Electronic Subsystem for Fleet Data Acquisition at Fleet Monitoring and Control System for Guided Bus

Ali Zaenal Abidin[1], Arief Syaichu R.[2], Arif Sasongko[3], Agung Darmawan[4]

Electrical Engineering Study Program, School of Electrical Engineering and Informatics, Institut Teknologi Bandung
Ganeca 10, Bandung, 40135, Jawa Barat, Indonesia

Email: [1]alizaenaal@gmail.com, [2]arief@stei.itb.ac.id, [3]asasongko@gmail.com, [4]agungdar68@yahoo.com

***Abstract*— Traffic jams and less-popular public transport as main transportation are the problems faced by Bandung. Guided Bus as a mass public transportation can be a solution for the problem. Guided bus running on special lane isolated from other vehicles will be an on-time, pleasant mass public transportation for the passengers. Guided bus consists of many subsystem. In this final project, one of the subsystem, fleet monitoring and control system (FMCS), was made as a prototype. FMCS is a communicarion system for fleets and control station to manage fleet schedulings with fleet positions and conditions. This system consists of electronic, server and GUI subsystems. This book contains informations about design, implementation and verification of electronic subsystem in FMCS for guided bus. Electronic subsystem is a part of FMCS for sending rpm, battery, fault and position data from fleet to control station, and display message from GUI at control station. During implementation process, electronic subsystem can do its functions. Testing results show that electronic subsystem can successfully satisfy the specifications with 88.48 % of GPS data error less than 6 meters and GSM latency at 76.1 ± 0.3 ms with 1.3% of the data have higher latency than the specification. For better latency performance, SIM5215A or Telit UC864-E are both capable to work in 3G network and can be used over SIM900. For better GPS data, can be used GPS module with better accuracy.**

Keywords— ECU, FMCS, CAN.

I. Introduction

Traffic jam is one of the problem in Bandung. In 2014, this problem wasted Rp 360 billion per month[1]. This loss was caused by wasted fuels while vehicles stopped. Public transportation should be a good solution for this. But, according to department of transportation, public transportation turns out to be one of the cause of traffic jams with case such as tariff deviation and breaking traffic laws. On top of that, public transportation usage in Bandung is not maximized yet, proven by private vehicles be the 96% of total vehicles in Bandung[2]. Because of this, we need a public transportation system that orderly obey traffic laws and reduce private vehicles on road.

Guided bus as mass public transportation is one of the solutions that can solve this problem. Guided bus is a electric vehicle bus operated in an isolated lane; separated from other vehicles. Bus is used because this vehicle can load up to 60 people.

To complement guided bus system, another system for monitoring and controlling guided bus fleets are needed. This system is called fleet monitoring and control system (FMCS). FMCS acted to monitor and control fleet position and condition in a scheduling algorithm. FMCS consists of electronic subsystem, server and a GUI.

This paper will explain electronic subsystem that collect and send fleet data to server, also receive and display instructions from GUI by internet.

II. Electrical Fleet Monitoring and Management

Below are some literature studies about electric fleet monitoring and management to support electronic subsystem of FMCS design and implementation.

A. *Fleet Position Monitoring*

Fleet position monitoring can be done with some methods. Some examples are with GPS[3] and RFID[4].

In the case of RFID, a tag is used in the fleet and there are readers on the road. The advantage of this method are very precise. But, this method have poor accuracy and need infrastructures to setup readers on road. This method also has poor performance on crowded area, because 433 MHz radio frequency can be interrupted and reader-tag communication fails frequently.

In the case of GPS, a GPS receiver and antenna are used. The advantage of this method is more accurate than RFID method. On top of that, this method is easier to implement because no need of infrastructures. But, GPS module has worse precision than RFID and need time before it can lock the location.

B. *Controller Area Network (CAN)*

CAN is a bus standard for electrical vehicles for communication between engine control uni (ECU), with a high speed up to 1 Mbps[5]. CAN Bus usually implemented with 2 twisted differential wires, CAN high and CAN low, with 2 120 Ω resistors at its ends.

