International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/22312803/IJCTT-V68I4P118
Volume 68 Issue 4, 100-106, April 2020

Original Article

Development of Arduino Based Real Time Bus Tracking and Monitoring System

Izzeldin I. Mohd[1], Chong Yew Kent[2], Nazar Elfadil[3]

[1,2] *College of Engineering, Universiti Malaysia Pahang, Razak, 26300 GambangKuantan, Pahang, Malaysia*
[3] *College of Computing, Fahad Bin Sultan UniversityTabuk, Kingdom of Saudi Arabia*

Received Date: 10 March 2020
Revised Date: 27 April 2020
Accepted Date: 29 April 2020

Abstract *- Recently, the urban population growth and increased rapidly. Currently, more than half of the world's population live in cities, and it is projected that by the year 2050, more than 2.5 billion will be added to them, mostly in Asia and Africa. This growth required efficient public transportation systems such as buses where the users need sufficient and accurate information of the arrival time of the particular bus to a particular station to enable them to plan their journey from and to their home. However, the company provides a bus schedule but unfortunately, the bus schedule is not that accurate for the users to follow. To address this issue, this paper aims to develop an IOT Arduino based on a real-time bus tracking and monitoring speed system using a GPS module, which functions informed the users of the current location of the buses via the pre-installed Android Apps called Blynk. Blynk apps showed the user the estimated the buses distance and arrival times to the station, which facilitated them in the process where they can monitor the buses locations across the world via internet connection. The developed system is capable of monitoring the bus location, speed, estimated distance and time, as well as providing real-time information to the bus company and to the passengers. The system performance was evaluated experimentally and showed excellent results, indicating its ability for tracking and monitoring.*

Keywords *- Tracking, Monitoring, System, Development, Microcontroller.*

I. INTRODUCTION

The world is changing at a rapid pace, driven by technological innovation, which is reshaping the world faster than ever before. The fast development of innovation and hardware has changed our lives from an essential errand of flicking the change on to the most muddled assignment of substantial machining [1, 2]. Today's cities have become increasingly automobile-dominated, where everyone is rushing to reach their destinations, which leads to transport-related challenges such as public transport weakening. Congestion and accidents. Figure 1 shows that the average daily traffic was increasing every year from 2008 to 2017, resulting in a traffic jam during busy hours between 7 am to 9 am and 5 pm to 7 pm on weekdays will cause people to be late to their destinations. According to a study done by Boston Consulting Group (BCG) known as Unlocking Cities, the analytics showed that people in Kuala Lumpur spend about 53 minutes stuck in a traffic jam every day [2, 3].

Efficient and reliable public transport is essential to the economic growth of urban where, for the majority of people, public transport is the main means to access employment, education, and public services [4]. Therefore, for those people who always rely on public transport in their life, they are mostly concerned about the real-time location of the bus which they are waiting for, and the time it will take to reach the bus stop. By knowing the time taken to reach, they can make better travelling decisions. Furthermore, bus tracking and monitoring system for the school bus provides safety for the students that enable the parents and school authorities to track the location of the bus as well as the authorities can be able to monitor the speed of the bus to assure that the driver is not in dangerous driving, due to the dearth of research in tracking and monitoring systems [5, 6].

This paper developed a real-time public transport tracking and monitoring system using a GPS module. The developed system monitored the bus location, speed distance and estimated arrival time. The paper is organized as follows: In Section 2, the methodology is presented. Section 3 presents the experimental results, and the system performance evaluation results are presented. Finally, In Section 4, the paper conclusion is presented.

II. METHODOLOGY

This section explains the overall process and approach of developing the proposed bus tracking and monitoring system. The prototype and the block diagram of the system are shown in Figure 2 and Figure 3, respectively.

JADUAL 1.13: PURATA TRAFIK HARIAN (ADT) DI 63 LOKASI, SEMENANJUNG MALAYSIA, 2008-2017
Table 1.13: Average Daily Traffic (ADT) at 63 Locations, Peninsular Malaysia, 2008-2017

