Performance of C-BUS Communication in CMX7032 IC with SPI Communication in ATxmega128A1 IC

1st Adelia Revani Sastaviyana
Satellite Technology Center
National Institute of Aeronautics and Space
Bogor, Indonesia
adelia.revani@gmail.com

2nd Rommy Hartono
Satellite Technology Center
National Institute of Aeronautics and Space
Bogor, Indonesia
rommy.hartono@lapan.go.id

***Abstract*—Automatic Identification System (AIS) system, that is equipped with transceiver device, works by continuously broadcasting status and position information of host vessel that enables other similar equipped vessels in the surrounding area to receive, decode, then display the information to provide maritime traffic information. One of AIS data processor IC that can be used in AIS transceiver device is CMX7032. For communicating with other IC, CMX7032 provides C-BUS interface feature that is compatible with SPI interface. ATxmega128A1, that has SPI feature and its operating voltage requirement is compatible with CMX7032, can be used to load and activate Function Image™ (FI) via C-BUS – SPI communication into CMX7032 so it can work properly. Method used to observe C-BUS – SPI communication performance is literature review and observation using various CMX7032 and ATxmega128A1 main clock frequency and C-BUS – SPI clock frequency. Observation results show that right checksum values and increased CMX7032 power consumption are successful FI loading indicators, while increased ATxmega128A1 power consumption exceeding in successful FI loading and activating case is unsuccessful FI activating indicator. Higher CMX7032 main clock frequency demands higher C-BUS – SPI clock frequency to support successful FI loading and activating process. Observation results also show that using same ATxmega128A1 main clock frequency, higher C-BUS – SPI clock frequency results less required time for loading and activating FI. Then, when using same C-BUS – SPI clock frequency, different CMX7032 main clock frequency does not affect significantly to the time required for loading and activating FI.**

Keywords*—*C-BUS, CMX7032, SPI, ATxmega128A1, Function Image™

I. INTRODUCTION

AIS is a system that provides identification, status, and position information of all other similar equipped vessels within the immediate area. This system becomes very important to the mariners and mandatory under the provisions of Safety of Life at Sea (SOLAS) Conventions, because if it is used properly, it can be functioned as vital awareness of the conditions that may affect the vessels' navigation and safety[1][2]. Moreover, because of AIS system capability to detect large numbers of ships, AIS is used in Vessel Traffic System (VTS) as port traffic management that can help coastguard and harbour authorities to control and organise shipping in their immediate area[2][3].

As a ship and shore based broadcast system, AIS operates in Very High Frequency (VHF) maritime band. AIS transceiver works by repetitively broadcasting information of host vessel's status and position that enables other similar equipped vessels in the vicinity to receive, decode, and display the information along with information from other navigational systems to provide comprehensive picture of maritime traffic in the local area[4]. AIS data can also be integrated with ship database based on its Maritime Mobile Service Identity (MMSI) to observe ship traffic patterns. The ship traffic patterns can be used for evaluating ship operation modes, such as maneuvering, hotelling, and cruising. The integrated AIS data can also be used for observing marine traffic density and ships distribution in the local area by its type and flag registered[5].

CMX7032 is an AIS data processor IC that is manufactured by CML Microcircuits and can be used in AIS transceiver equipment for encoding AIS data into AIS message[6][7]. CMX7032 has many features, such as: (1) it has half-duplex Gaussian Minimum Shift Keying (GMSK) and Frequency Shift Keying (FSK) modem, (2) it has flexible channel configuration (2 simultaneous Rx and 1 Tx), (3) it can support Self Organising Time Division Multiple Access (SOTDMA) and Carrier-Sensing Channel Access (CSTDMA) operation, (4) it has 2 RF synthesisers, and (5) it has 2 auxiliary system clock generators. With all those features, CMX7032 only needs low-power operation requirement (3.0 V to 3.6 V) and it can work with operating frequency from 9.6 MHz to 19.2 MHz[7].

Initialization routine is required to be performed before CMX7032 can work properly by loading and activating FI from external memory or microcontroller (µC)[8][9]. This process involves accessing (writing to and reading from) registers of CMX7032 through C-BUS communication protocol[7]. C-BUS is a synchronous serial µC interface developed by CML Microcircuits[10] that is very similar to Serial Peripheral Interface (SPI) communication protocol and compatible to communicate with SPI[11]. SPI is a high-

978-1-7281-3676-9/19/$31.00 ©2019 IEEE

speed synchronous data transfer interface using 3 or 4 pins (MOSI, MISO, SCK, and /SS)[12].