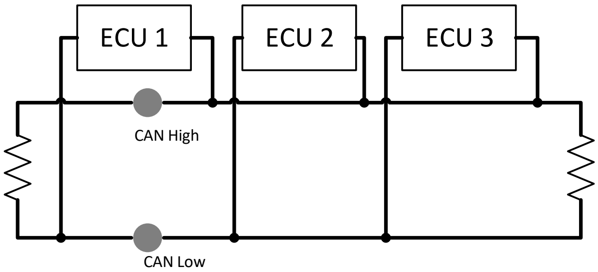

Figure 2. 1 CAN Bus

CAN Bus consists of 2 layers, physical and data link. Physical layer acts to ensure physical connection of every nodes in the network and standardize electrical bus characteristics. Data link layer allows all nodes to send and receive data in the bus and hold informations to identify data frame and error.

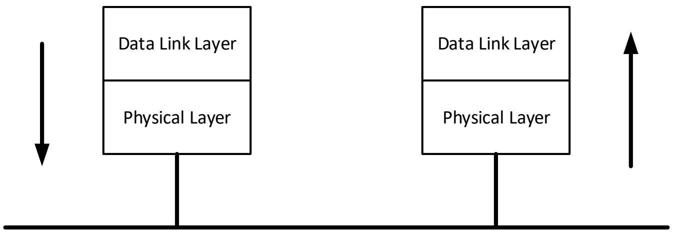

Figure 2. 2 CAN Bus communication

C. *Global Positioning System (GPS) and National Marine Electronics Association (NMEA) Sentences*

In the implementation, GPS used NMEA standard[6] to send GPS data such as speed, position and time. NMEA standard consists of sentences with unique data meaning. Every sentence begins with '$' character, terminated with '\r\n' character and use ',' to split data. For GPS, data have GP as its prefix.

Tabel 2. 1 NMEA Sentences

Message	Description
GGA	Time, position and fix type data
GLL	Latitude, longitude, UTC time of position fix and status
GSA	GPS receiver operating mode, satellites used in the position solution, and DOP values
GSV	Number of GPS satellites in view satellite ID numbers, elevation, azimuth, & SNR values
MSS	Signal-to-noise ratio, signal strength, frequency, and bit rate from a radio-beacon receiver
RMC	Time, date, position, course and speed data
VTG	Course and speed information relative to the ground
ZDA	PPS timing message (synchronized to PPS)
150	OK to send message
151	GPS Data and Extended Ephemeris Mask
152	Extended Ephemeris Integrity
154	Extended Ephemeris ACK

GPS uses serial communication protocol to communicate with controller.

D. *Message Queuing Telemetry Transport (MQTT)*

MQTT first invented in 1999 by Dr Andy Standford-Clark from IBM and Arlen Nipper from Arcom (now Eurotech). MQTT is an ISO standard (ISO/IEC PRF 20922) as information sending protocol to send lightweight information based on publish/subscribe, designed for devices with low bandwidth communication, high latency or located in poor network location. This protocol fits internet of things or machine to machine communications.

MQTT client that publish data is called publisher, and the one that receive data is called subscriber. Between them, there is a broker that passes data from publisher and distribute them to all subscribers.

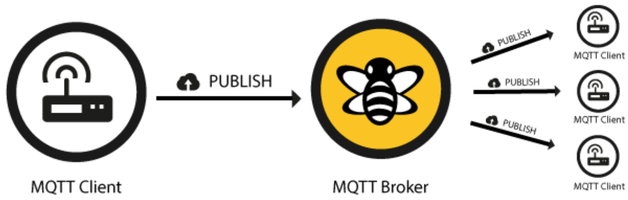

Figure 2. 3 MQTT data flow

Below some important parameters used in electronic subsystem implementation.

- Topic, information topic that published/subscribed.
- Payload, the message content.
- Quality of Service (QoS), level of receiving data confidence.

III. FMCS for Guided Bus in General

FMCS for guided bus design have architecture with GPS method. Below is the architecture and data flow diagram.

