An efficient LIN MCU design for In-Vehicle Networks

Kyu-Bong Yeon and Jong-Wha Chong

Abstract—This paper describes a design of LIN MCU using efficient memory accessing architecture which provides concurrent data and address fetch for faster communication. By using slew rate control it can reduce EMI emission while satisfying required communication specifications. To verify the efficiency of the LIN MCU, we developed a SoC and tested for several data packets. Measurements show that this LIN MCU improves network efficiency up to 17.19 % and response time up to 31.26 % for nominal cases. EMI radiation also can be reduced up to 10 dB.

Index Terms—Local interconnection network, microcontroller, system on chip, communication controller, automotive buses

I. INTRODUCTION

The number of electronic control units (ECUs) in a car is growing continuously; today's premium automobiles have fifty or more ECUs performing vehicle control functions such as steering, braking, stability control, active suspension, etc. [1]. Automotive ECUs usually consist of microcontrollers units (MCUs) in conjunction with network communication controllers.

In recent automotive systems, several communication protocols, such as Local Interconnect Networks (LIN), Time-Triggered CAN (TTCAN) and FlexRay for X-by-Wire applications are used [2, 3].

LIN buses provide time-critical connectivity for smart sensor and actuator units, such as smart ultrasonic sensor

for autonomous parking assist and smart actuator for adaptive front lighting system. In these units, the cost effective LIN network enables the high dependability of mechatronics elements.

The maximum data rate of LIN bus (single-wire) is limited to 20 kbit/s to keep Electromagnetic Interference (EMI) emission within the limit. For line lengths up to 40 meters the maximum recommended node count is 16. This takes into account the node and line capacitances as well as the maximum allowable response time of the LIN node described in the LIN specification [4].

In conventional LIN systems, the response time can be greatly increased according to the increase of sensor nodes and data. In order to reduce the response time, increasing the clock frequency may increase electromagnetic emissions [5].

This paper describes an efficient LIN MCU design for reduced response time in the LIN system. The proposed LIN MCU shares a compatible instruction set with a conventional MCU. Reduced clock cycles per instruction, Harvard architecture and concurrent-fetching method are adopted in the system's architecture to improve the LIN MCU's network efficiency. In the Central Processor Unit (CPU) architecture, memory accessing architecture and LIN communication control units are adopted to obtain high performance. As an IP core, the LIN MCU core can be easily integrated into the System on a Chip (SoC).

Following subsections briefly introduce conventional LIN communication architecture and LIN message frame structure.

1. LIN Communication Architecture

Communication in a LIN system is based on master-slave architecture. LIN system consists of a master node and one or more slave nodes. The LIN master has

‘Master Task’ to coordinate node communications. Coordination is achieved by periodic execution of LIN schedule that is organized into frame slots. The LIN master controls sending and receiving of all LIN frames using ‘Master Task’. A frame slot with ‘Inter-Frame Space’ must be long enough to transmit the associated LIN frame. The length of the ‘Inter-Frame Space’ depends on the execution cycle (time base) of LIN MCUs with ‘Master Task’. The LIN communication cycle architecture with Inter-Frame Space is shown in Fig. 1.

2. LIN Message Frame

The LIN message frame is composed of several fields with ‘Inter-Byte Space’ and ‘Response Space’. Each field is formed by Serial Communication Interface (SCI) frame. The SCI frame transmits each byte with the Least Significant Bit (LSB) first, and each byte is framed by a start bit and a stop bit. A typical LIN message frame is shown in Fig. 2.

The time delay of a LIN message frame is composed of ‘Response Space’ time and ‘Inter-Byte Space’ time. Each ‘Inter-Byte Space’ time depends on execution cycle time of SCI frame. SCI frame is handled by the microcontroller and LIN communication controller.

All fields of LIN message frame can be transmitted without spacing, giving the nominal length of a message

as follows:

$$T_{Message_Frame_Maximum} = 140\% \cdot T_{Message_Frame_Nominal} \quad (1)$$

Due to cost sensitivity issues, the LIN slaves may use on-chip resonators with a frequency tolerance of up to 14%. The Sync Break field is made up of at least 13 consecutive dominant bits, and it extracts a SCI error from all the LIN slaves. The LIN master transmits the communication clock pulse with the subsequent Sync Byte.

