System Considerations for LIN Bus Controlled Local Networks with Mechatronic Slave Nodes

Martin BARTH, ST Microelectronics, Munich, Germany

Summary

The increasing demand for enhanced comfort functionality in addition to reduced development efforts drives new system concepts based on more complex components, which can be used in several final product implementations. By leveraging the progress in semiconductor product technology, integrated system solutions in combination with a standardized interface are growing more and more important in order to satisfy all these requirements in an economical way. Using the example of a door zone approach, this paper describes a concept based on a LIN bus controlled local network architecture. It contains either full integrated System On Chip Mechatronic slaves as well as discrete slave ECU implementations with a reduced number of active devices and a new MCU with LIN specific features.

1. System Description

What’s the best solution for new system architectures in terms of functionality, flexibility, cost effectiveness and reliability? To answer this question e.g. for door zones, in many cases a compromise has to be found between requirements of several groups inside the carmaker. As result, the finally chosen solution often looks very similar in terms of the various carmaker system partitionings. For body applications, the trend is to connect all ECUs using standardized networks, such as the CAN bus. Thus the average number of nodes increases more and more and is now up to 40 in high-end car models. Regarding costs for a typical CAN node, the demand to lower the costs for local networks led to definition of the LIN bus. This architecture led to a further partitioning of the car network system into distributed local sub systems connected to the main communication bus.

The forecasted trend of increasingly decentralized sub systems using Mechatronic actuator implementation is currently stagnating due to the often higher calculated costs of electronic components compared to conventional solutions. This neglects cost reductions that often exist in a system-based calculation.

The system discussed here is intended to give an example of what a cost effective implementation of a LIN bus controlled local network can look like. The Mechatronic solution is used in that place where the mechanical boundary conditions set limits to increases in functionality. Furthermore a compact discrete solution is presented as an approach to build up function blocks, based on still existing devices completed with a special optimized LIN performance.

In figure 1, the network is shown. The master node is implemented in a central body control unit connected to a body CAN network. Each door consists of a door module (DM-Dr, DM-Pa, DM-Rr, DM-Rl) for power window and door latches. In addition to that, both front doors have a further slave node for mirror control (MMR, MML). Within this system approach a central switch panel (SP) mounted at the driver side was also implemented as a further stand-alone slave node. It controls all power windows; manual door locks and also mirror adjustment with retract function.
This leads to following requirements for the LIN network:

- Wake-up from the master via a received valid signal from the remote key; generally received via CAN bus
- Wake-up via a recognized key opening of the front doors (slave reacts directly on the key without any communication to the master)
- Polling of Switch Panel (SP) to react on activated switches for every actuator control (power window, mirror adjust, door lock)
- Polling of all slave nodes to get the status of window lift position and door opening
- Sleep-mode control of all slaves (battery powered operating mode)

To achieve an acceptable reaction time on pressed keys, the slave polling cycle time in master node has to be less than 200ms. It’s evident, that this network has to be driven with the fastest transmission rate, specified for the LIN: 20kbs. Thus to reduce system costs due to required high accuracy oscillator tolerances (ceramic oscillator), both door module and switch panel will benefit from the performance implemented in the new LIN optimized MCU as will be described later in detail.

Because the LIN bus is not fault tolerant compared to the body CAN, a fail-safe concept for shorted bus lines has to be considered. Thus each node should be capable of recognizing a shorted bus line. The required reaction follows a specially defined procedure (e.g. door latches stay in open state, power consumption of each node will be reduced as much as possible). It’s obvious, that a LIN network can’t have the same performance as a CAN bus controlled one. Thus depending on specific application requirements the decision whether the performance provided by the LIN bus is acceptable or not, cannot be made in general. This performance gap between CAN and LIN is still necessary. Otherwise, there would be no reason for a new bus system because the respective cost per node would be in the same range (e.g. double line for fault tolerant performance)

The major advantage of this chosen system architecture is the strongly reduced number of cables required for connecting each door to the master unit (compared with a common approach of driving all actuators from a central ECU). Furthermore, the costs per node are lower than for a comparable CAN bus controlled solution. It’s evident, that the introduction of a Mechatronic mirror is only competitive when high performance mirror functionality is required. However, the additional mirror features like heating and mirror retract are becoming more and more standard.
2. Door Module as LIN Slave

2.1. General description

Figure 2 shows the LIN slave door module using existing ST devices to implement all functional requirements. The considered actuators are typical for mid and high-class door module systems like:

- Lock and dead lock latch
- Power window lift
- Footstep light
- Switch panel illumination

The mounting position at top of the power window motor causes strongly restricted mechanical dimensions. Therefore a fully integrated power bridge VNH2SP30 in the new Multi Power SO30 package combined with L9949 in Power SO20 package provide a very compact solution for such modules.

With L4979 for rear door modules (no contact monitoring) alternative to the low quiescent current L4989 voltage regulator (both pin compatible in a SO8 package) ST provides another compact and flexible solution. It allows a unique PCB for all door module variants.

