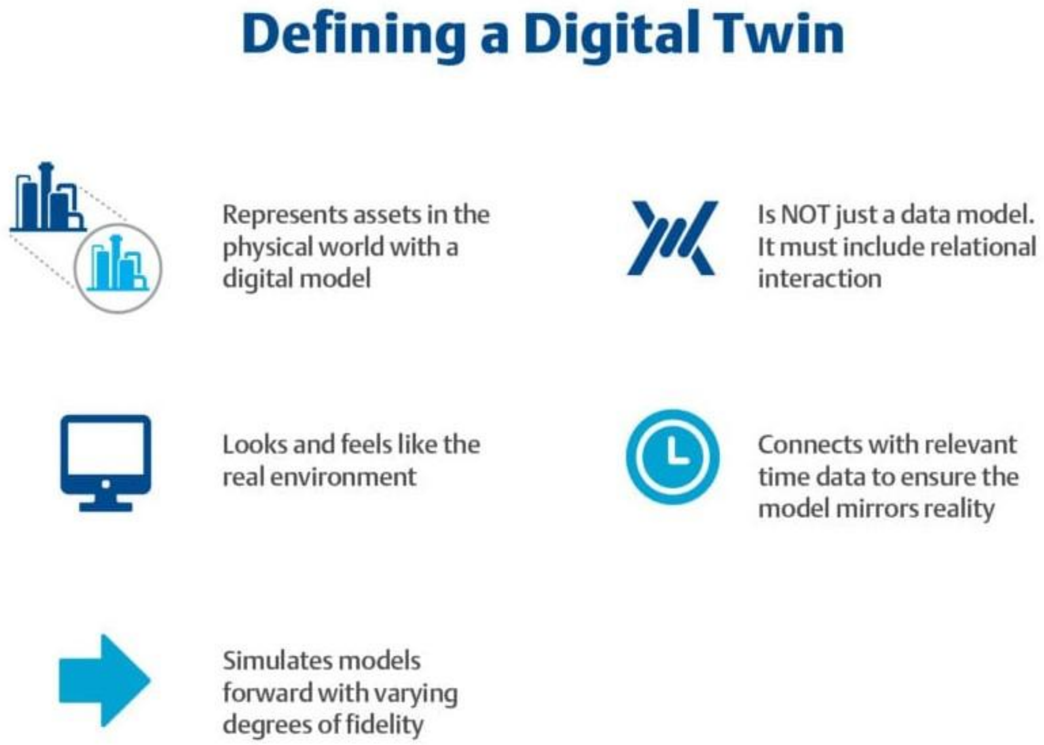


Fig. 1 Defining a Digital Twin

In Fig.1 Describes the Digital twin in 5 symbols which represent 5 properties of digital twin which are simulate,connect,interaction,real environment which collectively represent real world in virtual environment.

- Digital twin is the ability to take a virtual representation of elements and the dynamics of how an internet of things device operates and works.
- It's more than a blueprint it's more than a schematic it's not just a picture it's a lot more than how it looks and sounds it's a virtual representation of both elements and dynamics of how an IOT device operates and lives through its lifecycle.
- It's understanding all of its dynamics whether it's electrons that move or whether it's the device that's moving itself, it's about understanding the elements that compose it and dynamics of how the device is put together.

- If it's done correctly digital twin will influence how design builds and operations of a device are constructed in a single life cycle.

- The digital twin concept consists of three distinct parts: the physical product, the digital/virtual product, and connections between the two products.

- The connection between digital and physical product is bi-directional data from the physical system to the virtual model and vice-versa.

- A physical system can have multiple virtual twin depending on user usage

1.2 ADVANTAGES OF DIGITAL TWIN

Fig. 2 Advantages from a Digital Twin

In Fig. 2, the advantages of a digital twin are that it's an innovation catalyst, cost of prototyping is reduced and time and execution of System more smoother.

- Innovation Catalyst
  This property will reduce the risk of trying new implementations where the user has already tested the twin on a digital twin platform in the design stage in a way that the structure of matchinary or infrastructure is compatible for the real world or not.
- Cost reduction
  As everyone involved in the project will have keen sense on what they are making as they have seen on Digital twin platform which reduce the errors or mistakes which means this indirectly reduces cost of implementation
- Operational improvement
  One of the important features of digital twin is visualization through which when there is construction of complex engineering designs which involves hundreds of engineers they can just go to 3d visualization of digital twin which will give them their part in the bigger picture which significantly improves operational time by several times.

1.3 HOW TO MAKE A DIGITAL TWIN...?

For implementing Digital twin has followed 5 phases

I. **ANALYSIS PHASE:**
This is the theoretical phase because it will define all processes, data, digital twin and all the theoretical work needed to build a digital twin.We establish designs of the process that allow us to visually realize our twin making and things to be done.

II. **PROGRAMING PHASE:**
In the Digital twin platform we establish our virtual environment using programming languages.

III. **BUILD PHASE:**
In this phase we make a physical machine/Physical system we call it SCOUT.

IV. **CONNECTION PHASE:**
We established a connection between the virtual twin and physical system using Different data transfer methods in this phase.

V. **ACTIVATION PHASE:**
We made the system activate, this phase is similar to the go-live phase to test the system in different areas like speed, response time, cloud service etc.

1.4 OBJECTIVES

The objective of this project is to create a Physical system which is Wifi controlled and establish end to end data transmission from physical system to virtual system. We create a digital twin of the system which sends sensor data to the cloud(IoT Hub) where data is going to get streamed on the application interface. Then to create a web page which displays the system's 3D model which can be accessible to everyone.

1.5 CONTRIBUTION

We have given basic understanding on digital twin technology, the steps in creating a Digital Twin using Azure Digital twin platform and Hoops web platform. Using Azure Digital twin platform we have developed an application interface where data gets streamed on to and in Hoops web platform we have deployed a physical system in 3D virtual view which is accessible to everyone. We made Physical system wifi controlled with its sensor network with Temperature, humidity and Gyro Sensors which sends its data on to the IOT hub and gets displayed on to the application interface.

2. LITERATURE SURVEY

Review of literature

Digital twin technology is fast growing technology which significantly increases success rate and reduces effective construction time of complex engineering models which has been integrating for the industrial sector these days.This consists of three important aspects which are physical system, Virtual twin, Hub connecting physical and virtual model for data transmission.

2.1 Digital Twin: Enabling Technologies, Challenges and Open Research

Aidan Fuller, Zhong Fan, Charles Day, and Chris Barlow proposed that proper definition of Digital twin where there was no real definition for this concept of digital twin and introduction of digital twin in industrial sector clearing misconceptions on Digital twin and challenges faced during implementation of digital twin and tells how AI and IOT become components in Digital twin.Highlighting growing interest in Health and agriculture sector for digital twin.This paper basically paves a way for future implementation of digital twin.[1]

2.2 The Development of Digital Twin Technology Review

This paper proposed by Jiaju Wu, Xun Cheng, and Yonghui Yang which gives the theoretical perspective of Digital system construction, view on Product life cycle management(PLM), High data transfer techniques involved in Digital twin systems construction, Applications of Digital twin and role of digital twin in real world and about high data interaction between Physical and virtual twin. Where all these Fundamentally promote the efficient coordination of all stages of the product life cycle, and drive enterprise product innovation.[2]

2.3 Digital Twin in Industry: Technologies, Applications and challenges

Flavia Pires, Ana Cachada, Jose Barbosa proposed the Industry 4.0 program involving the adoption of the Digital twin in the industrial sector. This explains the evolution of digital twin from simple twin models to applying for real assets in industry which involve complex optimization techniques,real time data and machine learning. This involves origin of digital twin and actual usage in industrial sector which is like at what point this technology will become biggest assert to industrial sector and places where it need to improve its features to become a complementing tool for industrial sector.[3]

2.4 Digital Twin in Industry: State-of-the-Art

This paper was proposed by Fei Tao, He Zhang, Ang Liu, and A.Y.C. Nee. This outlines the key enabling features of Digital twin modeling, data fusion, interaction and collaboration, data, and service.review of current application of Digital twin in industries which concludes Digital twin is most popular topic concepts PHM, the core of DTs is modeling, and the most pressing issue is cyber-physical fusion.It outlines upcoming promising applications of digital twin which are dispatching optimization and operational control which are currently underexplored areas of digital twin.[4]

2.5 Digital Twin Technology

Zonyang Wang proposed that the digital twin technology is an ever growing technical area where this technology requires improved expertise of humans in visualization and simulation of virtual twin where this area is cited as an underdeveloped area currently which needs more improvement in applying real world conditions on virtual twin. This summarizes the definition of digital twin,connotation and implementation methods of digital twin technology.[5]

2.6 Potential of Digital Twin

This Paper was proposed by Abdulmotaleb El Saddik, Fedwa Laamarti, and Mohammad Alja'Afreh where it describes about how aggressively digital twin been adopting into industrial sector and how advantageous to real world in terms for cost of making, efficiency and five cutting edge technologies Big data, Multi modal interaction(MMI), IOT, AI technologies which are involved was discussed in this paper. Author stressed on how advantageous this technology is to the real world and the growth potential of Digital Twin in the future was highlighted.[6]

2.7 Digital twin technology : A Bird eye view

This Paper was proposed by Prarthana v, Lavanya ,Bhargav K where It explains how digital twin technology is used to build virtual prototype of real world product right from the designing of product, predicting possible effects on the product and study on how this technology will improve performance of the real world product. In this they used technologies like machine learning, AI , Virtual reality and cloud computing. This paper also addresses the comparison of IOT and IIOT in Digital Twin. This also provides or explains some real time applications on Digital twin like Automobiles . Gives an overview on how Digital Twin works and where this technology has been used to solve problems in the real world.[7]

2.8 Digital twin from smart manufacturing to smart cities:A survey

This Paper was proposed Georgios Mylonas,Athanasios Kalogeras, Georgios Kalogeras,Christos Anagnostopoulos where it explains about how Digital Twin technology is advancing these days with advancement in technologies like IOT,Big Data, Cloud computing and how it is adjusting with the growth in AI and ML and explains how can we use this technology within Smart city and clearly explain the complexity and hurdle we face in integrating it into smart cities.[8]

3. PROBLEM FORMULATION AND IMPLEMENTATION

3.1 OBJECTIVE

Our objective is to analyze Digital twin concept and understand what is its role in current stage of technological development, develop a 3D model which will exactly reflect the physical system where this can be done using CAD application where we implement our design for the physical system and by hoops platform we import that model on to the AZURE Web platform where using a web link anyone could access the digital twin and visually see the exact system on the webpage . To build the physical system similar to CAD model design. Then establish connection between physical system and Azure twin which is constructed virtually and establish a channel to transmit data from physical system on to the digital twin and test the digital twin whether it is giving accurate data of physical system through Azure Digital twin platform.