LIN slaves can delay transmission of the frame response with ‘Response Space’ and insert ‘Inter-Byte Space’ between transmissions of SCI frames. Overall, the maximum message frame may lengthen the frame response by 40%.

The response time of a slave is specified within a time window of 140% of the nominal length of the response frame [4, 6]. The maximum length is specified in the LIN specification as follows [6]:

$$T_{Message_Frame_Maximum} = 140\% \cdot T_{Message_Frame_Nominal} \quad (2)$$

Section II describes proposed LIN MCU architecture design with key contributions for efficiency and robustness. Section III shows some measurement results for each contribution. Section IV describes concluding remarks.

II. EFFICIENT LIN MCU DESIGN

As vehicle control systems become more advanced, an MCU is expected to handle more complex tasks. The efficiency (time base) of a LIN MCU gathers more attention because it occupies a significant portion of a system, as well as it becomes time-critical applications. Because of its cost and performance, the SoC approach is very attractive for LIN system designs. Additionally, the dependability of system can be increased.

The LIN communication controller unit is the most important part of the proposed LIN MCU. In our proposed LIN MCU, the LIN communication controller unit is incorporated in SoC architecture. Fig. 3 shows the block diagram of the LIN MCU SoC.

Fig. 1. The LIN communication with Inter-Frame Space.

Fig. 2. LIN message frame.

Fig. 3. Block diagram of the proposed LIN MCU SoC.

In this section, the high performance system architecture of a LIN MCU is discussed in two aspects: the memory accessing architecture with reduced clock cycles and EMI reduction.

The proposed LIN MCU uses less memory accessing as described in section II-1. The LIN MCU uses slow rate control methods to reduce EMI as described in section II-2.

1. Memory Accessing Architecture

The proposed LIN MCU uses Harvard Architecture, in which program and data are accessed from separate buses. This architecture increases bandwidth over the traditional von Neumann architecture, where program and data are fetched from the same memory and bus. In the proposed LIN MCU there are separate buses: program bus for ROM and data bus for RAM to fetch program and data at the same time.

The LIN MCU architecture with separate memories and buses for program and data uses pre-fetching instruction method [8]. Instructions are executed in two-stage pipelining. While an instruction is executed, the next instruction is pre-fetched in the instruction register of the LIN MCU.

The proposed memory accessing architecture for LIN communication can be performed in less time than that required to access the same memory array twice in conventional LIN systems. The LIN MCU uses both a Special Function Register (SFR) and a duplicate register to perform instructions requiring indirect memory accessing for LIN communications.

The on-chip memory map of the LIN MCU is shown in Fig. 4. There are three distinct memory areas: internal RAM with register bank, SFR and auxiliary RAM.

The internal register is important in the LIN MCU

Fig. 4. The on-chip memory of the LIN MCU.

because standard cell registers are used in the present design to decrease the number of clock cycles. LIN MCU is required to perform a LIN communication command via indirect addressing techniques. Registers are located in the 128 bytes of on-chip RAM that are divided into three sub-areas. A separate class of instructions is used to access the registers and the data memory for LIN communications. The upper 128 bytes of internal RAM are overlapped with 128 bytes of SFR in the memory map. The upper 128 bytes of internal RAM are accessed by indirect addressing and the SFR area is accessed by direct addressing. The lower 128 bytes of internal RAM can be accessed by direct or indirect addressing.

In the lower 128 bytes of internal RAM, there are 4 banks of 8 working registers in each bank. The working registers are general purpose RAM locations that can be addressed within the selected bank by any instructions that use R0-R7. The register bank selection is controlled via a program status register in the CPU area.

In the LIN MCU, the working registers are duplicated as latches outside the RAM. The latches can be any standardized cell that would represent RAM such as an 8 memory cells. The duplicate register locations can be called cache registers.

Fig. 5 shows a hardware design for decreasing the number of RAM reads from two to one read during an indirect addressing instruction.