BIL *No*	STESEN *Station*	KM	LOKASI *Location*	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
PERAK													
1	AR 101	106.6	Ipoh-Tanjung Malim (Slim River Toll house)	13,854	15,038	15,322	14,039	14,431	15,328	16,501	16,736	15,187	14,913
2	AR 204	78.9	Ipoh-Lumut	21,138	23,920	22,389	23,758	25,283	25,942	23,834	26,494	25,939	27,384
3	AR 301	35.9	Ipoh-Kampar	23,050	23,659	31,022	30,311	27,497	28,520	27,126	30,592	24,654	29,911
4	AR 303	5.6	Ipoh-Gopeng	71,205	79,513	84,135	73,487	85,819	77,195	78,210	78,136	77,483	87,006
5	AR 501	30.4	Ipoh-Kuala Kangsar (500m North of Sg. Siput Town)	18,036	19,085	19,426	22,500	19,251	20,478	18,578	19,895	19,936	20,443
6	AR 601	79.7	Ipoh-Batu Hampar-Changkat Jering	16,091	16,752	14,128	11,726	19,410	20,588	19,881	20,664	21,327	19,058
7	AR 603	82.1	Ipoh-Changkat Jering-Semanggol	10,034	7,076	10,270	18,646	12,586	12,938	12,531	12,949	13,919	9,227
8	AR 703	106.3	Ipoh-Teluk Intan-Simpang Empat	24,244	25,159	25,365	25,004	28,465	25,489	25,865	27,249	33,024	26,218
9	AR 801	96.6	Ipoh-Kuala Kangsar-Gerik	2,509	2,715	1,724	3,110	3,562	3,487	2,826	3,139	3,017	2,798
10	AR 803	2.4	Lebuh Raya Timur-Barat	2,945	3,078	3,337	3,148	3,944	3,796	3,636	3,293	3,818	3,923
SELANGOR													
11	BR 102	3.5	Klang-Port Klang (Jalan Watson)	47,817	46,690	52,901	53,983	57,835	56,513	52,563	49,706	47,687	47,521
12	BR 108	-	300m From The Federal Highway's Junction-North Klang Straits Bypass	129,408	122,468	141,121	135,678	140,780	143,223	136,424	131,348	128,988	101,893
13	BR 203	48.3	Klang-Morib-Batu Laut	6,711	7,023	7,101	6,852	7,128	7,777	7,741	7,890	7,999	5,903
14	BR 405	45.1	Kuala Lumpur-Kepong-Kuala Selangor	19,991	20,938	22,822	23,003	23,648	21,278	22,618	21,611	22,277	21,568
15	BR 501	72.1	Klang-Sabak Bernam	17,977	19,773	20,466	21,743	23,370	22,174	22,443	20,838	24,554	21,642
16	BR 604	21.7	Kuala Lumpur-Kajang	57,286	58,929	56,609	57,591	53,689	59,023	54,222	48,737	48,199	46,607
17	BR 701	58.6	Kuala Lumpur-Kuala Kubu Baru Junction (South of Junction)	12,307	13,209	18,293	13,430	13,320	15,082	15,022	15,538	16,063	13,476
18	BR 902	19.0	Kuala Lumpur-Karak Highway	137,870	150,213	129,345	144,334	151,486	147,449	144,308	129,579	138,711	153,545
PAHANG													
19	CR 102	67.0	Kuala Lumpur-Kuantan (along KL-Karak Highway)	20,703	22,224	28,109	27,653	28,470	26,509	28,300	30,436	31,036	1,833
20	CR 403	233.0	Kuantan-Maran	23,395	26,064	28,248	29,142	31,138	31,416	31,876	29,856	30,231	33,247
21	CR 410	32.5	Kuantan-Kemaman	20,260	22,249	22,079	23,278	22,911	21,601	22,995	20,827	24,850	30,052
22	CR 503	282.0	Kuala Lumpur-Kuala Lipis- Kampung Padang Tuanku-Kota Bharu	6,609	6,159	6,253	7,053	7,513	6,985	7,319	8,063	8,757	8,302
23	CR 603	3.0	Pekan-Nenasi	5,746	7,783	8,650	9,761	10,319	11,563	11,535	10,741	11,620	11,606
24	CR 801	115.0	Kuantan-Karak	11,567	11,614	11,771	11,597	13,281	13,328	15,673	14,874	15,730	16,731
25	CR 805	189.0	Kuantan-Maran	6,587	6,761	6,980	8,049	7,088	6,952	6,714	6,369	6,841	6,915
26	CR 902	-	Kuantan-Segamat (400m. Bkt. Ibam-Rompin Junction North Bound)	7,316	11,634	8,102	8,427	9,545	8,763	9,314	9,985	10,514	10,416