One of Atmel µC family that has SPI feature and 128 kB in-system self-programmable flash memory is ATxmega128A1[12][13]. Considering many commands that have to be executed in the process of loading and activating FI to CMX7032, flash memory capacity of ATxmega128A1 is expected to be sufficient. Moreover, the ability of ATxmega128A1 to work in operating frequency 0 – 32 MHz with operating voltage requirement from 2.7 V[13], is compatible with CMX7032 operating voltage requirement.

This paper is organized as follows, first, general theory about AIS, CMX7032, and ATxmega128A1. Then, description of methodology used in this paper, including literature review of C-BUS and SPI communication and observation of C-BUS communication in CMX7032 with SPI communication in ATxmega128A1 performance using various CMX7032 and ATxmega128A1 main clock frequency and C-BUS – SPI communication clock frequency. The final is observasion result and discussion.

II. METHODOLOGY

Methods used in this paper to observe performance of C-BUS communication in CMX7032 with SPI communication in ATxmega128A1 are described as flowchart in Figure 1.

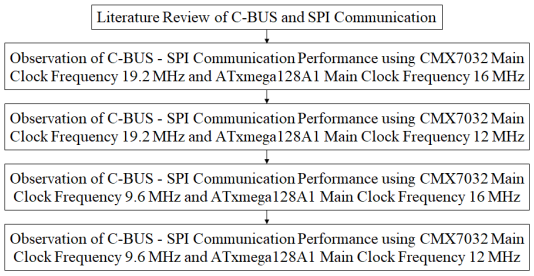

Fig. 1. Methodology Flowchart

A. *Literature Review of C-BUS and SPI Communication*

C-BUS serial interface uses 5 wires to communicate, they are: Chip Select (CSN), Serial Clock (SCLK), Command Data (CDATA), Reply Data (RDATA), and Interrupt Request (IRQN)[14]. In CMX7032, to enable C-BUS serial interface, CBUSMODE (pin 57) must be permanently connected to digital ground (DVSS). Figure 2 shows pin configuration and recommended external components of CMX7032[7].

As mentioned above, SPI serial interface, that is developed by Motorola, uses 3 or 4 wires to communicate, they are: Slave Select (/SS), Serial Clock (SCLK), Master Output Slave Input (MOSI), and Master Input Slave Output (MISO)[14][15]. In ATxmega128A1, SPI feature can be configured in PORT C, D, E, and F[13]. SPI communication clock frequency can be obtained from ATxmega128A1 main clock frequency with the value of main clock frequency devided by 2, 4, 8, 16, 32, 64, or 128[12]. Pin configuration and block diagram of ATxmega128A1 is shown in Figure 3[13]. For connecting C-BUS interface to SPI interface, C-BUS and SPI wires connection must follow configuration shown in Table I[14].

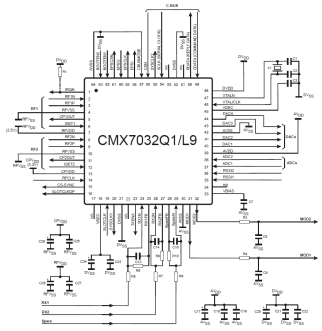

Fig. 2. Pin Configuration and Recommended External Components of CMX7032

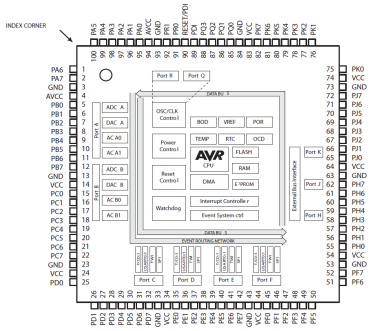

Fig. 3. Pin Configuration and Block Diagram of ATxmega128A1

TABLE I. C-BUS AND SPI WIRES CONNECTION CONFIGURATION

C-BUS Interface Line	Corresponding SPI Interface Line
Chip Select (CSN)	Slave Select (/SS)
Serial Clock (SCLK)	Serial Clock (SCLK)
Command Data (CDATA)	Master Output Slave Input (MOSI)
Reply Data (RDATA)	Master Input Slave Output (MISO)
Interrupt Request (IRQN)	None

Each C-BUS transaction consists of a single register Address/Command (A/C) byte sent by µC that can be followed by 1 or more data byte(s) to be written into CMX7032's write only register, or read out from CMX7032's read only register. The number of data bytes following an A/C byte depends on the word size of addressed register with the Most Significant Bit (MSB) of the address or data is sent first. Data sent from µC on CDATA line is clocked into CMX7032 on the rising edge of SCLK input. RDATA sent from CMX7032 to µC is valid when SCLK is high. CDATA and RDATA lines are never active at the same time, the address byte determines data direction for each C-BUS transfer. CSN line must be held low during data transfer and kept high between transfers[7]. Detailed SPI writing to and reading data from C-BUS timing diagrams are shown in Figure 4 and Figure 5[11].