3.2 3D VIRTUAL REPRESENTATION OF PHYSICAL SYSTEM

- For this we have created our 'SCOUT' Digitally through CAD Platform.
- CAD is Computer Aided Design which is a way of representing systems or things through 2D drawing on the CAD application which gives a base plan for how the real product should be made.

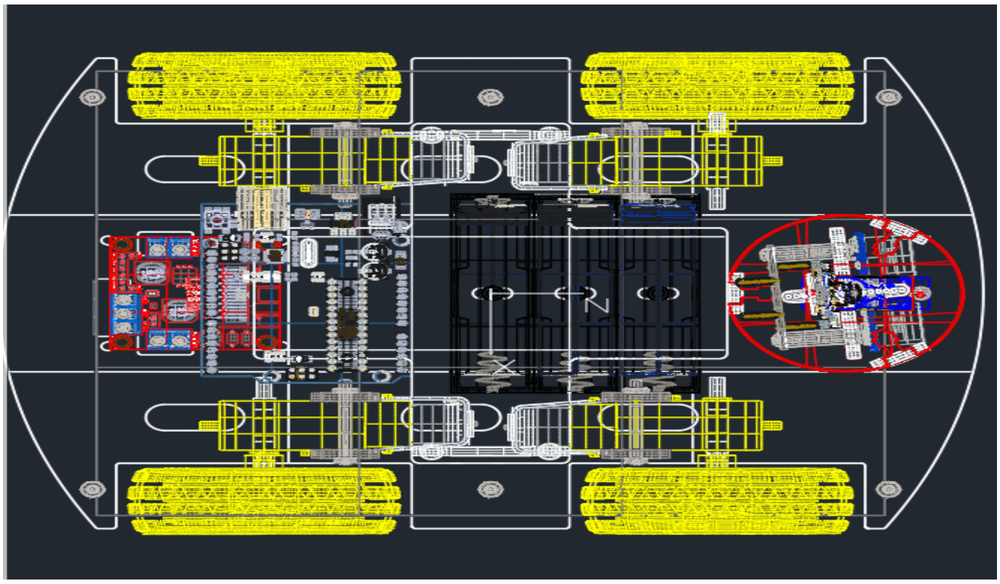


Fig. 3 CAD model of our device SCOUT.

In Fig.3, we can see the 3D representation of SCOUT in CAD application, where the wheels and motors are in yellow, frame in gray, Power supply unit in black, Sensor socket in right side(front end) o, NodeMCU in center,Motor driver in the back of controller.

- 3D representation is something which is really helpful and handy to understand how the product looks after finished.
- A software kit 'HOOPS web platform' perfectly complements ADT by importing CAD files which were created and displaying it in the web page.
- Using HOOPS and ADT we can create any system in the world with an attractive and user friendly interface.

- When ADT nodes are associated with 3D data such as Buildings,Bridges,Machinery we can view in HWV so that the owner could get a keen sense of that structure.
- HWV(Hoops web viewer) is the browsing page we get into when we import our CAD file and convert and run it on the server.
- While creating our 3D model of the physical system, we converted CAD file to stream cache standalone(SCS) file by using Hoops converter.
- **Stream Cache Standalone (SCS)** - These singular files can be easily loaded with REST API functions, just as you would some other file. The entire file must be downloaded before visualizing.[9][10]
- After that we deployed 'SCOUT' into AZURE APP service.

Virtual 3D-Model link:

https://finalproject2022.z30.web.core.windows.net/hoops_web_viewer_sample.html?scs=scs_models/scout.scs

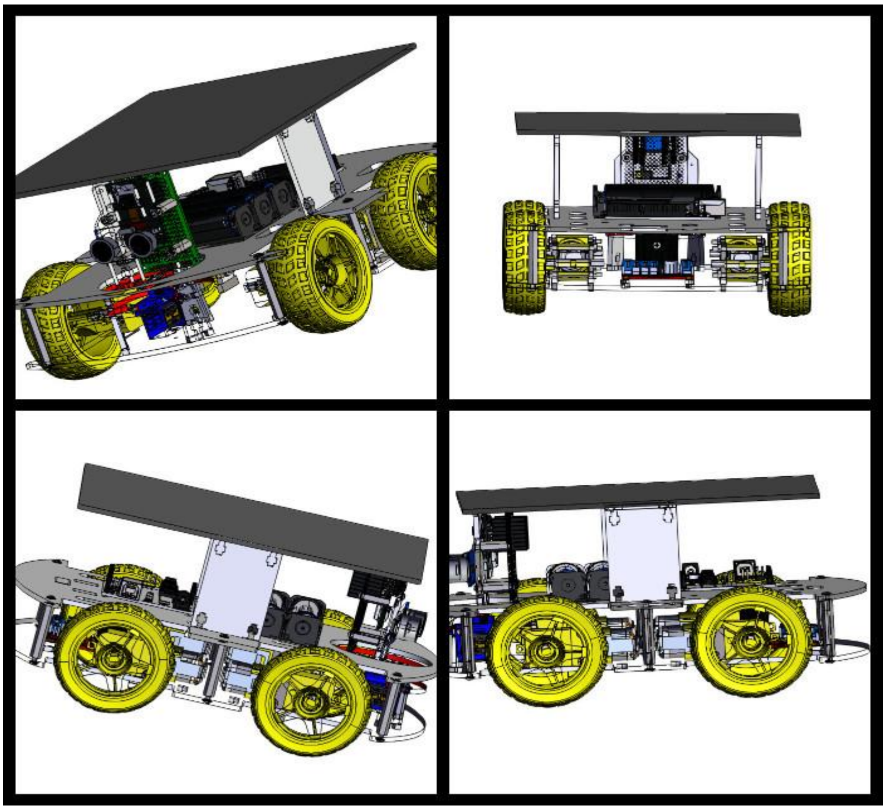


Fig. 4 Snapshots of Virtual twin on Webpage(HOOPS)

3.3 DIGITAL TWIN IMPLEMENTATION

Some of the softwares for implementing Digital twins are - Azure Digital Twin, Mosquitto Ditto and AWS. The software which we used for implementation is Azure Digital Twin.

Procedure:

- Create Azure Digital Twin Instance.
- Create an Azure IOT hub.
- Create Azure Function App to manage data flow between client and the digital twin instance.

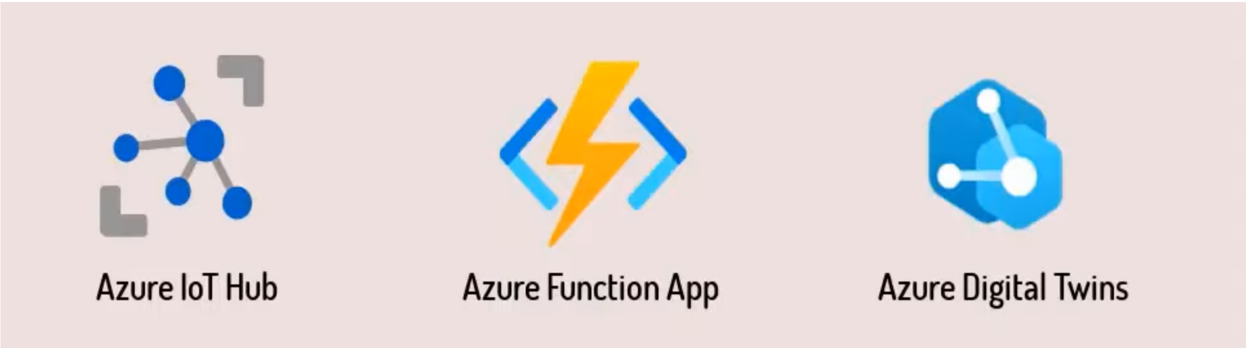


Fig 5. Required Azure Resources for Digital Twin implementation

3.3.1 Creating Azure Digital Twin Instance:

Before creating the Digital Twin Instance we should create a Digital Twin Model using DTDL language.

3.3.1.1 DTDL

The Digital Twins Definition Language (DTDL) is a language for describing models and interfaces for IoT digital twins. It is based on the open source language JSON-DL.

A general Digital twin model consists of-

- **Properties-** The variables with this type require a storage system to retain permanently and they can be updated when necessary. It can be used for name, serial no, color etc.
- **Telemetry-** It is mostly used for sensor data. It is not stored in the digital twin, instead it is used to do operations like machine learning the twin.
- **Components-** It is used to define separate models which are part of a model.

- **Relationships-** It is used to tell how one digital twin is related to another.

The written code is saved as a .json folder.

```
[
    {
      "@id": "dtmi:com:microsoft:azure:mproject_8;2",
      "@type": "Interface",
      "displayName": "Major Project DT",
      "@context": "dtmi:dtdl:context;2",
      "contents": [
        {
          "@type": "Property",
          "name": "Temperature",
          "schema": "double"
        },
        {
          "@type": "Property",
          "name": "Humidity",
          "schema": "double"
        },
        {
            "@type": "Property",
            "name": "X",
            "schema": "double"
        },
        {
            "@type": "Property",
            "name": "Y",
            "schema": "double"
        },
        {
            "@type": "Property",
            "name": "Z",
            "schema": "double"
        }
      ]
    }
]
```

Fig. 6 Digital Twin model Code

In Fig. 6, the name of the model is given as “Major Project DT” and is of the Interface type. Temperature, Humidity and tilt angles X, Y, Z are of the type property and their schema is double as all of them are decimal values and all of them need to be stored.