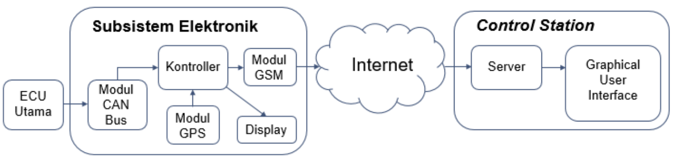

Figure 3. 1 FMCS architecture

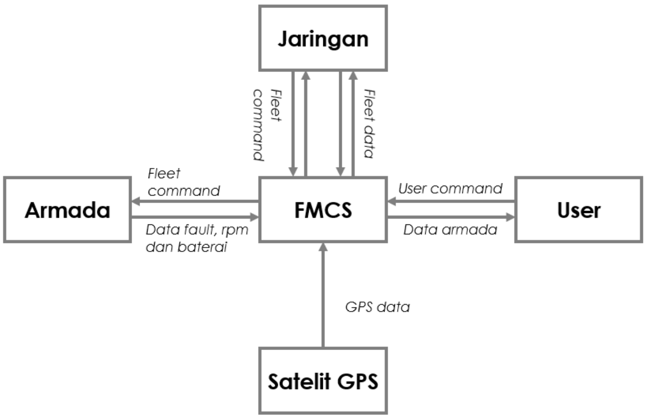

Figure 3. 2 Data flow diagram level 0 FMCS

Data stream in FMCS consists of fleet condition, position and instruction from control station. Fleet condition and position originated from sensors in guided bus and electronic subsystem of FMCS. These data then constructed and sent to GUI at control station via server and displayed. Thereafter, GUI will respond by sending fleet command to each fleet in the form of speed. This instruction is based from input data and scheduling alghoritm. Electronic subsystem then display the instruction for the driver to see.

IV. SPECIFICATION, DESIGN AND IMPLEMENTATION

Electronic subsystem need to fulfil these specifications below

- Read fleet position with maximum 6 meters of error.
- Read rpm, fault condition and battery energy from existed ECU in guided bus.
- Send fleet data with maximum interval of 0.8 seconds.
- Receive and display instructions from GUI each time a new instruction is available.
- Send sign to control station if the fleet cannot be used.

To obtain position error specification, lane display at GUI are segmented every 12 meters (the same as bus's length). From this segment, a maximum error can be measured as 6 meter, so that there are no 2 fleets displayed in 1 same segment.

To obtain send data interval, an assumption of fleet speed at 15 m/s is used. Send data interval must use segmentations (12 meters) size as a variable so bus still monitored while passed every segment. As a result, data must be sent every 12/15 seconds, or 0.8 seconds.

Below is data flow diagram and program flowchart of electronic subsystem.

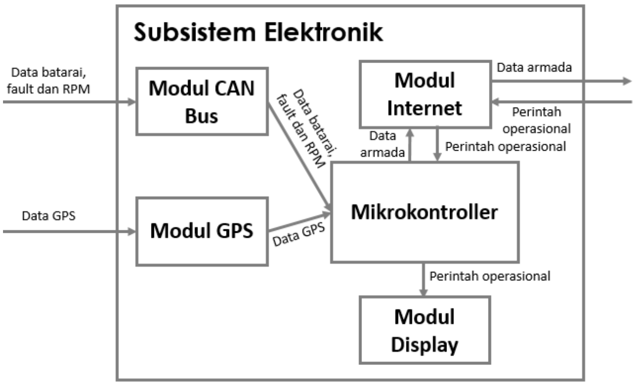

Figure 3. 3 Data flow diagram

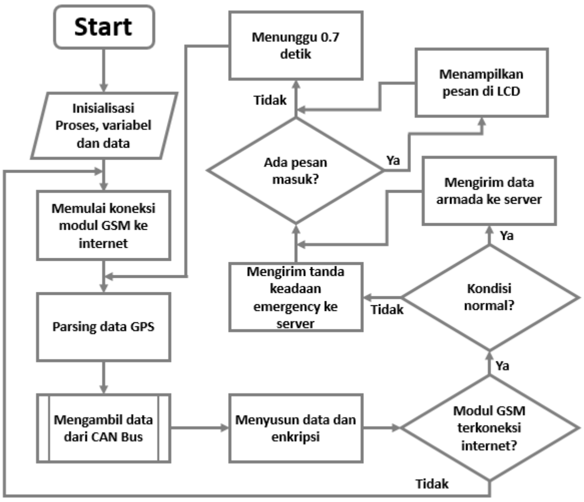

Figure 3. 4 Flowchart program

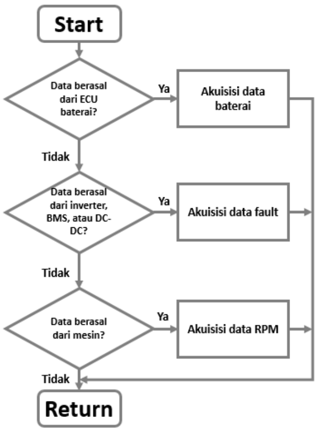

Figure 3. 5 CAN Bus data acquisition flowchart sub-program

To execute program to run this flowchart, a sequential machine is needed. Because there are no heavy processes such as image processing, there is no need to use a highspeed processor and thus AVR based processor is enough. In this design, ATMega2560 based Arduino Mega was chosen. It has 16 MHz clocspeed, 4 serial ports, an SPI and an I^2C for communication.