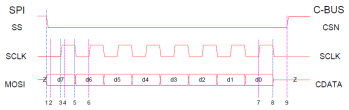

Fig. 4. SPI Writing to C-BUS Timing Diagram

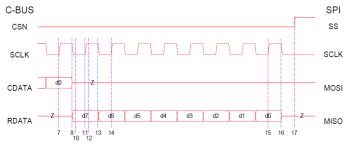

Fig. 5. SPI Reading Data from C-BUS Timing Diagram

In this observation of C-BUS – SPI communication performance, C-BUS is set as slave while SPI is configured as master[15], using SPI Mode 0[11]. There are 4 modes of SPI with different Clock Polarity (CPOL) and Clock Phase (CPHA) values settings as described in Table II, while Figure 6 shows relative timing diagram for all SPI modes[15].

TABLE II. SPI MODES

CPOL	CPHA	SPI Mode
0	0	0
0	1	1
1	0	2
1	1	3

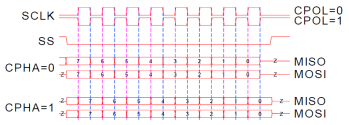

Fig. 6. Relative Timing Diagram for All SPI Modes

Successful loading FI process that results right checksum values can be one of indicators that C-BUS – SPI communication works properly. Detailed FI loading and activating from host µC process flowchart is shown in Figure 7[7]. FI used in this C-BUS – SPI communication performance observation is 7032/7042FI-1.x that is aimed for full AIS Class B operation[16]. Class B Marine AIS is usually used in light commercial and leisure markets. The units are smaller, more power efficient, and available at cheaper cost compared to Class A Marine AIS. Class A Marine AIS is targeted at large commercial vessels with full specification equipment that results in a large, heavy, and power-hungry unit. The units are also expensive, but capable to operate in extreme conditions[2].

B. Observation of C-BUS - SPI Communication Performance using CMX7032 Main Clock Frequency 19.2 MHz and ATxmega128A1 Main Clock Frequency 16 MHz

First observation uses CMX7032 main clock frequency 19.2 MHz and ATxmega128A1 main clock frequency 16 MHz, while C-BUS – SPI communication clock frequency varies from 500 kHz to 8 MHz. Figure 8 shows FI loading and activating process when CMX7032 main clock frequency is 19.2 MHz, ATxmega128A1 main clock frequency is 16 MHz, and C-BUS – SPI communication clock frequency is 8 MHz. In Figure 8, channel 1 in yellow color shows CSN signal, channel 2 in blue color shows SCLK signal, channel 3 in purple color shows CDATA signal, and channel 4 in green color shows RDATA signal. C-BUS – SPI communication wires signals capture in Figure 8 is in accordance with SPI writing to and reading data from C-BUS timing diagram in Figure 4 and Figure 5 and resulting right checksum values. It means that C-BUS – SPI communication works properly.