3.3.1.2 Digital Twin Instance

To create a digital twin instance in Azure we go to create a resource, then search Azure Digital Twin and click on create. Then we select our subscription, location, create a new resource group, and write the resource name and then review + create.

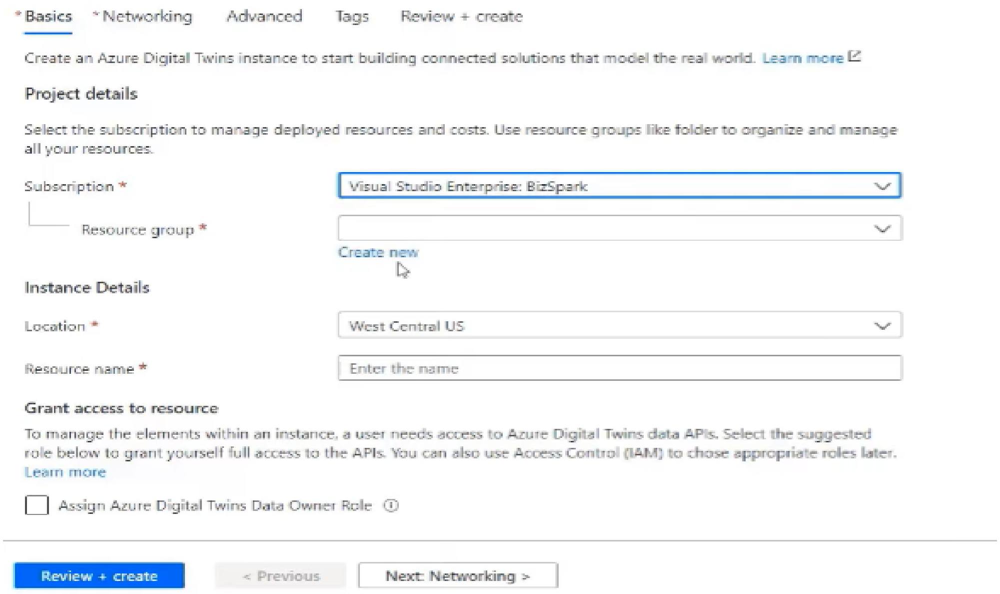


Fig. 7 User Interface for creating a Digital Twin Instance.

The host name, resource group and instance name should be noted. As the host name should be used to access the Digital Twin Through Digital Twin Explorer.

After deployment we assign the “Azure Digital Twin Data Owner” role to the Microsoft / Azure account which we use to login into our IDE. We do this by adding role assignments in Access Control tab.

To access the data from the Assure digital twin instance we can either use Azure Digital Twin Explorer or any client app. Using Azure Digital Twin Explorer we can upload models, retrieve data, change data and a lot more.

To work on Azure Digital Twin Explorer first we need to download Node.js(10+ version) and the zip file from the github link https://github.com/Azure-Samples/digital-twins-explorer[11]. Then go to the src file of the digital twins explorer, open the command prompt, and run the commands “npm install” and then “npm run start”. With this the Azure Digital Twin Explorer will be installed.

Then we need to login to the account which has the ownership role through our PC. When we open the Azure Digital Twin Explorer it asks to give the URL of the digital twin which is the “https://host name of the Instance”. The next step is to upload the digital twin model .json file into the Azure Digital Twin Explorer and then create a new digital twin with a unique name.[12] [13]

When we do this a new node is created in the graph view. Clicking on this node shows the properties of the digital twin. Also if the model has any components or relations they can also be visible by an arrow pointing to the next node.

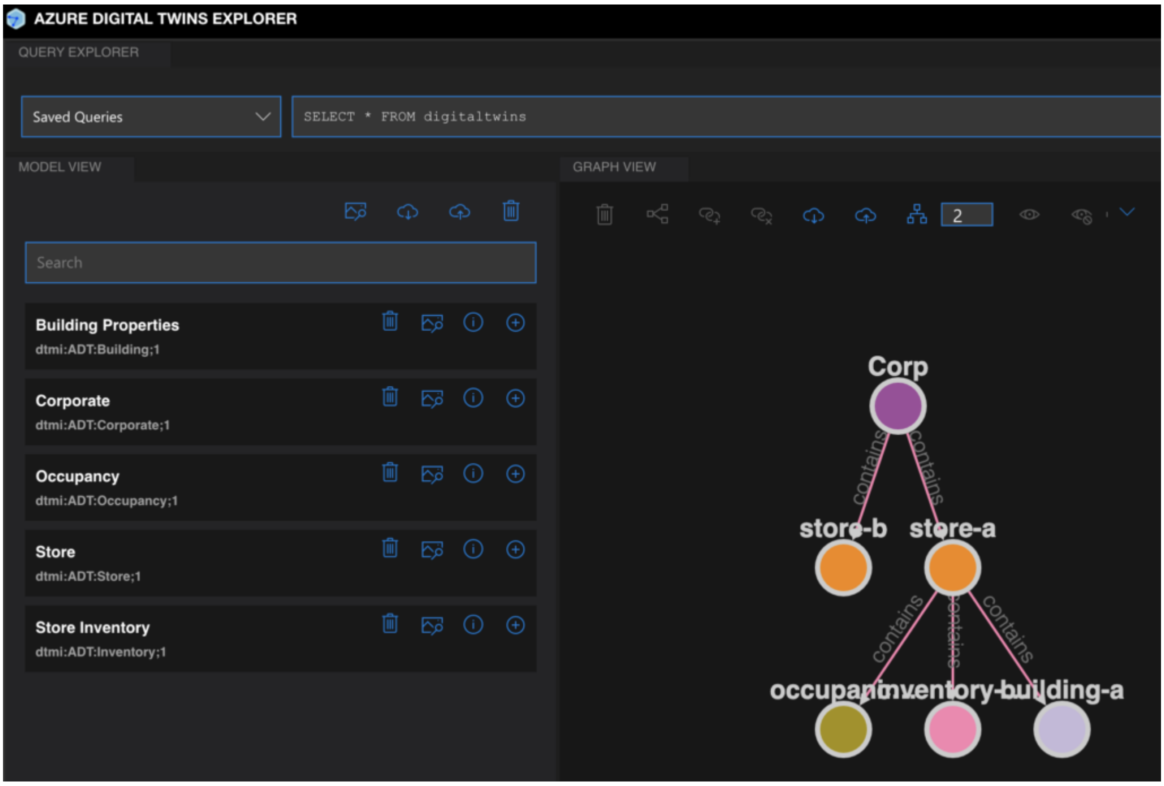


Fig. 8 Digital Twin Graph Example

In Fig. 8 store-a and store-b are components of Corp. occupancy-a, inventory-a, and building-a are components of store-a. Each cell is a digital twin which is created by clicking on the + symbol. Digital twins are connected by arrows which are either ‘contains’ or ‘relates’ telling whether the twins are components or related respectively.

3.3.2 Creating an Azure IoT Hub

Same like Azure Digital Twin instance we search for Azure IoT Hub and create a new IoT Hub. The resource group must be the same resource group as the Azure Digital Twin instance.

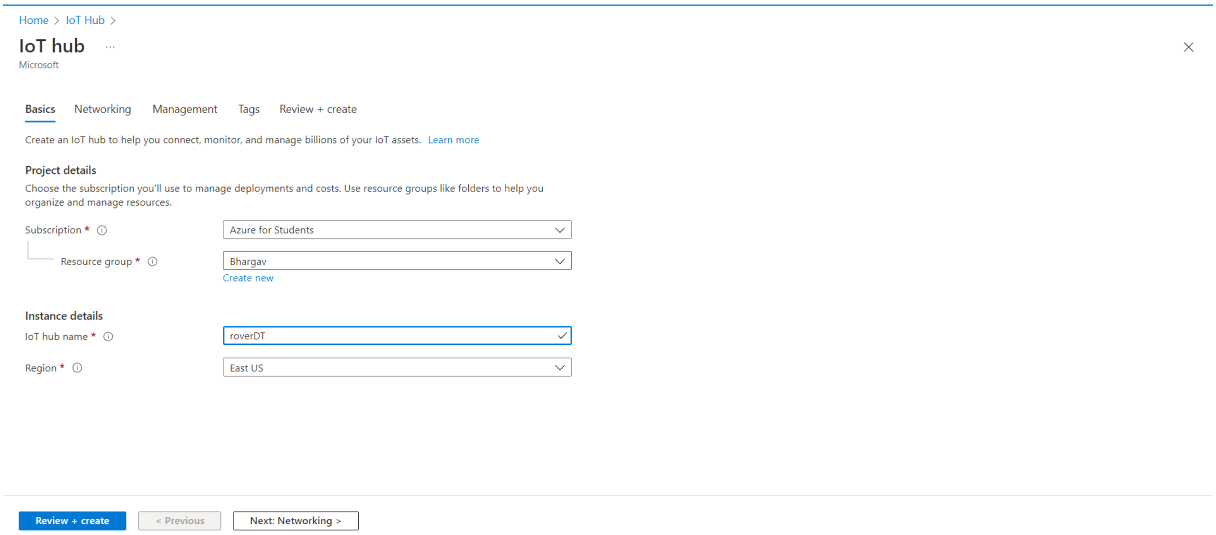


Fig. 9 User interface for creating Azure IoT Hub

After Deployment of the IOT Hub a new IoT device is created with the same name as Digital Twin in the Explorer. Now that the IoT Device is created the next step is to create a new Function App.