Below is design and implementation for each function needed.

A. *Position Data Collection with GPS*

There are many GPS sensors in the market. GPS Ublox Neo-M8N is one of the newest sensor producted at the time this paper written. This GPS sensor is suitable for usage in Indonesia because it can detect GLONASS satellite, an influential satellite for GPS data precision around Indonesia. On top of that, this sensor has an update rate up to 5 Hz and 2.5 meters accuracy. To complement this GPS sensor, an active antenna is used to catch GPS signal more effectively than passive antenna. This antenna is connected to GPS sensor with a long wire so that the antenna can be placed outside vehicle for better signal quality.

In the implementation, GPS refresh rate was set to 5 Hz so it is faster than specification needed. Moreover, GPS baudrate was set to 57600 bps to make position data collecting faster.

B. *Fleet Condition Collecting by CAN Bus*

To communicate with other ECU in guided bus, a CAN Bus communication component is needed. One of the component that can do this is MCP2515 microchip. This chip can read CAN Bus data up to 1000 kbps. This chip also has 2.7-5.5 V operating voltage, suitable for Arduino. Moreover, MCP2515 can communicate with Arduino by SPI and serial communication.

In this implementation, MCP2515 only used for CAN Bus data reading with speed of 1000 kbps. The data are located in ID 1, 3 and 5 in the CAN Bus. These ID each contain fault data, battery and rpm. Afterwards, MCP2515 will pass those data to Arduino by SPI communication.

C. *Normal and Emergency Condition*

In order to decide normal and emergency condition, a switch is used and operated by the driver of guided bus. In a normal condition, switch placed in normal mode. In this condition, electronic subsystem will send fleet data to control station. In an emergency situation, switch placed in emergency mode. In this situation, electronic subsystem will send emergency message to control station.

D. *Data Transmission and Receiving by Internet*

Fleet data will be constructed in this format below.

10 byte longitude	:	11 byte latitude	:	2 byte battery data	:	2 byte RPM	3 byte fault

For sending and receiving data via internet, GSM SIM900 module is used. This modul can open internet connection with GSM network, suitable for usage of moving vehicles. Communication protocol used for this system is MQTT.

GSM SIM900 module was chosen because it has low power consumption (1-2 watt when sending data, 1 mA idle[7]), can communicate with controller by serial protocol and has quad-band work frequencies. SIM900 data rate vary depends on signal quality.

Electronic subsystem always monitor internet connection. When lost connection, a reconnection will be done to reopen the connection to server.

Data transmission will be done in QoS = 0, a one way transmission, to make the transmission faster. This QoS was chosen because if a data failed to be sent, the next data have higher priority to send compared to the last data. Topic used in this system is “fleet” followed by fleet number. As example, for first fleet the topic will be “fleet1”.

Data receiving will be done in QoS = 2, a two way transmission that ensure subscriber receive every published data. This QoS was chosen because instruction from control station only sent once, so it must be ensured every fleet driver receive the instruction.

E. *Electronic Subsystem Integration*

Every component used in electronic subsystem will be combined as one and a packaging.

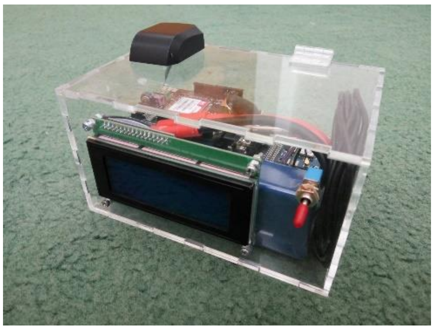

Figure 3. 6 Elektronic subsystem

Li-Po battery and casing is used for testing purposes. Active antenna placed outside casing for better GPS signal quality. LCD display are used fro debugging and simulate instruction interface from control station.