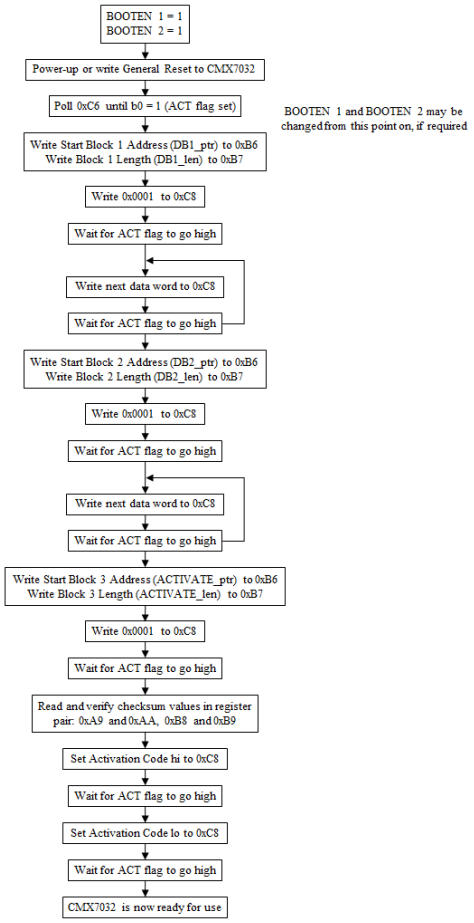

Fig. 7. FI Loading and Activating from Host µC Process Flowchart

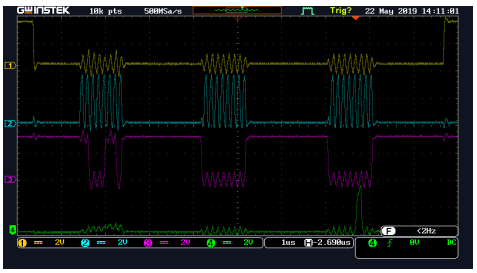

Fig. 8. FI Loading and Activating Process when CMX7032 Main Clock Frequency is 19.2 MHz, ATxmega128A1 Main Clock Frequency is 16 MHz, and C-BUS – SPI Communication Clock Frequency is 8 MHz

Observation result of C-BUS – SPI communication performance using CMX7032 main clock frequency 19.2 MHz and ATxmega128A1 main clock frequency 16 MHz, while C-BUS – SPI communication clock frequency varies from 500 kHz to 8 MHz, is shown in Table III. C-BUS – SPI communication clock frequency is obtained by deviding ATxmega128A1 main clock frequency (as master) by 2, 4, 8, 16, or 32.

Successful FI loading and activating process is indicated by right checksum values read from CMX7032's registers and all steps in FI loading and activating from host μC process flowchart described in Figure 7 are successfully executed. Meanwhile, in failed FI activation case, FI loading process is successful and resulting right checksum values, but FI activation steps process is not successfully executed.

TABLE III. OBSERVATION RESULT OF C-BUS – SPI COMMUNICATION PERFORMANCE USING CMX7032 MAIN CLOCK FREQUENCY 19.2 MHZ AND ATXMEGA128A1 MAIN CLOCK FREQUENCY 16 MHZ

CMX7032 Main Clock Freq (MHz)	**ATxmega128A1 Main Clock Freq (MHz)**	**C-BUS – SPI Clock Freq (MHz)**	**Before FI Loading and Activating Process**	**Success FI Loading and Activating Process? (Y/N)**	**After FI Loading and Activating Process**		**Time to Load and Activate FI (ms)**
			CMX7032 power consumption (mW)		***CMX7032 power consumption (mW)***	***ATxmega128A1 power consumption (mW)***	
19.2	16	8	21.5	Y	214.8	128.0	592
19.2	16	4	21.4	Y	214.8	127.5	671
19.2	16	2	21.4	Y	214.8	129.0	847
19.2	16	1	21.5	N (Activation Failed)	214.8	145.0	-
19.2	16	0.5	21.4	N (Activation Failed)	215.8	147.0	-

C. Observation of C-BUS - SPI Communication Performance using CMX7032 Main Clock Frequency 19.2 MHz and ATxmega128A1 Main Clock Frequency 12 MHz

Table IV shows second observation result that uses CMX7032 main clock frequency 19.2 MHz and ATxmega128A1 main clock frequency 12 MHz, while C-BUS – SPI communication clock frequency varies from 375 kHz to 6 MHz.

TABLE IV. OBSERVATION RESULT OF C-BUS – SPI COMMUNICATION PERFORMANCE USING CMX7032 MAIN CLOCK FREQUENCY 19.2 MHZ AND ATXMEGA128A1 MAIN CLOCK FREQUENCY 12 MHZ

CMX7032 Main Clock Freq (MHz)	**ATxmega128A1 Main Clock Freq (MHz)**	**C-BUS – SPI Clock Freq (MHz)**	**Before FI Loading and Activating Process**	**Success FI Loading and Activating Process? (Y/N)**	**After FI Loading and Activating Process**		**Time to Load and Activate FI (ms)**
			CMX7032 power consumption (mW)		***CMX7032 power consumption (mW)***	***ATxmega128A1 power consumption (mW)***	
19.2	12	6	21.5	Y	215.8	113.0	719
19.2	12	3	21.4	Y	215.2	113.0	848
19.2	12	1.5	21.5	Y	214.8	112.0	1,088
19.2	12	0.75	21.5	N (Activation Failed)	214.8	126.0	-
19.2	12	0.375	21.5	N (Activation Failed)	214.8	125.5	-

D. Observation of C-BUS - SPI Communication Performance using CMX7032 Main Clock Frequency 9.6 MHz and ATxmega128A1 Main Clock Frequency 16 MHz

Third observation uses CMX7032 main clock frequency 9.6 MHz and ATxmega128A1 main clock frequency 16 MHz, while C-BUS – SPI communication clock frequency varies from 500 kHz to 8 MHz, and the result is shown in Table V.