SUMBER: KEMENTERIAN KERJA RAYA
Source: Ministry of Works

Fig.1 Average Daily Traffic (ADT) at Specific location

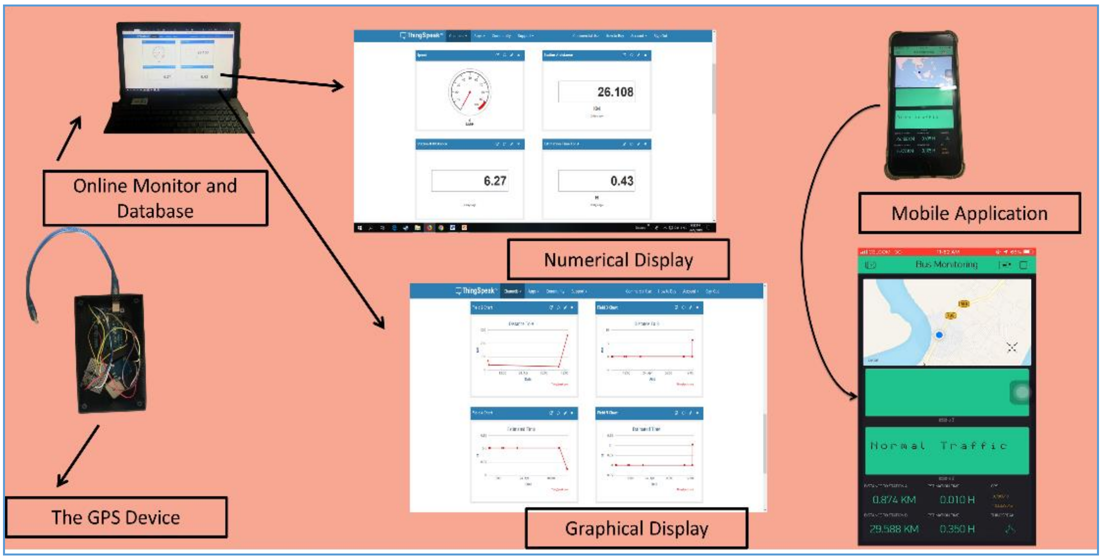


Fig.2 System Prototype

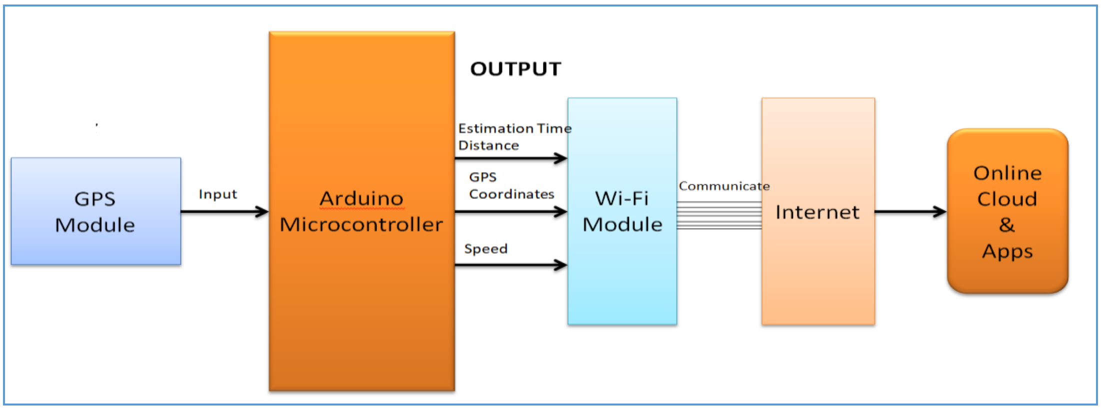


Fig. 3 System Block Diagram

Fig. 4 shows the schematic diagram representing the components connection of the system prototype. It consists of three (3) main parts, which are the Arduino microcontroller, the Wi-Fi module ESP8266 and the GPS module, uBlox Neo M8N.

The system detected the longitude and latitude, coordinated the moving bus location, and estimated its moving speed. Then, the bus distance and estimated arrival are calculated and displayed on PC at bus operator command centre through Wireless Sensor and to the users through the mobile application as shown in the flowchart in Figure 5.