In response to an instruction requiring indirect instruction for LIN communication, the CPU core would receive the bank selection signal along with the address latched from the data bus. The CPU core decodes the instruction as an indirect instruction. The duplicate memories provide the address via muxes and address

Fig. 5. Circuit diagram of a memory access portion of the LIN MCU.

mux to be read from the RAM and the proper data will then be output via the data output lines.

When one of the designated addresses is transferred by the LIN communication controller and a 'write' is being performed, the data being written from the data bus into the RAM is also written into the appropriate duplicate register. Thus, there are two locations that store the same data. Conventional MCUs require two reads, one for the register and the other for the RAM address [7]. In the proposed LIN MCU only one read for the RAM is sufficient to perform an indirect instruction because the duplicate register location contains the same information in the RAM.

This indirect instruction can be performed in less clock cycles because the same memory array is not being addressed twice. Therefore, this memory accessing architecture provides concurrent fetch of address (opcode address) and data (operand).

The proposed LIN MCU reduces the number of dummy fetch and wasted cycles, thereby improving efficiency by at least 2 times the rate of conventional MCU. Three instructions are executed in a 12 clock cycles while a conventional MCU takes 48 clock cycles with including 12 clocks of wait state cycle after each instruction [8].

Fig. 6 shows reduced clock cycles for each instruction. The proposed LIN MCU architecture uses 4 clock cycles with one fetch per instruction while conventional MCUs use 12 clock cycles with two fetches.

2. EMI Reduction

Since most of the automotive applications are affected

Fig. 6. Timing diagrams for three instructions.

Table 1. The proper mode of LIN bus rate

	Slew time (rate)	Duty Cycle
Conventional	Not limited	58%
Proposed	$\pm 2.5 \mu\text{s}$ ($5 \mu\text{s}$)	50%

by EMI [9], stringent Electromagnetic Compatibility (EMC) requirements are used [10].

The EMI behavior of the LIN bus depends on the slew rate and other factors such as current and voltage. The higher slew rate with shorter time and greater difference between high and low voltages, generates more EMI emission. To allow high speed communication of conventional LIN systems, slew rate is not limited which produce wide spectrum on EMI [11].

The maximum slew rate of rising and falling bus signals are limited by the active slew rate control of the proposed LIN MCU. The EMI of LIN communications is very susceptible to slew rate control, which consequently reduce propagation delay.

In the Table 1, the slew rate is controlled for the digital input signal to have a limited value of $5 \mu\text{s}$ and duty cycle of 50% in order to reduce electromagnetic emission. These limited values are obtained by experiments to reduce EMI emission.

In the LIN MCU, the slew rate control represents a non negligible effect on reducing excessive EMI emission from the LIN bus.

III. MEASUREMENT RESULTS

The LIN MCU is designed with HDL and synthesized with Design Compiler. A microphotograph of the LIN MCU is shown in Fig. 7. The LIN MCU chip is manufactured with 0.35 μm CMOS automotive process.

1. Improvement of Network Efficiency

The experiments performed at the detailed network measurement level show the results of the LIN bus behavior. This analysis shows that response time delay has an impact on the vehicle network environment. The proposed LIN MCU bus was continuously monitored on the vehicle bus analyzer; the output signal of the LIN MCU is shown in Fig. 8. Measurements also show the proposed LIN MCU test for response time of LIN buses.

Table 2 summarizes the measurement results of response time. Measurements show that the LIN MCU improves average response time 31.26% compared with conventional MCU.

Fig. 9 shows the response time for the worst case and nominal case of the proposed and conventional LIN system.

Fig. 7. Microphotograph of the LIN MCU. The size of the die is approximately $3 \times 3 \text{ mm}^2$.

Fig. 8. Measurement of the actual response space of LIN buses.