V. Testing and Verivication

Electronic subsystem testings and verivication was done to test every specification. Below are testing configuration for the verivications.

- Laptop Intel i5-6200U 2.8 GHz, RAM 4 GB, operating system Windows 10 as client to measure interval between data received.
- Electronic subsystem power from Li-Po battery, 5000 mAh 20C.
- CAN Bus from Mini AGT at PT. LEN Industri.

A. *GPS Data Acquisition*

In this test, GPS data accuration was observed to measure the error. This test was done by locating electronic subsystem in a test point that the coordinate has been known. A program then recorded poistion data sent by electronic subsystem to measure the distance with test point. From 4148 data, below is distribution graph of GPS error data.

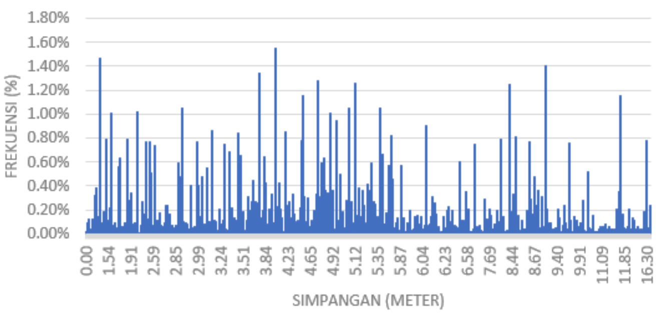

Figure 4. 1 GPS data error distribution

From this figure can be concluded that error distribution was not normally distributed. From all valid data, 11.52 % has error more than specification needed, so can be concluded that this GPS satisfy 88.48 % of location error specification needed.

B. *Can Bus Data Acquisition*

In this test, a communication between electronic subsystem and other ECU in guided bus by CAN Bus was observed. Electronic subsystem was connected to CAN Bus in Mini AGT PT Len Industry to observe battery data in the appropriate ID explained in implementation section. Below is reading result from CAN Bus Mini AGT.

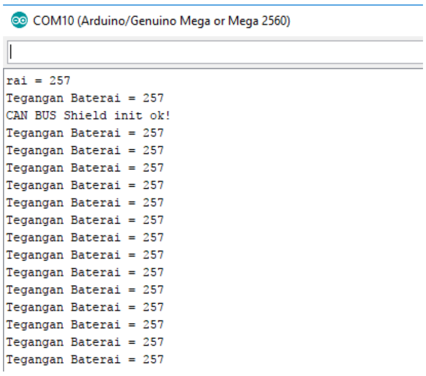

Figure 4. 2 CAN Bus data reading for battery

C. *Data Receiving from Server*

in this test, MQTTLens was used as dummy data sender. Bloe is MQTTLens interface and display result after data was sent from MQTTLens.

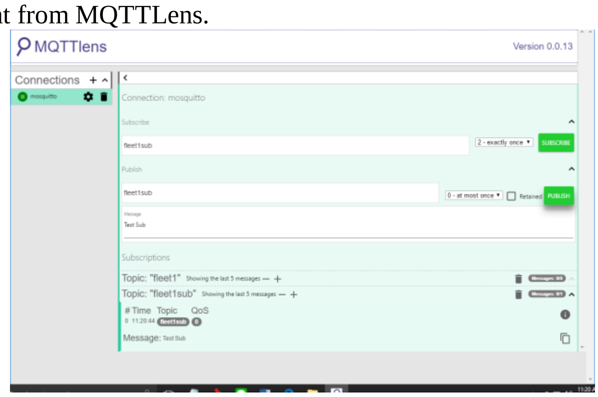

Figure 4. 3 MQTTLens

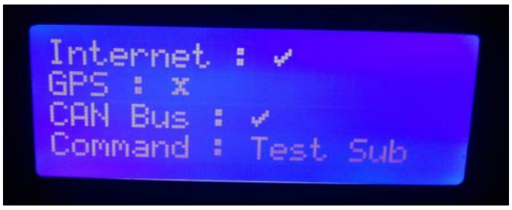

Figure 4. 4 Data display from server

From both figures can be seen that data sent from MQTTLens can be displayed by electronic subsystem.