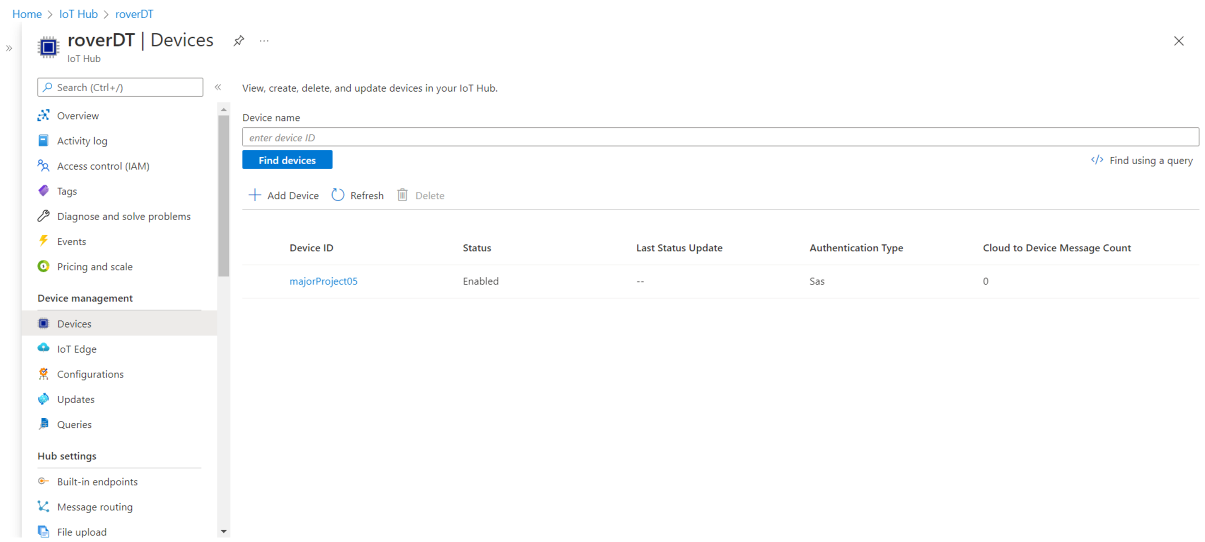


Fig. 10 IoT Hub with new DeviceID majorProject05

In Fig. 10, the new device with the deviceID majorProject05 is created by clicking on add device, type the deviceID name and then save.

3.3.3 Creating an Azure Function App

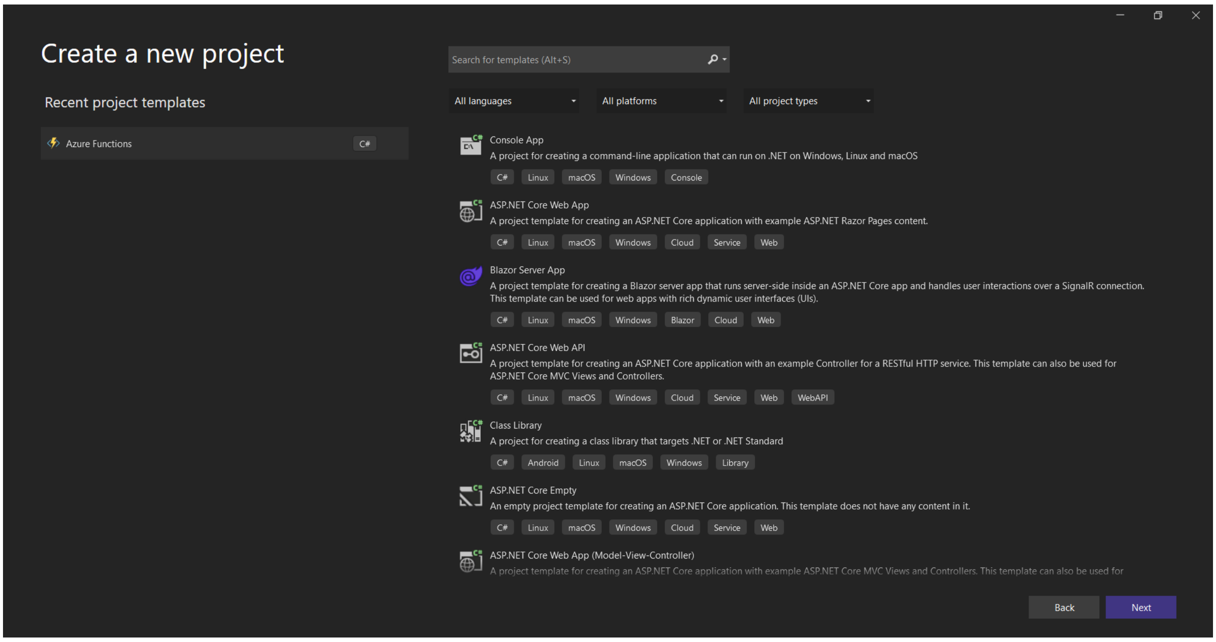


Fig. 11 Visual Studio user interface.

A new Azure Function App is created using Visual Studio(Fig. 11) and event grid trigger is selected as a template. The necessary extensions are installed using NuGet packages(Fig. 12). The extensions that are required to be installed and included in the code are-

```
using System;
using Azure;
using System.Net.Http;
using Azure.Core.Pipeline;
using Azure.DigitalTwins.Core;
using Azure.Identity;
using Azure.Messaging.EventGrid;
using Microsoft.Azure.WebJobs;
using Microsoft.Azure.WebJobs.Extensions.EventGrid;
using Microsoft.Extensions.Logging;
using Newtonsoft.Json;
using Newtonsoft.Json.Linq;
```

Fig. 12 NuGet packages required to be installed and included in code.

The function app that is should send and receive messages by deserializing the messages from event grid, separate the required properties and then serialize the message in .json format which is sent to the Digital Twin Instance.The App is then published with a unique name. Then go to the Function App in the Azure portal and add a new application setting named ADT_SERVICE_URL with the Value: “https:\\Host name” in Configurations.

Then permissions and role assignments are given to the function app from the IoT Hub and Digital Twin Instance by informing that they all belong to the same resource group. An event subscription is done for IoT

Hub to Event Grid with the Function App as an end point.

The Telemetry messages are sent to the Function App through the event grid. Then the Function App updates the Azure Digital Twin Explorer. This is how the IoT Hub and Digital twin Instance communicate with each other.

This is the basic setup of the Digital Twin in a virtual platform. The next step is building a physical system.

Function App Code:

```
namespace AA_DTHubtoTwins
{
    0 references
    public class AA_HubtoTwins
    {
        private static readonly string adtInstanceUrl = Environment.GetEnvironmentVariable("ADT_SERVICE_URL");
        private static readonly HttpClient shttpClient = new HttpClient();

        [FunctionName("IoTHubtoTwins")]
        0 references
        public async static void Run([EventGridTrigger] EventGridEvent eventGridEvent, ILogger log)
        {
            if (adtInstanceUrl == null) log.LogError("Application setting \"ADT_SERVICE_URL\" not set");

            try
            {
                // Authenticate with Digital Twins
                var cred = new DefaultAzureCredential();
                var client = new DigitalTwinsClient(
                new Uri(adtInstanceUrl),
                cred,
                new DigitalTwinsClientOptions
                {
                    Transport = new HttpClientTransport(shttpClient)
                });
                log.LogInformation($"ADT service client connection created.");
                if (eventGridEvent != null && eventGridEvent.Data != null)
                {
                    log.LogInformation(eventGridEvent.Data.ToString());

                    JObject deviceMessage = (JObject)JsonConvert.DeserializeObject(eventGridEvent.Data.ToString());
                    string deviceId = (string)deviceMessage["systemProperties"]["connectionDeviceId"];
                    var temperature = deviceMessage["body"]["Temperature"];
                    var humidity = deviceMessage["body"]["Humidity"];
                    var X = deviceMessage["body"]["X"];
                    var Y = deviceMessage["body"]["Y"];
                    var Z = deviceMessage["body"]["Z"];

                    log.LogInformation($"Device:{deviceId} Temperature is:{temperature} Humidity is:{humidity} X is:{X} Y is:{Y} Z is:{Z}" );

                    var updateTwinData = new JsonPatchDocument();
                    updateTwinData.AppendReplace("/Temperature", temperature.Value<double>());
                    updateTwinData.AppendReplace("/Humidity", humidity.Value<double>());
                    updateTwinData.AppendReplace("/X", X.Value<double>());
                    updateTwinData.AppendReplace("/Y", Y.Value<double>());
                    updateTwinData.AppendReplace("/Z", Z.Value<double>());
                    await client.UpdateDigitalTwinAsync(deviceId, updateTwinData);
                }
            }
            catch (Exception ex)
            {
                log.LogError($"Error in ingest function: {ex.Message}");
            }
        }
    }
}
```

Fig. 13 Function App Code.

In Fig. 13 The code deserializes the messages in EventGrid which are in .json format to extract the required information and then it serializes the required message(Temperature, Humidity, X, Y, Z) in .json format and updates the digital twin explorer.

3.4 OUR PHYSICAL ENVIRONMENT

- Our Physical system is a robot controlled Car we call **“SCOUT”.**
- As the name says it monitors and finds things.
- It’s Functionalities are
  - Sensing the environment
  - Wifi controlled
  - Cloud connected
- It’s just a simple system that we are building, our main goal is to realize this physical system through digital twins.
- It's true that the digital twin was made for large systems like aircrafts,bridges, factories but this project will be a first step in learning and building the large projects.

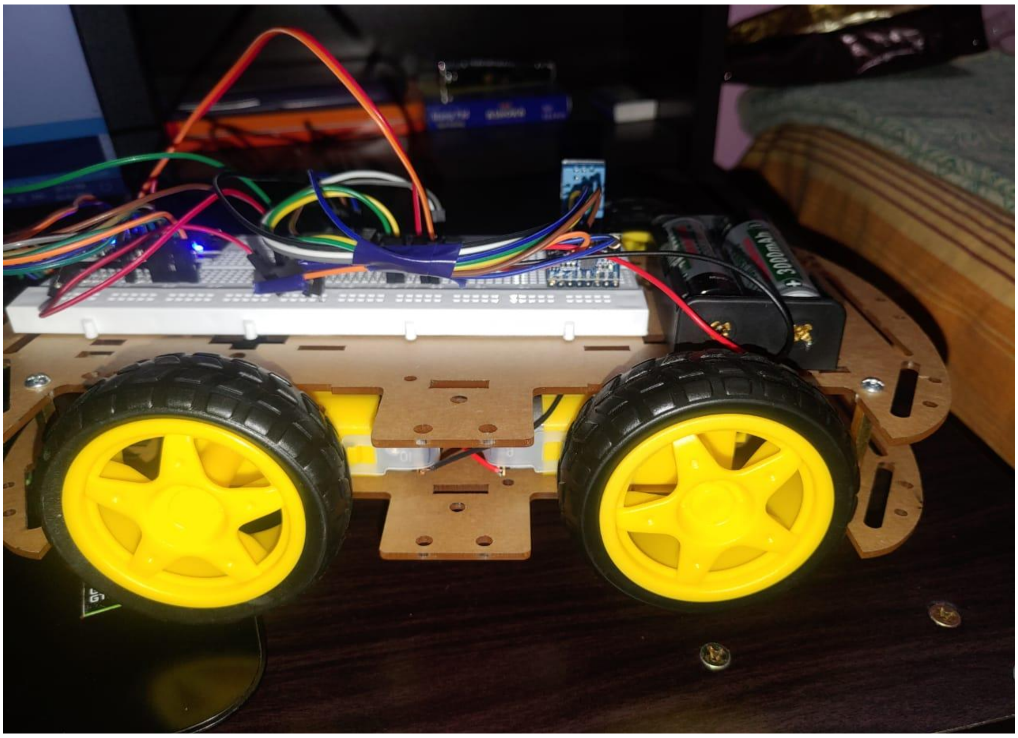


Fig. 14 Image of SCOUT (side view).