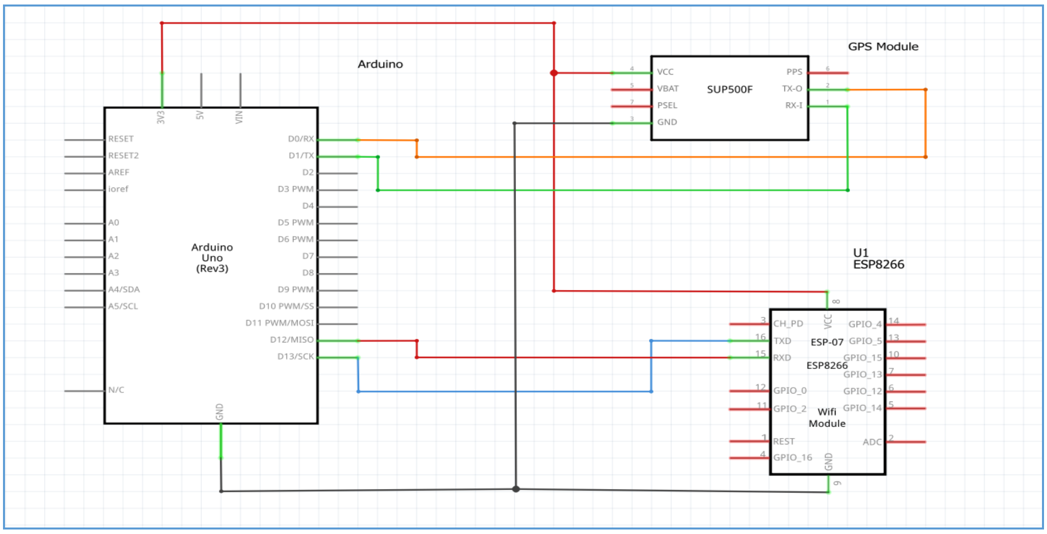


Fig. 4 System Schematic Diagram

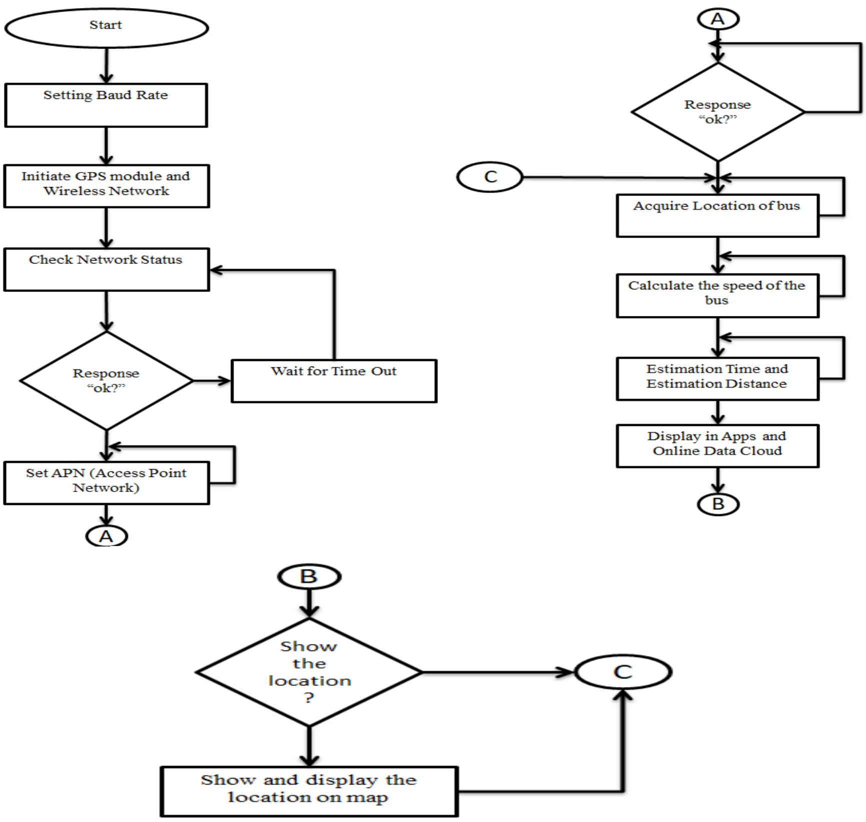


Fig. 5 System Flowchart

Meanwhile, Arduino 1.8.8 (IDE) and ThingSpeak were used for the system software development. Arduino is used to designing and developing the tracking and monitoring system while, ThingSpeak is an open-source Internet of Things application and API used to store and retrieve the tracking data using the HTTP protocol over the Internet or via a Local Area Network. The bus tracking and monitoring data captured by my Arduino microcontroller system can be stored, received and monitored from anywhere by utilizing the free IOT service provider-ThingSpeak graphical user interface (GUI) shown in Figure 6, or it is exported and downloaded as a CSV file for documentation purpose as shown in Figure 7.