TABLE V. OBSERVATION RESULT OF C-BUS – SPI COMMUNICATION PERFORMANCE USING CMX7032 MAIN CLOCK FREQUENCY 9.6 MHZ AND ATXMEGA128A1 MAIN CLOCK FREQUENCY 16 MHZ

CMX7032 Main Clock Freq (MHz)	**ATxmega128A1 Main Clock Freq (MHz)**	**C-BUS – SPI Clock Freq (MHz)**	**Before FI Loading and Activating Process**	**Success FI Loading and Activating Process? (Y/N)**	**After FI Loading and Activating Process**		**Time to Load and Activate FI (ms)**
			CMX7032 power		***CMX7032 power***	***ATxmega128A1 power***	

			consumption (mW)		*consumption (mW)*	*consumption (mW)*	
9.6	16	8	11.9	Y	190.4	128.0	592
9.6	16	4	11.9	Y	190.7	129.0	687
9.6	16	2	11.9	Y	190.7	129.0	848
9.6	16	1	11.9	Y	190.7	127.5	1,231
9.6	16	0.5	11.9	N (Activation Failed)	191.4	146.5	-

E. Observation of C-BUS - SPI Communication Performance using CMX7032 Main Clock Frequency 9.6 MHz and ATxmega128A1 Main Clock Frequency 12 MHz

Table VI shows fourth observation result that uses CMX7032 main clock frequency 9.6 MHz and ATxmega128A1 main clock frequency 12 MHz, while C-BUS – SPI communication clock frequency varies from 375 kHz to 6 MHz.

TABLE VI. OBSERVATION RESULT OF C-BUS – SPI COMMUNICATION PERFORMANCE USING CMX7032 MAIN CLOCK FREQUENCY 9.6 MHZ AND ATXMEGA128A1 MAIN CLOCK FREQUENCY 12 MHZ

CMX7032 Main Clock Freq (MHz)	**ATxmega128A1 Main Clock Freq (MHz)**	**C-BUS – SPI Clock Freq (MHz)**	**Before FI Loading and Activating Process**	**Success FI Loading and Activating Process? (Y/N)**	**After FI Loading and Activating Process**		**Time to Load and Activate FI (ms)**
			CMX7032 power consumption (mW)		***CMX7032 power consumption (mW)***	***ATxmega128A1 power consumption (mW)***	
9.6	12	6	11.9	Y	191.7	113.0	736
9.6	12	3	11.9	Y	190.7	113.0	848
9.6	12	1.5	11.9	Y	190.7	112.0	1,072
9.6	12	0.75	11.9	Y	190.4	112.0	1,584
9.6	12	0.375	11.9	N (Activation Failed)	190.7	125.5	-

III. RESULT AND DISCUSSION

Result and discussion of observasion result of C-BUS – SPI communication performance will be described more detail in 7 points below.

A. Correlation of IC Main Clock Frequency to IC Power Consumption

According to observation results shown above, comparation of each average CMX7032 and ATxmega128A1 power consumption when using different main clock frequency can be pictured as in Figure 9. Figure 9 shows that average CMX7032 power consumption when using main clock frequency 19.2 MHz (215 mW) is higher than when using main clock frequency 9.6 MHz (190.8 mW). Average ATxmega128A1 power consumption when using main clock frequency 16 MHz (128.3 mW) is also higher than when using main clock frequency 12 MHz (112.6 mW). It means that the use of higher main clok frequency demands higher power consumption and it applies to both CMX7032 and ATxmega128A1.