Table 2. Measurements of response time

N_{DATA}	Conventional response time (ms)		Proposed response time (ms)		Improvement (%)	
	Worst case	Nominal case	Worst case	Nominal case	Worst case	Nominal case
T_0	3.33	2.50	2.29	1.72	31.23	31.20
T_1	4.17	2.92	2.86	2.01	31.41	31.16
T_2	5.00	3.33	3.44	2.29	31.20	31.23
T_3	5.42	4.17	3.72	2.86	31.37	31.41
T_4	6.25	4.58	4.30	3.15	31.20	31.22
T_5	7.08	5.00	4.87	3.44	31.21	31.20
T_6	7.92	5.42	5.44	3.72	31.31	33.37
T_7	8.33	6.25	5.73	4.30	31.21	31.20
T_8	9.17	6.67	6.30	4.58	31.30	31.33
Avg.	6.30	4.54	4.33	3.12	31.27	31.26

Fig. 9. Measurement of response time.

The measurements of real time properties performed for each case is limited to the specific transfer of the LIN message on the bus. Measurements results show that the proposed LIN MCU at T_8 gains improvement of 2.87 ms in the worst case response time compared with a conventional MCU.

The efficiency of a LIN varies by the delay of the slave responses. The efficiency of the proposed LIN MCU is calculated assuming unconditional frames.

The figures for efficiency are derived as follows: The specified 16 messages with 2 bytes each require $T_{\text{net}} = 16 \cdot 2 \cdot 8 T_{\text{bit}} = 256 T_{\text{bit}}$ to be sent over the network.

In case of the LIN MCU, we can send 16 frame headers and 16 unconditional messages with data bytes and a check byte. The LIN efficiency determines the

response time of a frame by a nominal value and a maximum value.

Using LIN messages will change the number of required messages and increase the number of data bytes per message by 1 because the first data byte of a frame has to carry the protected identifier of the corresponding unconditional message. In the best case, when each set of nodes produces one update message per request, the number of messages drops down to minimum, giving T_{net} as a minimum time to communicate the information of the messages, resulting in an increased efficiency rate.

Fig. 10 shows network efficiency for messages from one to sixteen in the nominal case. Measurements show that the LIN system improves network efficiency up to 17.19% in nominal case compared with conventional system.

Fig. 11 shows network efficiency on the number of messages from one to sixteen in the worst case. There is worst case (stress case) in LIN conformance test [12] for in-vehicle network, such as stress condition for physical layer and data link layer [13]. Measurements show that the LIN system improves network efficiency up to 12.16% in the worst case compared with conventional systems.

Fig. 12 shows network efficiency of the proposed LIN MCU system for 16 messages. Measurements results show that the proposed LIN MCU improves average network efficiency 14.67% for the worst and nominal cases compared with conventional MCUs.

Fig. 10. Network efficiency on the number of messages for nominal cases.

Fig. 11. Efficiency on the number of messages for worst cases.

Fig. 12. Network efficiency for 16 messages.

2. EMI Reduction

Fig. 13 shows radiated EMI in a conventional LIN system and the proposed LIN system with both peak and average radiated emissions. The spectrum analyzer is used for automotive electromagnetic assessments. EMI in the range of 50 ~ 60 MHz decreases 10 dB compared to conventional LIN Systems. Both systems comply with the vehicle EMI regulations [13]. However, the proposed LIN MCU system may provide functional safety in vehicles.

Fig. 13. Comparison of measurements results of electromagnetic emissions by the proposed LIN system.

IV. CONCLUSIONS

This paper presents the design of an efficient LIN MCU which instruction-level is compatible with conventional MCUs. The LIN MCU improves network efficiency and reduces EMI than these of conventional MCUs.

Experiment show that the LIN MCU improves network efficiency up to 17.19% for nominal case and response time up to 31.26%. EMI radiation also can be reduced up to 10 dB.

The proposed LIN MCU can contribute functional safety issues in future complex vehicles.

ACKNOWLEDGMENTS

The authors wish to thank KATECH (Korea Automotive Technology Institute) for their cooperation, and the MKE for partial funding.