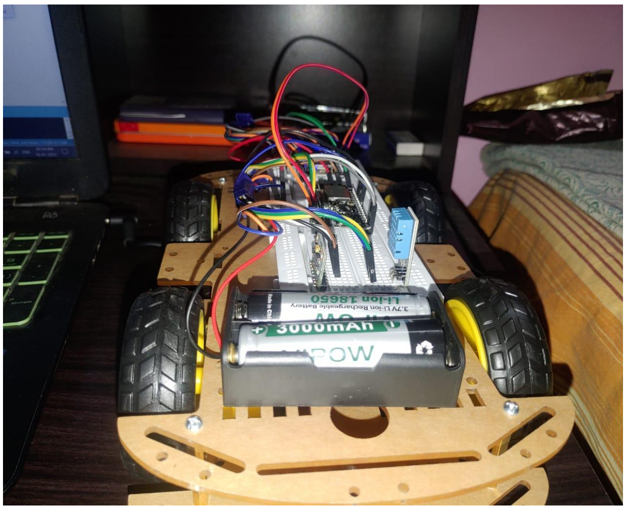


Fig. 15 Image of SCOUT (front view)

3.4.1 Hardware

1. **2 NOdeMCU ESP8266 :**

   NodeMCU is a microcontroller, it is specifically built forIoT based Applications. It is infused with the firmware that works on the ESP8266 Wi-Fi system on chip technology from Espress-if-Systems with hardware based on the ESP-12 module(internal Chip that comes on the NOdeMCU ESP8266).

   SPEC:

   32-bit RISC Xtensa CPU, Operating Voltage(3.3V), Input Voltage(7-12V), 16 Digital I/O Pins (DIO), 1 Analog Input Pins (ADC), UART, Flash Memory (4 MB), SRAM(64 KB), Clock Speed(80-160 MHz), USB-TTL, PCB Antenna

2. **DHT11 Humidity and Temperature Sensor:**

   DHT11 is a low cost highly reliable and stable sensor for measuring humidity and temperature.It consists of capacitive humidity sensor and a thermistor which measures surrounding atmosphere(air) gives result as output in digital signal on the data pin.It works on the principle of digital signal acquisition technique and temperature&humidity sensing tech.

   SPEC:

   voltage(3.3v~5.5v DC)
   Measurement range of humidity from 20-90%RH
   Temperature from 0-50* celsius ,
   The accuracy of humidity is 1%RH and temperature is 1* celsius.

3. **L298N Motor Driver:**
   The L298N IC is used to control the speed of a DC motor using PWM technique and spin direction of the motor is controlled by changing the polarity of the input voltage using H-bridge.

4. **Gyro Sensor:**
   MPU6050 sensor is used to measure the tilt angle of the rover scout using a 3-axis gyroscope with MEMS(Micro Electro Mechanical System) technology. It also contains an accelerometer module which can measure acceleration up to 16g.

5. **Base for Scout:**
   Cardboard pieces are assembled to build the body and support the modules and wheels.

6. **4 Motors:**
   A gear motor is used which is an all-in-one combination of a motor and a gearbox. The addition of a gear head to a motor reduces the speed while increasing the torque output to move the rover even with heavy load.
7. **4 Tires**

3.4.2 Building Scout

Scout is a rover which can be controlled using the wifi through blynk app. By pressing the UI command in the blynk app(created manually) sends the data to the nodemcu board by blynk cloud.
Steps to create blynk UI:

Step1 :Create a blank template and add four switch buttons and one slide button
Step2: Create five virtual pins(four switches and 1 variable counter) to connect the UI buttons and slider.
Step3: Arrange the user interface buttons as a controller.
Step4: Obtain auth key of the device to interconnect nodemcu and blynk app
Step5: Place the unique auth key in the code and upload it to the nodemcu

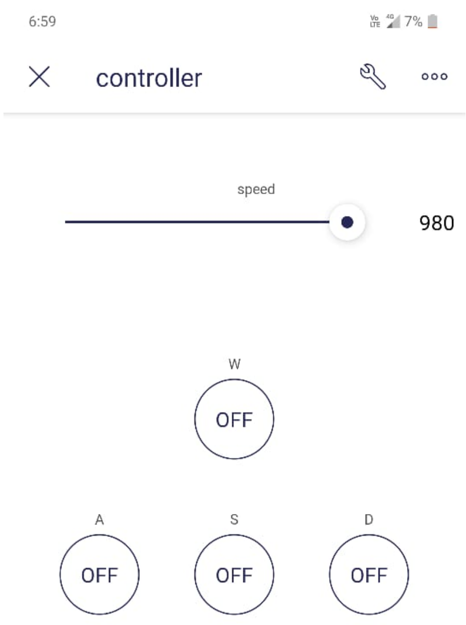


Fig. 16 User Interface of SCOUT in Blynk App

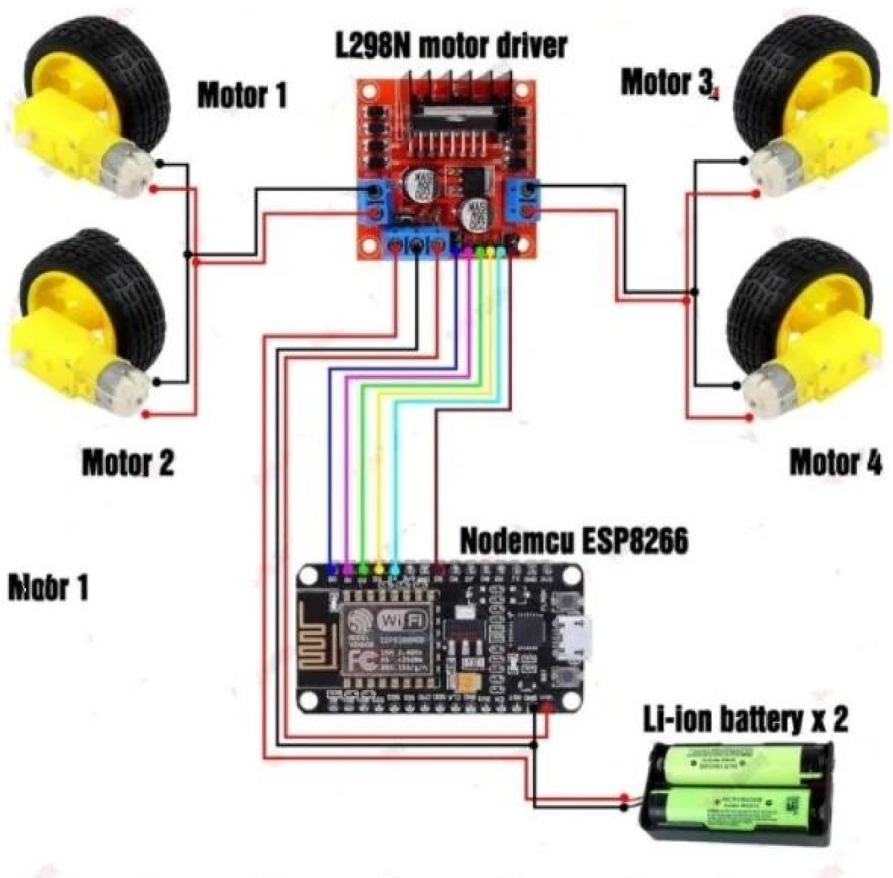


Fig. 17 Circuit diagram of SCOUT

In Fig.17, Node MCU is the controller used for controlling operation SCOUT where it is connected to the power unit and to the L298N motor driver which controls motors connected to 4 wheels of scout.

3.5 CONNECTING ROVER SCOUT TO IOT HUB

Sensor telemetry is sent to the iot hub using NodeMCU (ESP8266 wifi module) by HTTP Post protocol.These device telemetry is sent to the eventgrid and we can further save the telemetry into the blob storage by creating a container end point. The sensor telemetry is sent in the JSON object from the NodeMCU and the iot hub receives and stores the telemetry in the JSON file.

Step1: Create an instance in the iot hub named roverDT.

Step2: Create a device in the iot hub with the device-id(majorProject05).