The L9638 LIN transceiver provides additional fail-safe features e.g. short circuit state. After a MCU identified shorted LIN bus line, the ECU can switch off itself by leaving the transceiver capable to react on a wake-up after eliminated short circuit condition (e.g. a occasionally bus blocking due to a damaged cable isolation).

The requirements on the MCU are:

- **Window Lift**
  - Anti pinch function
  - PWM to control the motor
  - Position monitoring

- **Door Lock**
  - SPI interface to control the lock motors by mean of the L9949

- **Contact Monitoring in battery supplied mode**
  - Released key
  - Door opening contact

- **Switch Panel**
  - Only for front door passenger side and rear doors

ST72F361 provides another compact and powerful solution with additional intrinsic LIN features that will be discussed in the following chapter.
2.2 The Door Module MCU

2.2.1 System Constraints for the MCU

Timing Accuracy
To operate correctly, the door module needs a time reference with a tolerance of less than 3%. This accuracy is required for the complex algorithm of anti-pinch function but also for diagnostic or error signaling within a specified time frame.

Real-Time Constraints
Some real-time constraints are related to safety, like anti-pinch function, others are for convenience, for example key detection until door is unlocked. Both are obvious but the LIN communication constraints must also be taken into account:

- Data reception handling; at 20kbps the CPU reaction time must be less than 1ms, otherwise data get lost and LIN communication will fail.
- Transmission of requested data within a defined time frame, T_FRAME_MAX.
- Time-out detection; T_HEADER_MAX, T_FRAME_MAX, T_TIME_OUT, etc.

Power Consumption
The power consumption is always critical for modules that must provide a function periodically even if the car is switched off. This is the case for door modules that must cyclically monitor, if a key has been introduced in to the lock.

With respect of short latency time it is all the more difficult for the ECU to meet the consumption requirement. The short latency time of less than 200ms, once the key has been turned until the lock is open implies a short polling period. The actuator (a motor) requires about 100ms to open the lock, leaving 100ms for the MCU to exit from low power mode, detect the key and trigger the actuator.

LIN Communication Bandwidth
LIN bus access is based on a master / slave principle requiring periodic slave polling by the master. The polling period must be chosen according to the real-time constraint between an event detected by a slave node and respective transfer to the master.

2.2.2 The MCU solution - ST72F361

The ST72F361 is based on the well-known 8-bit ST7 architecture. This MCU offers all features required for a door module as described in the previous chapter:

- LINSCI with LIN specific features
- Internal clock with +/-15% accuracy
- Low power mode with cyclical wake-up
- 32Kbyte of Flash memory

With the ST72F361 ST aims to provide a standard 8-bit MCU offering all features required to develop an optimized slave nodes taking advantage of all LIN capabilities.

2.2.2.1 LINSCI Features

To meet low cost requirements and guarantee availability of hardware components, the LIN protocol has been defined to be feasible with a standard SCI (Serial Communication Interface or UART). Today’s existing LIN networks are the best evidence that this objective has been met.

But a LIN implementation on a standard SCI cannot benefit from the entire LIN capabilities and thus limits the possible optimizations at system level especially for:

- **LIN slave node**
  - SYNCH BREAK detection requires CPU handling
  - No synchronization capability on SYNC FIELD
  - Each byte received/transmitted requires CPU handling
  - Irrelevant messages must be read to keep the LIN interface synchronized on the byte stream
  - Limited error signaling, in particular no time-out detection

But also to a certain extent for:

- **LIN master node**
  - Each byte received/transmitted requires CPU handling
  - Generation of a 13-bit SYNCH BREAK requires software „tricks“ consuming CPU resource and communication Bandwidth

To optimize a LIN system, in particular the slave nodes, and therefore reduce the costs, following parameters must be considered:
- CPU load required for the LIN communication
- Clock accuracy required by the application
- LIN communication bandwidth
- LIN interface robustness/validation

The LINSCI is embedded in a small 8-bit MCU, therefore its features aim to support slave functions rather than master ones.

**Header Detection**

The SYNC BREAK and SYNC FIELD contain no application information. They are only required in the LIN protocol and therefore this LIN HEADER should be handled autonomously by the LINSCI. Only once an identifier has been correctly received the LINSCI signals the identifier reception to the CPU. The CPU can then read the identifier and handle the data if it is concerned by the message or if not, discard it by putting back the LINSCI back into mute mode. In this mode the LINSCI just detects and signals correct headers as just described above. If an error is detected in the header the error is signaled to the CPU. For errors detected by the LINSCI please refer to the paragraph „Error Detection“ below. In contrast with CAN controllers, the LINSCI does not provide hardware filters to select messages to be handled and to discard others.

In a LIN bus the software filtering is not critical. This is because even at maximum baud rate (20kbps) with shortest message (0 data byte) at 100% busload a header will be received every 3ms. Such an occurrence is not critical at all, even for an 8-bit MCU.

Slaves have few identifiers (less than 10) to handle and the LIN identifier is coded on one byte only. This makes software comparison efficient and flexible.