D. *Field Test*

In this field testing, electronic subsystem was used to send moving fleet position data in Soekarno-Hatta street, Bandung. In this test, motorcycle was used to simulate vehicle. As long as electronic subsystem send position data, an operator monitored the GUI to observer data sent. In addition to this normal situation, emergency situation also tested to send emergency sign to server.

The result of this test is electronic subsystem can send position data with the format explained in design section.

For emergency situation, electronic subsystem also can send emergency sign to server.

E. *Data Transmission Latency*

Data transmission latency test done by sending 32 bytes of data from electronic subsystem to a mosquito server periodically 0.6 seconds for 12 hours. Then, a graph was made to show data receiving interval of server and statistic data are analysed.

From 12 hours of test, 84722 data obtained. Among these data, 1.3% data are sent more than the transmission interval. If data was sent with this latency, position read by control station will be not precise/valid.

To measure valid data latency, 45927 data with reasonable latencies are used. From these data, a distribution graph was made as shown below.

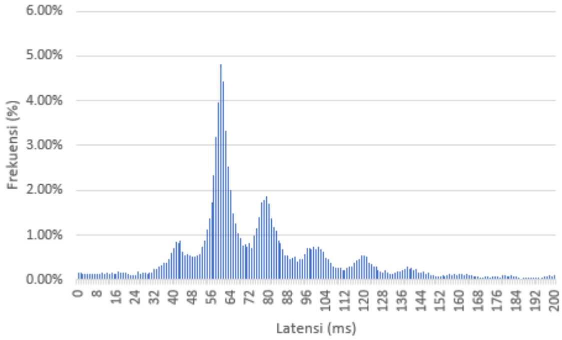

Figure 4. 5 Latency distribution

From these data, an average of 76.1 ms was obtained. This average is different from the median, 66 ms. This difference show that the distribution is not distributed perfectly normal (Gaussian). But, from above graph, the shape of latency distribution graph is close from Gaussian distribution construction shifted to the left. Because of that, statistical measurement can be done with Gaussian approach.

From these data, a Gaussian approach was used to determine statistical parameters.

Rata-Rata	Standar Deviasi	Margin of Error
76.1 ms	33.8 ms	0.3 ms

Margin of error obtained by using 95 % confidence level. From the measurement, can be said that SIM900 data transmission has 76.1 ± 0.3 ms latency with 95 % confidence level. From this value, can be shown that data sent by electronic subsystem will be received by control station at maximum interval of 600 + 76.1 + 0.3 ms, or 676.4 ms. This value is within specification.

VI. Conclusion

From the tests, electronic subsystem can collect data from other ECU in guided bus. Electronic subsystem also can collect position data from GPS with 88.48% data error less than 6 meters. These data can be sent to server by GSM network with latency of 76.1 ± 0.3 ms. This latency was still within specification, but there was 1.3% of the data have more non-tolerable latency.

In order to minimalize latency to cut non-tolerable latencies, SIM5215A or Telit UC864-E can be used over SIM900. These modules can operate in 3G network with speed up to 384 kbps. For better GPS data, GPS module with better accuracy can be used.

REFERENCES

[1] Lukihardianti, Arie. (2014, 16 November). *Kerugian Akibat Macet di Bandung Capai Rp 4,36 Triliun* [Online]. Available: http://www.republika.co.id/berita/nasional/daerah/14/11/16/nf46jk-kerugian-akibat-macet-di-bandung-capai-rp-436-triliun

[2] *Kota Bandung dalam Angka*, Badan Pusat Statistik Kota Bandung, Bandung, 2015, pp 228-230.

[3] Fabbri G. et al, "Development of an On-Board Unit for the Monitoring and Management of an Electrical Fleet" in XXth International Conference on Electrical Machines, 2012.

[4] Sriborriux, Wiroon, et al, "The Design of RFID Sensor Network for Bus Fleet Monitoring" in 8th International Conference on ITS Telecommunications, 2008.

[5] Corrigran, Steve. (2002, August). *Introduction to the Controller Area Network (CAN)* [Online]. Available: http://www.ti.com/lit/an/sloa101a/sloa101a.pdf

[6] Dale DePriest. *NMEA Data* [Online]. Available: http://www.gpsinformation.org/dale/nmea.htm

[7] SIM900 Hardware Design V2.00, SIMCom, 2010, pp. 8-9.