B. Successful FI Loading Indicator

Considering overall results of observation, it can be stated that in all observation cases, FI loading process is always successful and resulting right checksum values. This is also followed by risen CMX7032 power consumption compared to before FI loading process as pictured in Figure 10. When using main clock frequency 19.2 MHz, average CMX7032 power consumption rises from 21.5 mW to 215 mW after successful FI loading process and when using main clock frequency 9.6 MHz, average CMX7032 power consumption rises from 11.9 mW to 190.8 mW after successful FI loading process. It means that rising CMX7032 power consumption from before to after FI loading process can be an indicator of successful FI loading process.

C. Unsuccessful FI Activating Indicator

According to observation results above, in some cases, FI loading process is successful and resulting right checksum values, but FI activating process is unsuccessful. According to FI loading and activating from host µC process flowchart in Figure 7, after sending activation code to register 0xC8, the next step is to wait for ACT flag to go high. In unsuccessful FI activating cases, ACT flag does not go high after activation code is sent, so ATxmega128A1 does infinite looping to check ACT flag status. It causes ATxmega128A1 power consumption to rise more than in successful FI loading and activating proses cases. Figure 11 shows that when using ATxmega128A1 main clock frequency 16 MHz, average ATxmega128A1 power consumption is 128.3 mW in successful FI loading and activating process case, but in unsuccessful FI activating process case, it rises to 146.2 mW. While, when using ATxmega128A1 main clock frequency 12 MHz, average ATxmega128A1 power consumption is 112.6 mW in successful FI loading and activating process case, but in unsuccessful FI activating process case, it rises to 125.7 mW.

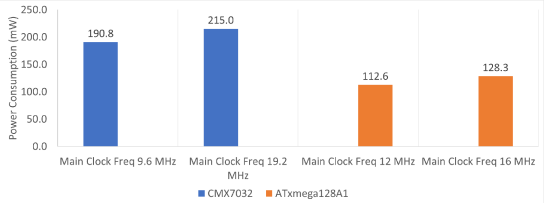

Fig. 9. Comparation of CMX7032 and ATxmega128A1 Power Consumption using Different Main Clock Frequency

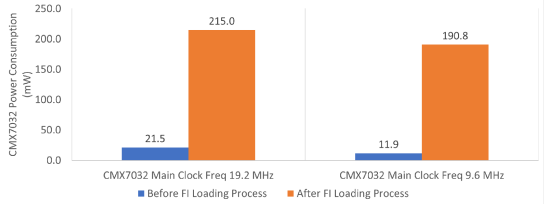

Fig. 10. Comparation of CMX7032 Power Consumption Before and After FI Loading Process

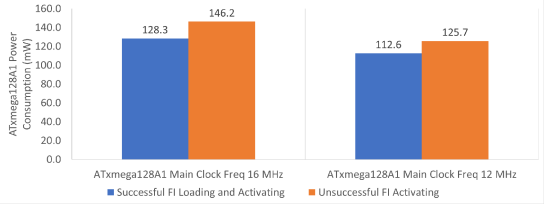

Fig. 11. Comparation of ATxmega128A1 Power Consumption After Successful and Unsuccessful FI Loading and Activating Process

D. *Correlation of CMX7032 Main Clock Frequency and C-BUS – SPI Clock Frequency to the Success of FI Loading and Activating Process*

Observation results above show that when using same CMX7032 main clock frequency, FI loading and activating process can still be successful, although C-BUS – SPI clock frequency used is different. The result also shows that using same C-BUS – SPI clock frequency does not ensure successful FI loading and activating process if CMX7032 main clock frequency used is different. This can be noticed from observation results that show when using C-BUS – SPI clock frequency 1 MHz or 750 kHz and CMX7032 main clock frequency 19.2 MHz, FI activating process is unsuccessful. But, when CMX7032 main clock frequency is changed to 9.6 MHz, FI loading and activating process becomes successful. This also means that using higher CMX7032 main clock frequency demands higher C-BUS – SPI clock frequency to support successful FI loading and activating process.

E. *Correlation of ATxmega128A1 Main Clock Frequency and C-BUS – SPI Clock Frequency to the Time Required for Loading and Activating FI*

According to observation results shown above, when using same ATxmega128A1 main clock frequency, higher C-BUS – SPI clock frequency results less required time for loading and activating FI. But, when using different ATxmega128A1 main clock frequency, higher C-BUS – SPI clock frequency does not always result less required time for FI loading and activating process. This can be caused by lower ATxmega128A1 main clock frequency requires more time for executing all commands for loading and activating FI that affect total required time for FI loading and activating process. Comparation of C-BUS – SPI clock frequency correlation to the time required for loading and activating FI using ATxmega128A1 main clock frequency 16 MHz and 12 MHz is shown in Figure 12.