Step3: Acquire the sas token of the iot hub roverDT using following command in azure shell:-az iot hub generate-sas-token –device-id {YourDeviceID} –hub-name {YourHubName}
example:az iot hub generate-sas-token --device-id majorProject05 --hub-name roverDT
sas: "SharedAccessSignaturesr=roverDT.azure-devices.net%2Fdevices%2FmajorProject05&sig=PULDKG3Ow45f9%2B9kP05GYZKFhHnUIZRTjsQnV%2FRsZ4w%3D&se=1652731092"

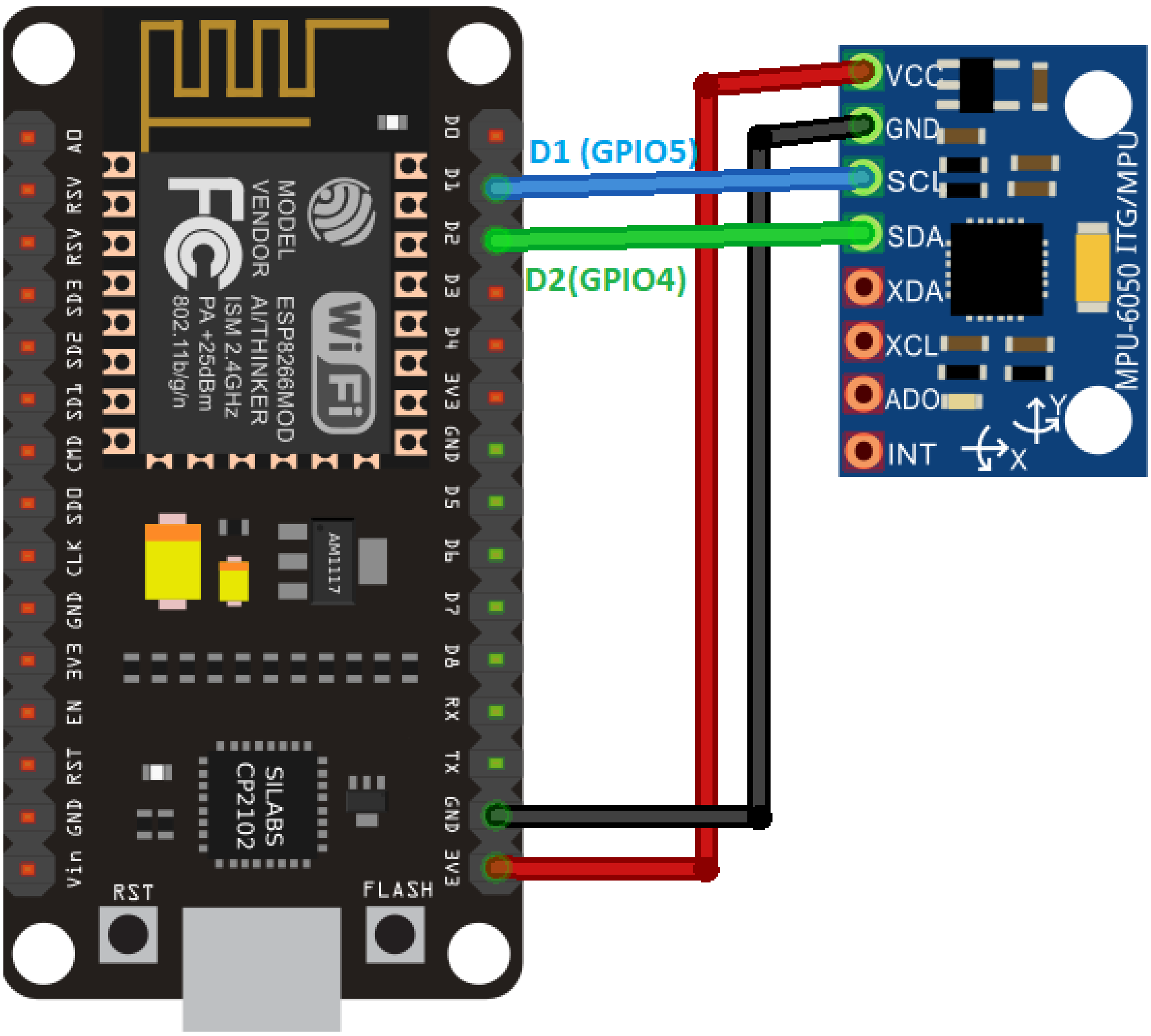
Step4: Generate fingerprint for the Iot hub using Open SSL command prompt using the following command:
openssl s_client -servername {IoT Hub Host Name} -connect { IoT Hub Host Name}:443 | openssl x509 -fingerprint -noout

Example: openssl s_client -servername roverDT.azure-devices.net -connect roverDT.azure-devices.net:443 | openssl x509 -fingerprint -noout
Fingerprint=89:EC:30:21:CC:F6:E6:70:39:33:AC:DB:C6:84:3E:F9:0B:EF:28:A5

Step5: Obtain URI of Iot hub to send device event HTTP Post API.generally URI:https://{iot_hub_name}-iothubname.azure-devices.net/devices/{device-id}/messages/events?api-version=2020-03-13
myURI:https://roverDT.azure-devices.net/devices/majorProject05/messages/events?api-version=2020-03-13

Step6: Add all the above data into the respective variables in the code and then upload into the node mcu.

Step7: Read the serial output in COM in the arduino IDE and check for the return code of the HTTP. 401 indicates failure in posting the data to the bub,code 204 indicates successful post and code -1 indicates internet is not active.



D4 pin -> data pin of DHT11

Fig. 18 Circuit diagram for connection between NODEMCU ESP8266 and MPU-6050

In Fig. 18, SCL and SDA of MPU-6050(Gyroscope) should be connected to D1 and D2 respectively as they are the ports for SCL and SDA to communicate in NODEMCU ESP8266.

4. RESULT

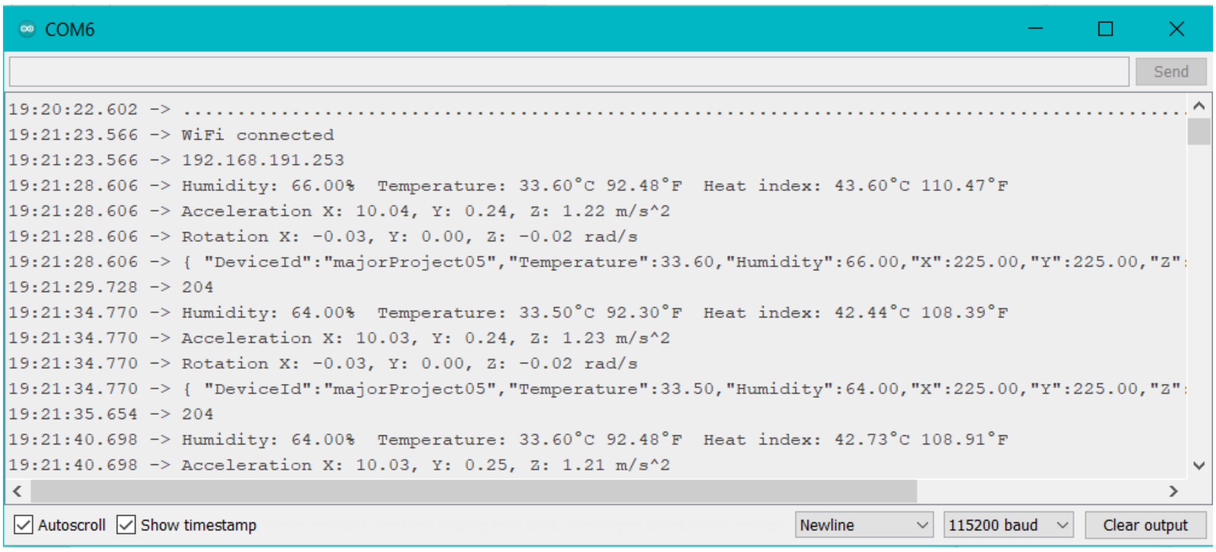


Fig. 19 Output from the serial monitor of SCOUT in Arduino IDE.

In Fig. 19, the messages that are seen in serial monitor are being sent by POST http and the message consists of the deviceID, Temperature, Humidity and the Tilt angles of SCOUT X,Y, Z.

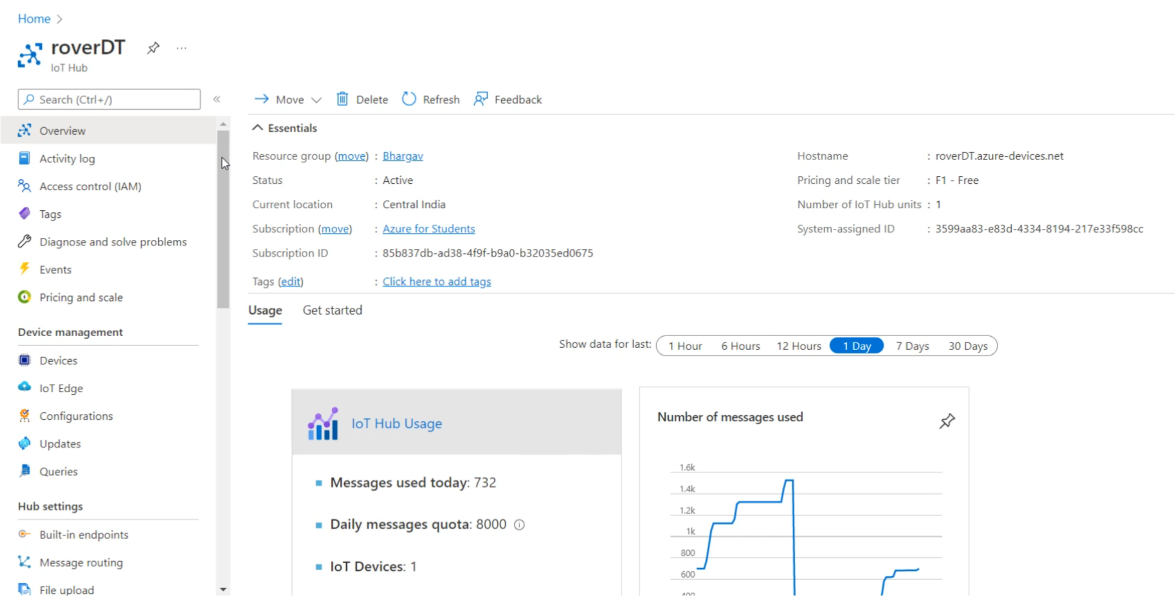


Fig. 20 Current number of messages received on IoT Hub from SCOUT: 732

In Fig. 20, The messages that are being sent from the device are received at the IoT Hub with deviceID majorProject05 and here the current messages are 732. When we refresh the value should increase.