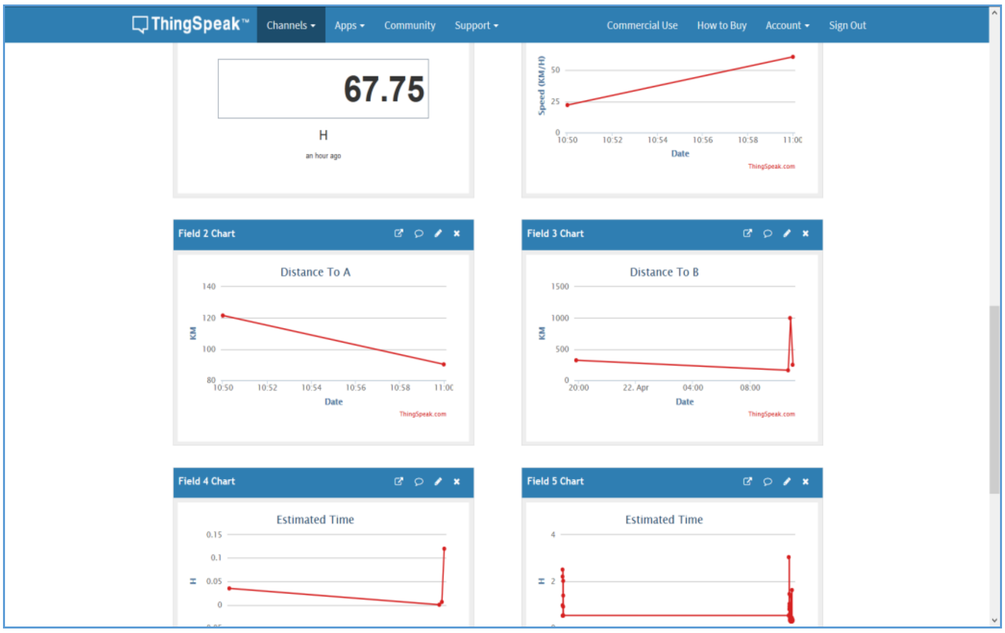


Fig. 6 System GUI Online Cloud and Data Analyse

created_at	entry_id	field1	field2	field3	field4	field5
2019-04-21 11:43:39 UTC	390	66.642	9.452	24.552	0.142	0.368
2019-04-21 11:43:54 UTC	391	76.582	9.194	24.811	0.12	0.324
2019-04-21 11:44:10 UTC	392	80.643	8.898	25.121	0.11	0.312
2019-04-21 11:44:26 UTC	393	79.415	8.611	25.433	0.108	0.32
2019-04-21 11:44:46 UTC	394	77.558	8.235	25.804	0.106	0.333
2019-04-21 11:45:01 UTC	395	37.691	8.044	26.017	0.213	0.69
2019-04-21 11:45:21 UTC	396	0.795	8.052	26.026	0.134	0.434
2019-04-21 11:45:42 UTC	397	0.942	8.052	26.026	0.134	0.434
2019-04-21 11:46:02 UTC	398	58.466	7.895	26.156	0.135	0.447
2019-04-21 11:46:18 UTC	399	77.482	7.541	26.444	0.097	0.341
2019-04-21 11:46:35 UTC	400	74.171	7.217	26.727	0.097	0.36
2019-04-21 11:46:50 UTC	401	66.022	6.966	26.944	0.106	0.408
2019-04-21 11:47:07 UTC	402	37.27	6.729	27.148	0.181	0.728
2019-04-21 11:47:27 UTC	403	42.182	6.553	27.295	0.155	0.647
2019-04-21 11:47:43 UTC	404	75.537	6.311	27.499	0.084	0.364
2019-04-21 11:47:58 UTC	405	87.329	6.004	27.733	0.069	0.318
2019-04-21 11:48:13 UTC	406	86.223	5.681	27.975	0.066	0.324
2019-04-21 11:48:29 UTC	407	86.367	5.354	28.221	0.062	0.327
2019-04-21 11:48:50 UTC	408	92.09	4.846	28.566	0.053	0.31
2019-04-21 11:49:05 UTC	409	92.867	4.503	28.796	0.048	0.31
2019-04-21 11:49:22 UTC	410	82.305	4.166	28.986	0.051	0.352
2019-04-21 11:49:37 UTC	411	90.778	3.922	29.054	0.043	0.32
2019-04-21 11:49:53 UTC	412	88.685	3.679	29.133	0.041	0.329
2019-04-21 11:50:13 UTC	413	84.031	3.299	29.39	0.039	0.35
2019-04-21 11:50:33 UTC	414	83.056	2.913	29.658	0.035	0.357
2019-04-21 11:50:48 UTC	415	77.865	2.685	29.801	0.034	0.383
2019-04-21 11:51:03 UTC	416	69.85	2.635	29.778	0.038	0.426
2019-04-21 11:51:19 UTC	417	81.906	2.468	29.906	0.03	0.365
2019-04-21 11:51:34 UTC	418	66.506	2.332	30.028	0.035	0.452
2019-04-21 11:51:49 UTC	419	73.472	2.11	30.253	0.029	0.412

Fig. 7 Export CSV File

Public transport Bus tracking system is an IOT base project which functions is to inform the users of the current location of the buses via the pre-installed Android Apps called Blynk. Blynk apps will show us to estimate the buses distance and their arrival times to the station, which will facilitate us in the process where we can monitor the buses locations across the world via internet connection [7]. After obtaining the longitude and latitude coordinates of the moving bus location, Arduino will send the sensor data to the Blynk server via the ESP8266 wifi module. Blynk apps will display values of the distance to station, estimation time and speed, as shown in Figure 8.