REFERENCES

- [1] N. Navet, Y. Q. Song, F. Simonot-Lion, C. Wilwert. "Trend in automotive communication systems," *Proc. IEEE Spec. Issue Ind. Commun. Syst.*, 93, 1024–1223. 2005.
- [2] A. Shrinath, A. Emadi. "Electronic control units for automotive electrical power systems: communication and networks," *Proc. IMechE D: J. Automob. Eng.* 218, 1217–1230. 2004.
- [3] U. Abelein, H. Lochner, D. Hahn, S. Straube. "Complexity, quality and robustness - the challenges of tomorrow's automotive electronics," *Design, Automation & Test in Europe Conference (DATE)*. 870 – 871, 2012.
- [4] LIN Consortium, LIN specification package revision 2.2, <http://www.lin-subbus.org>, available online. 2010.
- [5] J.-M. Redoute, M. Steyaert. "An EMI resisting LIN Driver in 0.35-micron high-voltage CMOS," *IEEE Journal of Solid-State Circuits*, vol.42, no.7, 1574–1582. 2007.
- [6] W. Elmenreich, H. Piontek, J. Kaiser. "Interface Design for Real-Time Smart Transducer Networks – Examining COSMIC, LIN, and TTP/A as Case Study," *15th International Conference on Real-Time and Network Systems*, 2007.
- [7] K. S. Sanal, K. Rani. "Hierarchical Automobile Communication Network using LIN Subnet with VI based Instrument Clusters and OBD-II Regulations," *International Journal of Scientific & Technology Research*, Volume 1, Issue 3, 2012.
- [8] J. Sanchez, Canton, P. Maria. "The 8-bit microprocessors that preceded the 80x86 family did not include multiplication," *Software Solutions for Engineers And Scientists*, Taylor & Francis, p.65. 2008.
- [9] H. Casier. "Electronic circuits in an automotive environment," *International Solid-State Circuits Conference*, tutorial, February 2004.
- [10] K. Appeltans. "LIN transceivers: a case study of automotive design," *International Solid-State Circuits Conference, ATAC design forum*, Feb. 2005.
- [11] J.-M. Redoute, M. Steyaert. "EMI Resisting Smart-power Integrated LIN Driver with Reduced Slope Pumping," *IEEE Custom Intergrated Circuits Conference*, 2008.
- [12] W. Lawrenz and T. Roskam. "LIN Protocol Conformance Test," *SAE technical paper*, 2005-01-1535, 2005.
- [13] V. Kerzérho, H.G. Kerkhoff, G-J Bollen, Y. Xing. "The Search for Resilience Weak Spots in

Automotive Mixed-Signal Circuits,” *IEEE 17th International Mixed-Signals, Sensors and Systems Test Workshop*, 2011.

- [14] International standard CISPR 25, “Vehicles, boats and internal combustion engines - Radio disturbance characteristics - Limits and methods of measurement for the protection of on-board receivers,” International Electrotechnical Commission, 2008.

Kyu-Bong Yeon received the M.S. degree in Electronics Engineering from Hanyang University, Seoul, Korea, in 2000. Currently he is a Ph.D. candidate at the Hanyang University, Seoul, Korea. He is currently a senior researcher in the

Korea Automotive Technology Institute (KATECH). His interests include automotive semiconductor for high dependability systems, communication systems design, and the design of vehicle networks. His current research interests are automotive semiconductor design for high dependability systems and communication system design for vehicle networks.

Jong-Wha Chong received the B.S. and the M.S. degree in Electronics Engineering from Hanyang University, Seoul, Korea, in 1975, and 1979 respectively and the Ph.D. degree in Electronics & Communication Engineering from Waseda University,

Japan, in 1981. Since 1981, he has been a professor of the Department of Electronics Engineering, Hanyang University. From 1979 to 1980, he was a researcher in the C&C Research Center of Nippon Electronic Company (NEC). From 1983 to 1984, he was a visiting researcher in the Korean Institute of Electronics & Technology (KIET). In 1986 and 2008 respectively, he was a visiting professor at the University of California, Berkeley, USA. He was the chairman of CAD & VLSI society in 1993 and President in 2007 respectively at the Institute of the Electronic Engineers of KOREA (IEEK). He was the President of KIEEE (Korean Institute of Electrical and Electronics Engineers) from 2009 to 2010. He is currently the Chairman of Fusion SoC Forum. His current research interests are of SoC design methodology including memory centric design and physical design automation of 3D-IC, indoor wireless communication SoC design for ranging and location, video system and power IT system.