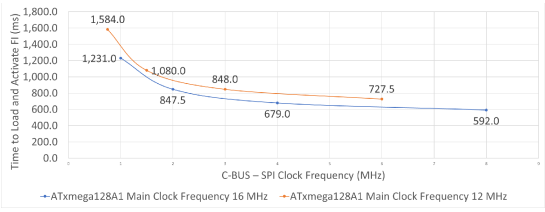

Fig. 12. Comparation of C-BUS – SPI Clock Frequency Correlation to the Time Required for Loading and Activating FI Using ATxmega128A1 Main Clock Frequency 16 MHz and 12 MHz

F. *Correlation of CMX7032 Main Clock Frequency to the Time Required for Loading and Activating FI*

Observation result shows that using same C-BUS – SPI clock frequency, different CMX7032 main clock frequency does not affect significantly to the time required for loading and activating FI. Observation result of CMX7032 main clock frequency correlation to the time required for loading and activating FI shown in Table VII shows that highest time difference is 17 ms.

TABLE VII. OBSERVATION RESULT OF CMX7032 MAIN CLOCK FREQUENCY CORRELATION TO THE TIME REQUIRED FOR LOADING AND ACTIVATING FI

C-BUS – SPI Clock Freq (MHz)	**Time to Load and Activate FI (ms)**		**Time Difference (ms)**
	CMX7032 main clock freq 19.2 MHz	***CMX7032 main clock freq 9.6 MHz***	
8	592	592	0
6	719	736	17
4	671	687	16
3	848	848	0
2	847	848	1
1.5	1,088	1,072	16

G. *Comparation with I2C – SPI Communication*

Other most used serial communication protocol is I2C (Inter-Interface Circuit). Communication between master and slave in I2C protocol begins with START condition followed by slave address to be reached, one read/write bit, a bit of recognition that can be ACK (if the communication was successful) or NACK (if the communication was unsuccessful or the end of the message is set by master), 8 bits of data to send or to receive, the ACK or NACK bit, and finishes with a STOP condition or a condition Repeated START[17]. Compared to I2C that only needs 2 wires (SDA and

SCL), C-BUS needs more wires for CSN, SCLK, CDATA, RDATA, and IRQN. But, C-BUS usage advantages are in its data transfer protocol. C-BUS does not need START/STOP condition, read/write bit, and ACK/NACK bit as in I2C. C-BUS transaction can be started when CSN signal is held low and stopped when CSN signal is high, address byte in C-BUS transaction determines data direction for each transfer, and to check CMX7032 readiness for next register writing process, we just need to wait for ACT flag to go high as decribed in Figure 7. Moreover, when C-BUS interface lines are connected to SPI interface lines, there is no need for additional device. But, when I2C interface lines are about to be connected to SPI interface lines, there has to be additional SPI to I2C bridge device. Figure 13 shows SPI to I2C bridge architecture[18].

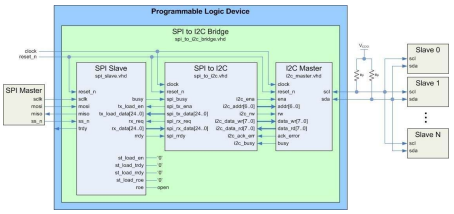

Fig. 13. SPI to I2C Bridge Architecture

IV. Conclusion and Future Works

Literature review and observation of C-BUS communication in CMX7032 with SPI communication in ATxmega128A1 performance are successfully performed. Observation results show that C-BUS – SPI communication performance using various CMX7032 and ATxmega128A1 main clock frequency and C-BUS – SPI clock frequency affect to CMX7032 and ATxmega128A1 power consumption, success of FI loading and activating process, and time for loading and activating FI. Considering observation results above, the use of CMX7032 and ATxmega128A1 main clock frequency and C-BUS – SPI clock frequency can be adjusted to be complied with design needs.

Future works for this project is an AIS transceiver module using CMX7032 and ATxmega128A1 that can be used as an alternative design, but can still meet the requirements. Future observation also needs to be performed to find a solution for reducing noise ripple in C-BUS – SPI communication wires.

Acknowledgment

The authors would like to thank Mr. Mujtahid, as Director of Satellite Technology Center of LAPAN, and Mr. Wahyudi Hasbi, as Head of Dissemination Division of Satellite Technology Center of LAPAN, for their support so that this research can be well completed.