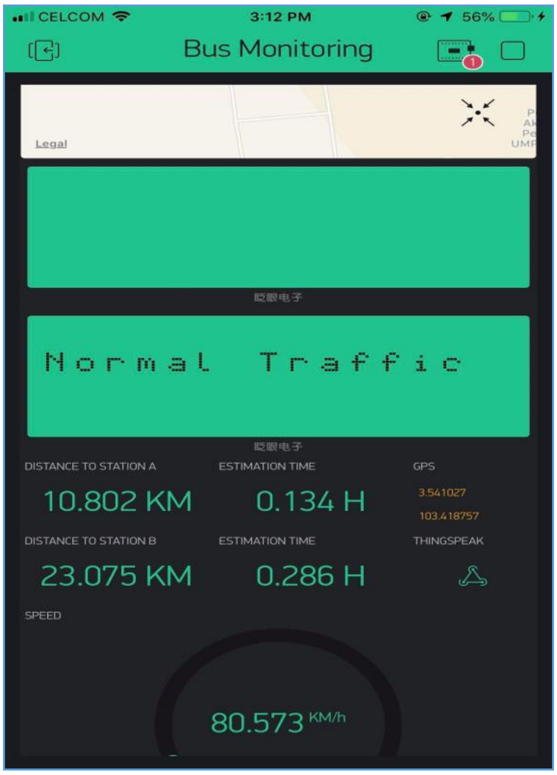


Fig. 8 Design of Blynk-App

Mobile-App speed and security are of huge importance. Hence, when creating a web service, one of the most important things is choosing the right authorization method [8]. Therefore, to ensure data transmission security Auth Token is used. Auth consists of sending the users an authorization request with credentials to issue a random token from the authorization server. This is sent to the resource server to verify if the user is authorized to use the resource and perform specific operations.

III. RESULTS AND DISCUSSION

This section presents the experimental results, findings and discussion of the work. The experiments are conducted on the bus route from Kuantan town to Universiti Malaysia Pahang (UMP), Pekan campus, as shown in Figure 9.

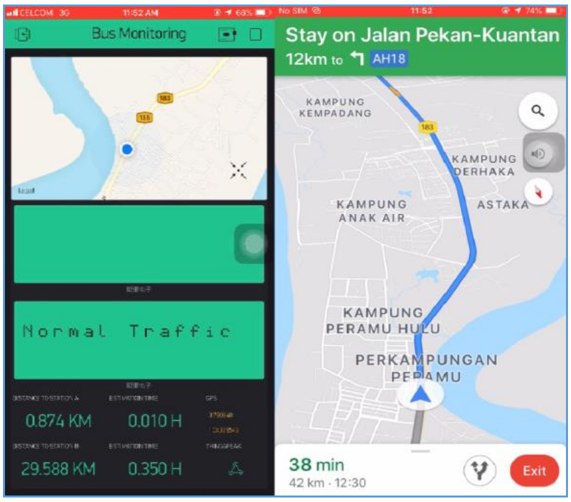


Fig. 9 The Route from Kuantan to UMP

First, we ensured that the proposed system was successfully connected to the Wi-Fi and ready for updating the information data by displaying the word Ready in the serial monitor, as shown in Figure 10. Next, the application started to transmit the bus location data where the bus location coordinates the altitude, the speed and the estimated distance and time are displayed on the serial monitor, as shown in Figure 11. This information is updated every five (5) seconds.