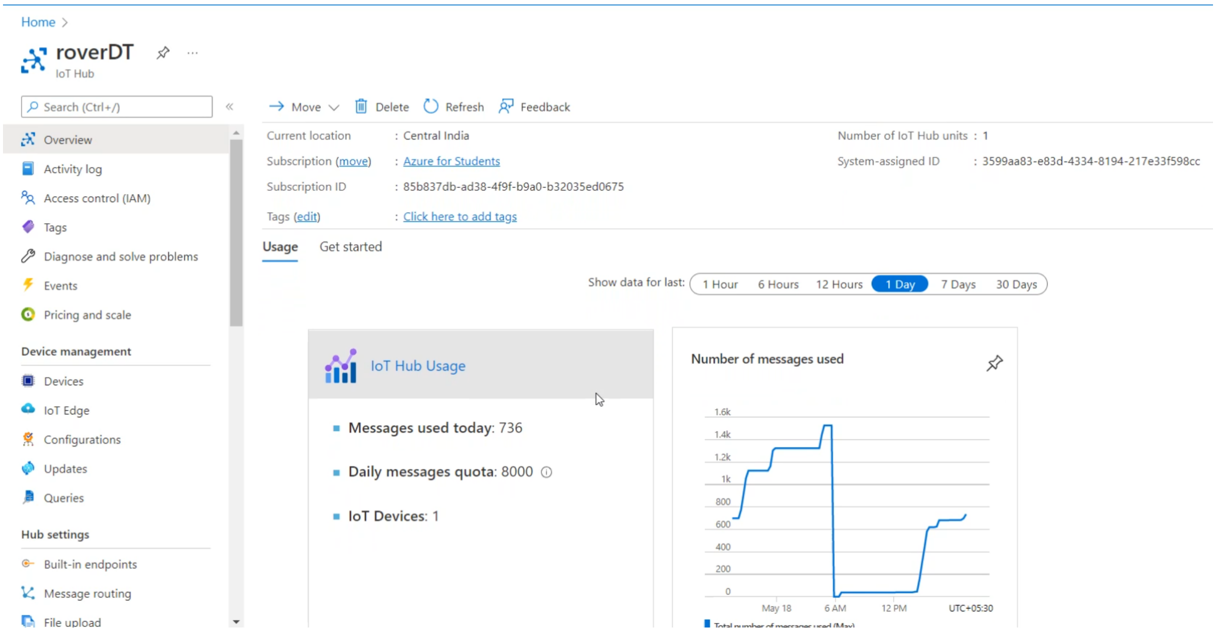


Fig. 21 Number of messages received on IoT Hub after refreshing: 736

In Fig. 21, we can see that after refreshing the messages have increased from 732 to 736. This indicates that messages are being received by the IOT hub.

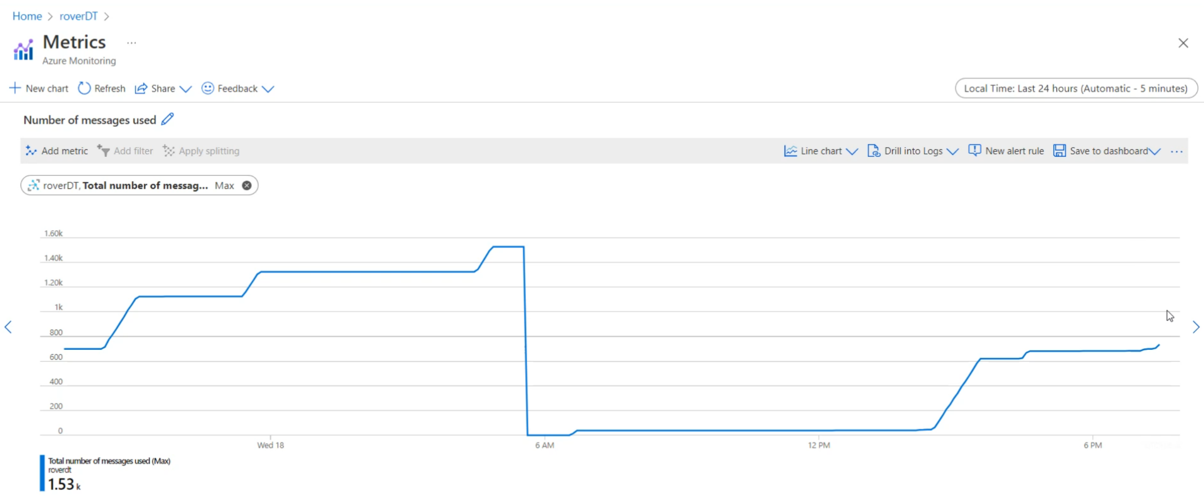


Fig. 22 Graph of total number of messages used.

In Fig. 22 the above graph shows the number of messages used every day. We can see at the end of the graph there is a slight increase in messages this shows messages are being sent at that instant.

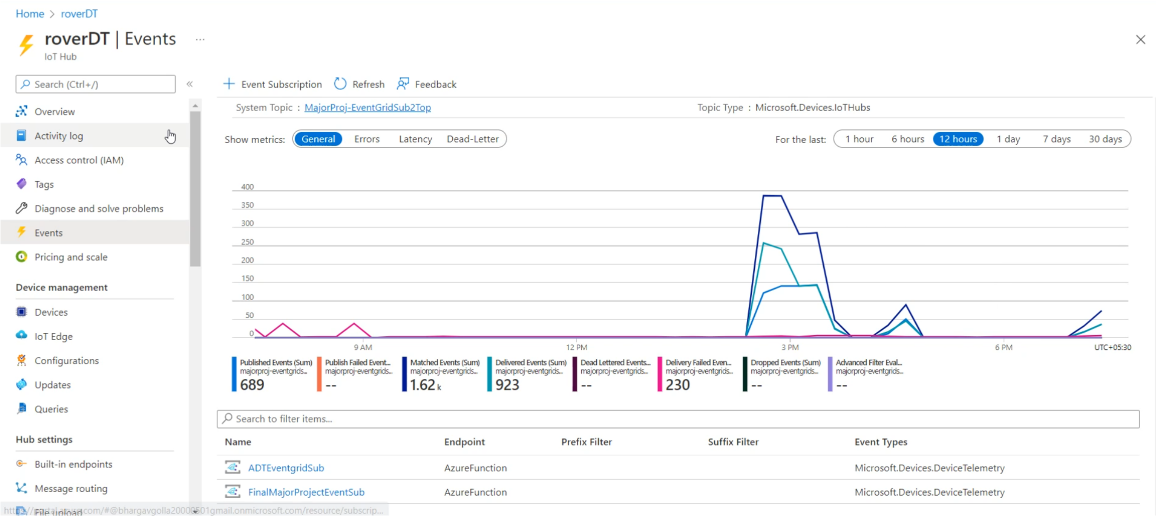


Fig. 23 Graph showing messages received on Event Grid.

This graph shows that messages from IoT Hub are being received by the Event Grid. At the end of the graph we can see messages being received as there is a slight increase.

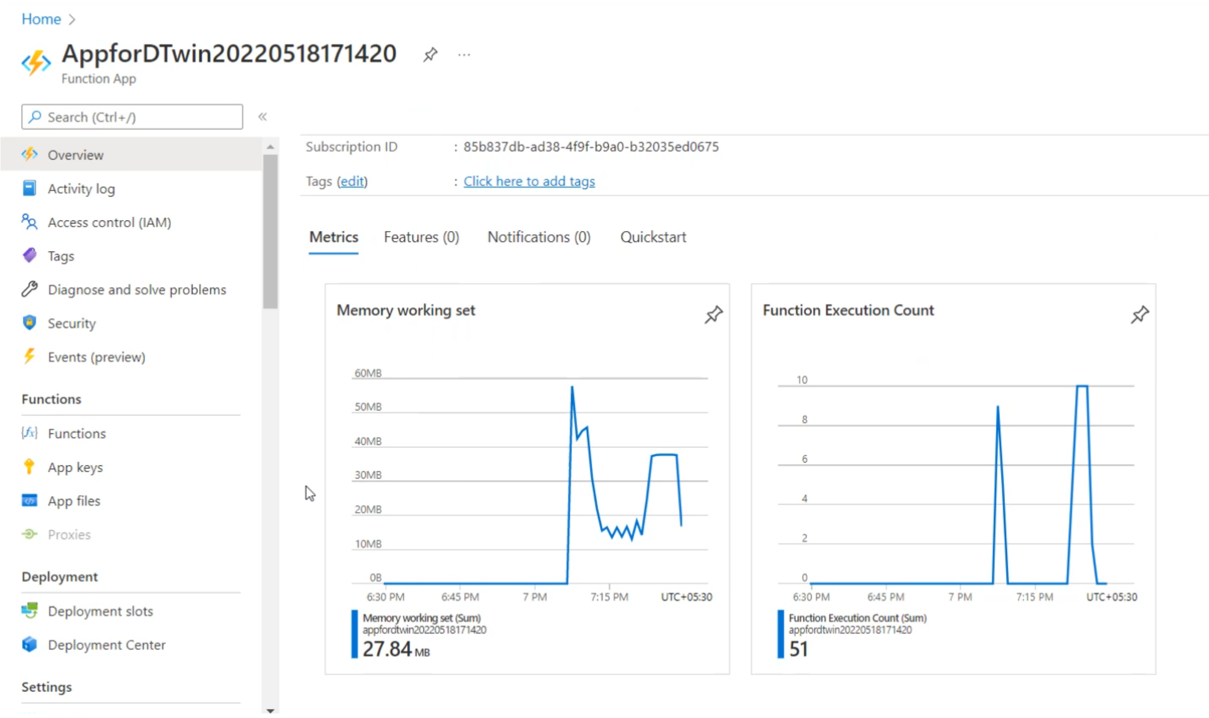


Fig. 24 Azure Function App

In Fig. 24, we can see in the Function Execution Count Graph at the end there is change in it, meaning the Function app is triggered by Event Grid. So if we run Query in Azure Digital Twin Explorer the properties should change.