References

[1] M. Fiorini and J. C. Lin, "Clean Mobility and Intelligent Transport Systems", Ch. 12-13, ISBN 978-1-84919-895-0, The Institution of Engineering and Technology, London, January 2015, Book DOI: 10.1049/PBTR001E.

[2] CML Microcircuits. Marine AIS, White Paper, June 2008, Available: https://www.cmlmicro.com/wp-content/uploads/2017/06/WP_AIS.pdf.

[3] H. Saputra, A. Budi K. A., D. Istardi, and S. Wiratno S., "Penggunaan Data Automatic Identification System (AIS) untuk Mengetahui Pergerakan Kapal (Studi Kasus pada Lalu Lintas Kapal di Selat Singapura dan Perairan Batam)", *Jurnal Integrasi*, *October 2016*, *p-ISSN: 2085-3858*, *pp. 139-143*.

[4] CML Microcircuits. DE70322T Marine AIS Class B CSTDMA/SOTDMA Demonstrator/Reference Design, Information Pack, June 2018, Available: https://www.cmlmicro.com/wp-content/uploads/2018/06/DE70322TInformationPack-1.pdf.

[5] J. Koto, M. Rashidi, and A. Maimun, "Tracking of Ships Navigation in the Strait of Malacca using Automatic Identification System", *Developments in Maritime Transportation and Exploitation of Sea Resources, London, 2014, ISBN: 978-1-138-00124-4, pp. 721-725*.

[6] M. Agus, S. W. Widyanto, Ma'muri, S. Wisnugroho, and S. Asuhadi, "Automatic Identification System (AIS) Berbasis Mikrokontroler untuk Pengawasan Nelayan di Wakatobi", *Seminar Nasional Sains dan Teknologi, Jakarta, 2018, p-ISSN: 2407-1846, e-ISSN: 2460-8416, pp. 1-7*.

[7] CML Microcircuits. CMX7032 CMX7042 AIS Data Processor, Datasheet, March 2018, Available: https://www.cmlmicro.com/wp-content/uploads/2018/03/CMX7032_42_FI1_ds.pdf.

[8] CML Microcircuits. Writing a Function Image™ to Serial Memory, Application Note, January 2007, Available: https://www.cmlmicro.com/wp-content/uploads/2017/09/CMX703xCMX704x-Writing-a-Function-Image-File-to-Serial-Memory.pdf.

[9] CML Microcircuits. DE70322T Demonstration Kit, User Manual, June 2018, Available: https://www.cmlmicro.com/wp-content/uploads/2018/06/DE70322T_UM.pdf.

[10] CML Microcircuits. C-BUS Microcontroller Interface, Application Note, December 2002, Available: http://www.g1sle.com/files/downloads/cbus_2.pdf.

[11] CML Microcircuits. C Code For C-BUS, Application Note, March 2011, Available: www.cmlmicro.com.

[12] Atmel. 8-bit Atmel XMEGA A Microcontroller, Manual, November 2012, Available: http://ww1.microchip.com/downloads/en/DeviceDoc/doc8077.pdf.

[13] Atmel. 8/16-bit XMEGA A1 Microcontroller ATxmega128A1 / ATxmega64A1, Datasheet, June 2013, Available: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8067-8-and-16-bit-AVR-Microcontrollers-ATxmega64A1-ATxmega128A1_Datasheet.pdf.

[14] CML Microcircuits. C-BUS Coding Example using the CMX868A, Application Note, April 2007, Available: https://www.cmlmicro.com/wp-content/uploads/2017/06/AN_Gen_CBUS_868ACode_2.pdf.

[15] CML Microcircuits. C-BUS to SPI Interfacing, Application Note, December 2007, Available: www.cmlmicro.com.

[16] CML Microcircuits. DE70322T Marine AIS Class B Demonstration Kit AIS Class B CSTDMA/SOTDMA, Product Preview, June 2018, Available: https://www.cmlmicro.com/wp-content/uploads/2018/06/DE70322TProductPreviewMay2018_5.pdf.

[17] T. L. Río, G. Juarez-Gracia, and L. N. Oliva-Moreno, "Implementation of the Communication Protocols SPI and I2C Using a FPGA by the HDL-Verilog language", *Research in Computing Science 75*, *2014*, *pp. 31–41*

[18] S. Larson, SPI to I2C Bridge (VHDL), May 2019, Available: https://www.digikey.com/eewiki/pages/viewpage.action?pageId=11042934