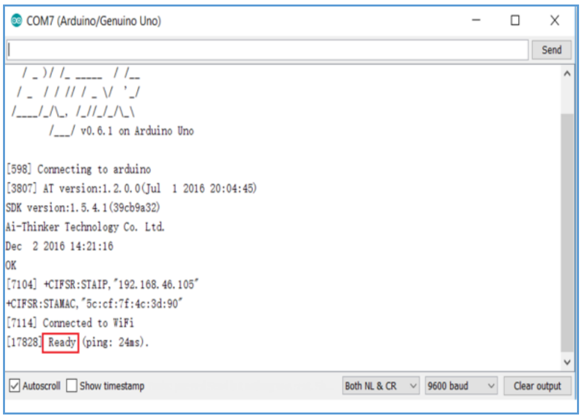


Fig. 10 System Successfully Connect to Wi-Fi

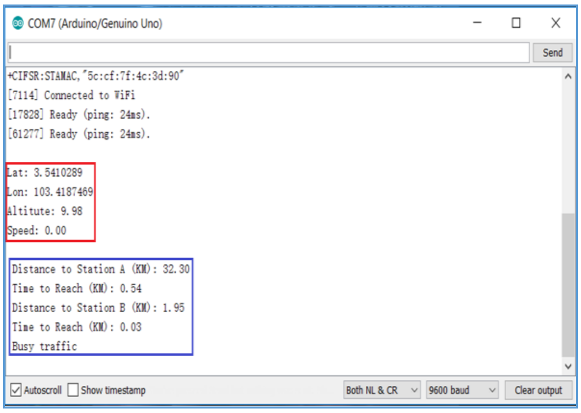


Fig. 11 Transmitted Bus Information

Next, the real-time transmitted data is displayed numerically on the ThingSpeak dashboard as shown in Figure 12, or it can be exported and saved in Microsoft Excel format for documentation purposes.

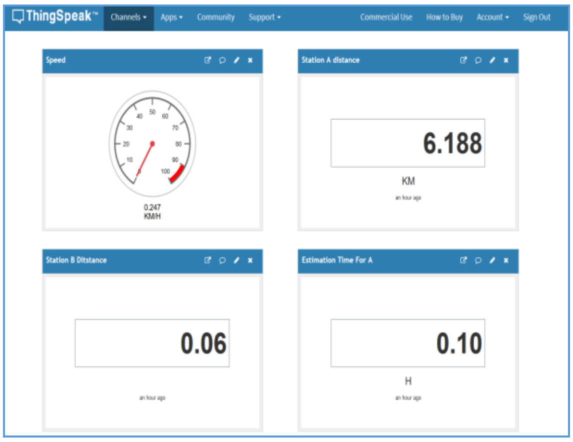
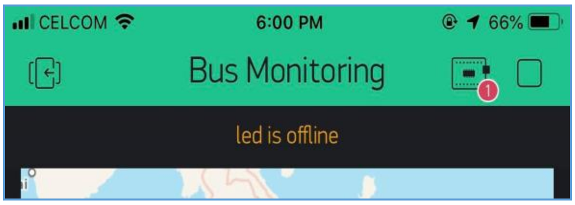
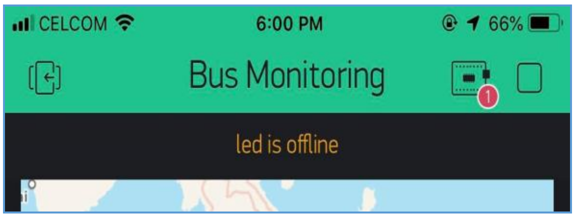


Fig. 12 ThingSpeak Dashboard

The passengers displayed the real-time information of the bus on their smartphones using Blynk apps. Blynk has two operating modes denoted ON and OFF, as shown in Figure 13 (a, b). ON operating modes indicated that the system is connected to the application, while the OFF operating mode indicated that the system is not connected to the application.



(a)



(b)

Fig. 13 Blynk Operating Modes

Blynk application displayed the real-time updated information begins when the bus starts moving. All bus stations' coordinates along the bus route are set and stored. Hence, when the bus reaches around 300 meters from the station, the Blynk application will notify the users by displaying Reach Station. If the bus speed is more than 30km/h, this indicates that there is normal traffic, as shown in Figure 14. However, if its speed is less than 30km/h, it means that the road is congested and there is heavy traffic, as shown in Figure 15.