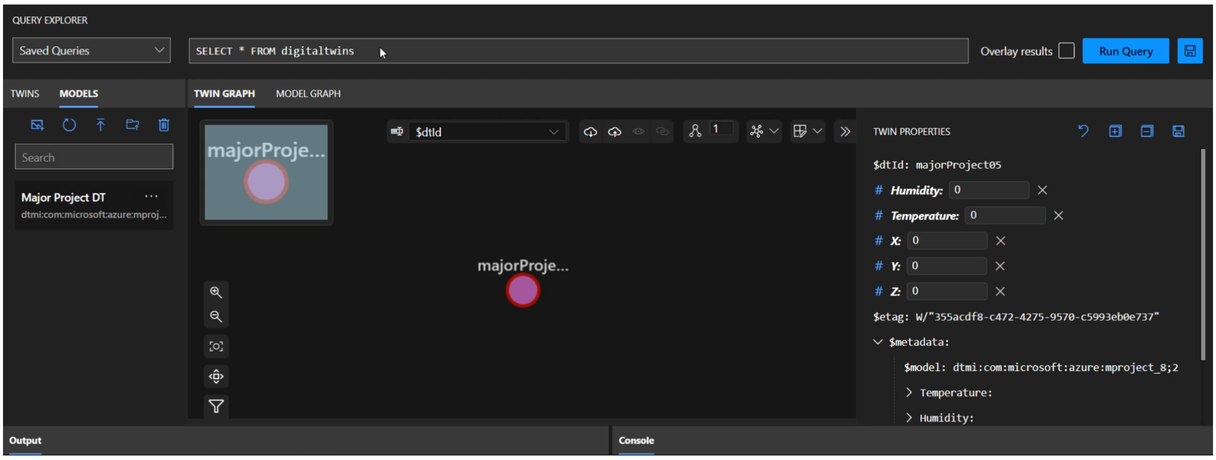


Fig. 25 Azure Digital Twin Explorer before updating

Fig. 25 is the image of our Azure Digital Twin Explorer. On the right side we can see our Digital Twin ID($dtId) which is majorProject05. Here all the properties Humidity, Temperature, tilt axises X, Y and Z are all initiated to 0. To the left under Major Project DT we can see our model ID which we have given to the digital twin model when we made it using DTDL.

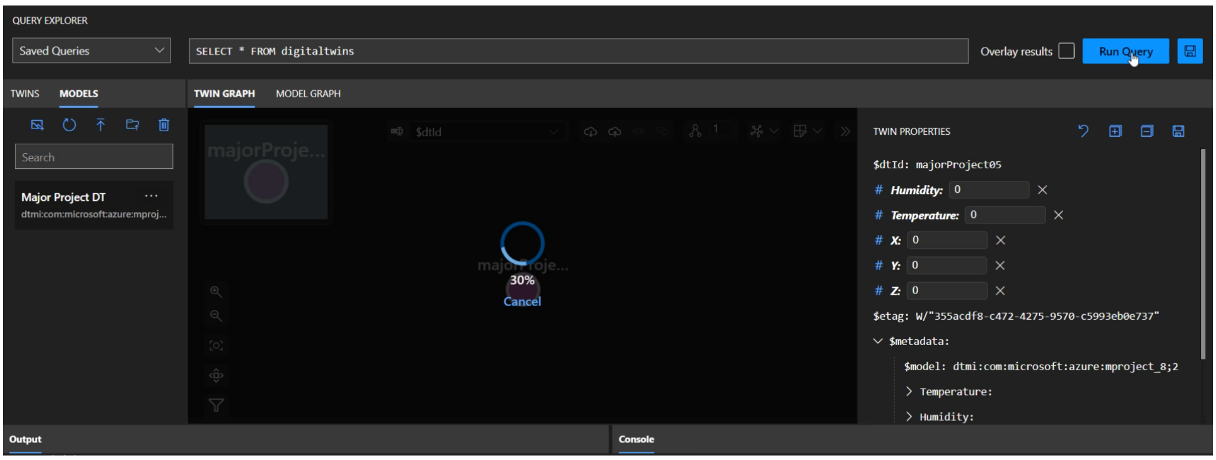


Fig. 26 Azure Digital Twin Explorer during query execution.

In Fig. 26, we run the query "SELECT * FROM digitaltwins" in our Azure Digital Twin Explorer to update the Digital Twin Explorer. Here "*" indicates to select all the digital twins in “digitaltwins”. After running the query the screen refreshes, the above image shows the Digital Twin Explorer refreshing. The query language is similar to SQL. We can even select the digital twins which only have humidity>60 by running the appropriate query.

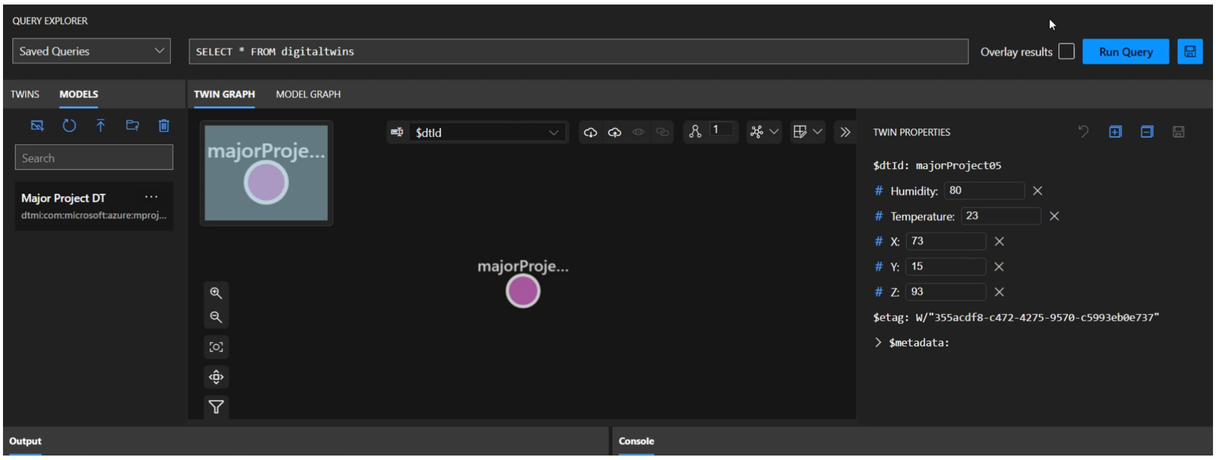


Fig. 27 Azure Digital Twin Explorer after Updation.

Fig. 27 is the Azure Digital Twin Explorer after running the query “SELECT * FROM digitaltwins”. We can see that the properties of the digital twin have been updated.

By Observing the above results we can see the Device(SCOUT) sending messages to IoT Hub which is sending the same messages to the Event Grid which then triggers the function app to update the Digital Twin. At the end of this project we can see a 3D model of the physical system on webpage and through Azure Digital Twin(ADT) we could sense the condition of the environment around the physical system and state of the physical system which is temperature,Humidity and position of system in the empty space with respect to X,Y,Z axis(basically it say how tilted was the system from original position) .

5. CONCLUSION

This project briefly gives you an understanding of how to construct a digital twin for your model which significantly becomes a strong foundation in making complex digital twin models. Digital twin is Work in progress technique which significantly increases the success rate of large and costly complex engineering models.In upcoming decades Digital twin will become most dependent and most used technique for the Human race starting from complex aircrafts or in construction of sky towers as this application will significantly increase the success rate of any engineering model.Specifically through this project we achieved end to end sensor data transmission from physical system to AZURE platform and 3D visualization of physical system in web page.It is fast growing technology in future digital twin platform providers are trying to make twin more interactive and real time data display on the webpage. Keep up with creators and let's evolve together.

5.1 FUTURE SCOPE

Hoops developers are taking more initiative to make 3D models more interactive and developer friendly which needs integration between HOOPS and ADT .In future they are bringing Authoring Tool for making Web Page more interactive and any update on webpage directly reflect on ADT graph and SignalR technology for real time streaming of data onto the web page.Developers are adopting Object instancing technology to make every Instance in 3D object in ADT loaded onto the 3D scene.AZURE platform is making Digital twin Definition language (DTDL) more flexible to adapt to user specific needs and making it a common language in Azure ecosystem.

6. REFERENCES

[1] A. Fuller, Z. Fan, C. Day and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," in *IEEE Access*, vol. 8, pp. 108952-108971, 2020, doi: 10.1109/ACCESS.2020.2998358.

[2] J. Wu, Y. Yang, X. Cheng, H. Zuo and Z. Cheng, "The Development of Digital Twin Technology Review," *2020 Chinese Automation Congress (CAC)*, 2020, pp. 4901-4906, doi: 10.1109/CAC51589.2020.9327756.

[3] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira and P. Leitão, "Digital Twin in Industry 4.0: Technologies, Applications and Challenges," *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, 2019, pp. 721-726, doi: 10.1109/INDIN41052.2019.8972134.

[4] F. Tao, H. Zhang, A. Liu and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405-2415, April 2019, doi: 10.1109/TII.2018.2873186.

[5] Wang, Zongyan. (2020). Digital Twin Technology. 10.5772/intechopen.80974.

[6] A. El Saddik, F. Laamarti and M. Alja'Afreh, "The Potential of Digital Twins," in *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 3, pp. 36-41, May 2021, doi: 10.1109/MIM.2021.9436090.

[7] P. V, L. P, N. J. Jain, N. L P, N. H and B. K, "Digital Twin Technology: A Bird Eye View," *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2021, pp. 1069-1075, doi: 10.1109/ICIRCA51532.2021.9545020.

[8] G. Mylonas, A. Kalogeras, G. Kalogeras, C. Anagnostopoulos, C. Alexakos and L. Muñoz, "Digital Twins From Smart Manufacturing to Smart Cities: A Survey," in *IEEE Access*, vol. 9, pp. 143222-143249, 2021, doi: 10.1109/ACCESS.2021.3120843.

[9] Getting started with Hoops communicator application guide.

[10] Visualizing ADT in 3D by Jonathan Girior published on 29/09/2021.

[11] [https:// github.com/Azure-Samples/digital-twins-explorer](https://github.com/Azure-Samples/digital-twins-explorer)

[12] ADT explorer user guide.

[13] https://docs.microsoft.com/en-us/samples/azure-samples/digital-twins-explorer/digital-twins-explorer/