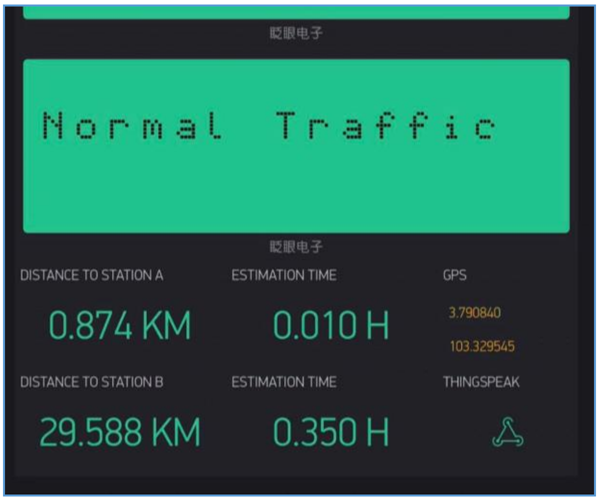


Fig. 14 Blynk Notification for Normal Traffic

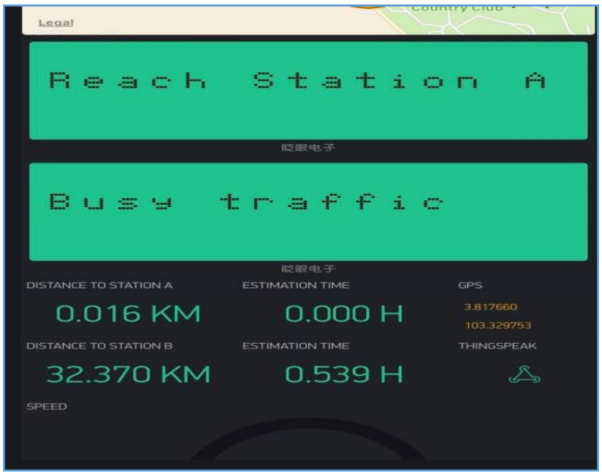


Fig.15 Blynk Notification for Reaching Station and Heavy Traffic

The system performance is evaluated by running an experiment for five (5) times between RTC Kuala Pahang bus station and UMP main entrance bus station and estimated the bus arrival time using the proposed system and compared it with the actual arrival time as shown in Figure 15. The system shows excellent results, which indicates the proposed system capability to estimate the bus arrival time.

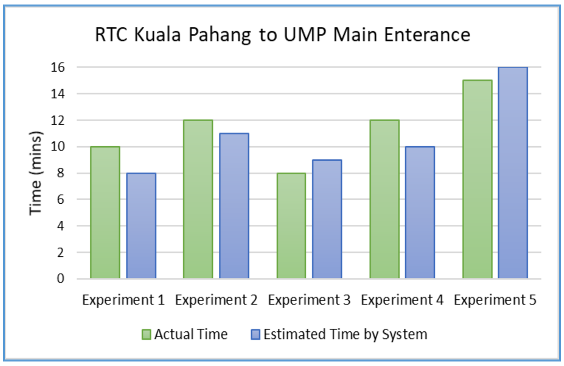


Fig. 16 Arrival Time Estimation

IV. CONCLUSION

This work describes and presents the development and implementation of an IOT bus tracking and monitoring system to tell us the bus location, speed distance and estimated arrival time to the station through an Android Apps called Blynk. Algorithms for tracking and monitoring are implemented using Arduino IDE on a microcontroller. The microcontroller sent the obtained data to the Blynk server via the ESP8266 Wi-Fi module. Experimental results show the proposed system capability to track and monitor the bus location. For future work, in order to improve the system, we recommended that the bus operators required developing their database management system because the database is provided freely by ThingSpeak able to store data for only two (2) days.

REFERENCES

[1] Malaysia, M. o., Transport Statistics Malaysia 2017. Malaysia: Ministry of Transport Malaysia, (2018).

[2] Shrenika R M, Swati S Chikmath, Dr Ravi Kumar A V, Mrs Divyashree Y V, Mrs Roopa K Swamy., Non-Contact Water Level Monitoring System Implemented using LabVIEW and Arduino., International Conference on Recent Advances in Electronics and Communication Technology. Bangalore, India: IEEE, (2017).

[3] Supriya A Salunke, Vitthal B. Jagtap, Avinash D Harale. Vehicle Tracking System for School Bus by Arduino, International Research Journal of Engineering and Technology (IRJET), 04(03) (2017).

[4] Ling, N. S. (2018). Utilizing On-Board GPS in City Buses to Determine Traffic Conditions, Journal of Telecommunication, Electronic and Computer Engineering (JTEC), (2018).

[5] M. T. Kamisan, A.A.Aziz, W.R.W.Ahmad, N. Khairudin. (2017). UiTM Campus Bus Tracking System using Arduino Based and Smartphone Application., IEEE 15th Student Conference on Research and Development (SCOReD), (2017).

[6] P. John Samuel, P. C. Naveen Shri, S. Ravikumar., Digitization of Speedometer Incorporating Arduino and Tracing. Int, Journal of Engineering Research and Applications, 6(3) (2016) 14-18.

[7] Grand View Research, I., Global Positioning Systems (GPS) Market Size, Share & Trends Analysis Report By Deployment, By Application (Aviation, Marine, Surveying, Location-Based Services, Road), and Segment Forecasts, 2018 - 2025. Grand View Research, Inc, (2018).

[8] Abdul Haleem SL, Sabraz Nawaz Samsudeen., Real-Time Bus Tracking and Scheduling System using Wireless Sensor and. Research Gate, (2016).