**Identifier and Irrelevant Byte Filtering**

Identifier filtering gets a real benefit only if the LIN interface is able to automatically discard all bytes received until next identifier reception. This is not possible on a standard SCI as a SYNC BREAK or a data byte reception are signaled by the same flag and the same interrupt.

Solving this restriction, the LINSCI provides a mute mode to mask all bytes received until a header has been received correctly or an error has been detected during the header reception.

**Extended Error Detection**

The LINSCI detects all header errors and signals them to the CPU. Status flags and interrupt generation make the error handling easy for LIN driver.

Following errors will be detected:
- **Synchronization error**
  - SYNC FIELD deviation > 15%
  - SYNC FIELD not equal to 0x55
- **Identifier error**
  - Frame error (stop bit == 0)
  - Parity error
- **Header time-out**
  - LIN header longer than T_HEADER_MAX (57 bit times).

The slave must be able to measure the T_HEADER length in order to make sure it is less than T_HEADER_MAX, but also to calculate the T_FRAME_MAX with enough precision. Thanks to the accurate time-out handling, the system designer can specify the LIN message scheduling with less margin and so increase the LIN communication bandwidth.
**Resynchronisation**
One of the key features of the LIN protocol is the high clock tolerance for LIN slaves with resynchronization capability allowing the use of low cost oscillators. For correct communication the slave SCI must be able to resynchronize and remain stable for the time of a LIN frame with accuracy below 2%.

**Accuracy of a standard SCI compared to LINSCI**
Due to the standard SCI bit time sampling principle the 2% accuracy cannot be reached for relevant LIN baud rates (10 and 20kbps). This is caused by limited resolution of baud rate prescaler. With allowed 15% clock tolerance, this results in an quantization error of 2.33% (assuming $f_{CPU} = 8$MHz and LIN baudrate of 20kbps).

A significantly error reduction requires an improved SCI bit timing prescaler principle. This has been done in the LINSCI by replacing the baudrate prescaler, an 8-bit integer value, by LDIV, a 12-bit unsigned fixed point value with an 8-bit mantissa and 4-bit fraction. Thus, the quantization error drops down to 0.15%.

For total inaccuracy calculation, an additional failure in the SYNCH-FIELD measurement has to be considered, as well as errors caused by process deviation, temperature gradients and deviation in supply voltage. With assumed +/-1.25% tolerance for the voltage regulator, it can be shown, that the total accuracy for the LINSCI is 1.5% and therefore meets the requirement of LIN specification.

**2.2.2.2 Low Power Modes**
The door module can be woken-up by two events:
- **The LIN master**, which has received the RKE signal to open the doors. This event is not critical for the door module as no periodic task is required to detect the LIN wake-up signal. The event can be handled by the LINSCI receive signal, RDI, with the external interrupt capability. Or the LIN transceiver detects the falling edge on the bus and switches on the voltage regulator, thus starting the door module.
- **The key is turned in the door lock**; to detect the contact the MCU must cyclically monitor the contact to detect the key.

The second type of event is more critical as the MCU must be active every 50ms in order to detect the key and drive the motor to unlock the door.

The ST72F361 provides a low power mode called Auto Wake-up from HALT (AWUH). In this mode the MCU and the main oscillator are switched off. Only a low precision RC oscillator (+/-40%) remains running and will generate an interrupt to wake-up the MCU after a programmable delay. The consumption of this RC clock and the associated counter is around 25uA.

The high tolerance on this RC clock can be easily compensated by the application. The output of the RC clock can be connected to a timer Input Capture signal and, while the MCU is still running on the main oscillator, the deviation is measured by this means.

**2.2.2.3 Embedded Data EEPROM**
In the product life of the ECU, once the first LIN communication has taken place, in the car or at the ECU supplier production site, new nominal values for the LINSCI LDIV registers and the AWUH register can be stored in the data EEPROM to compensate the static deviation of the internal +/-15% clock and the low power RC clock introduced by the process. This accelerates the calibration of the RC clock before entering AWUH mode. Further the LIN clock information can be used to calibrate the timer generating application tick and driving application task scheduler.

**2.2.3. Application Clock Tuning on LIN**
15% clock tolerance might be too high for several applications. Thus the LIN clock information can be used to tune the timer providing system clock.
2.2.4. LINSCI Validation

Even if the LIN protocol is much less complex than the CAN protocol, its implementation in software makes conformance testing of a standalone LIN driver useless. LIN protocol implementations must be validated within the application. However implementing the LIN protocol in hardware requires conformance testing of the implementation only once.

ST is a member of the LIN „Conformance Test“ Working Group and the LINSCI has been validated by ST with the same methodology as the ISO 16845 „CAN conformance Testing“ and also recommended by the LIN WG.

Further robustness tests have been performed based on the Vector LINda tool.

2.2.5. Conclusion

ST72F361 provides most advanced SCI on a standard MCU supporting the LIN features to save CPU load and remove a precise clock source like a crystal or a ceramic resonator. Furthermore this MCU offers all features a LIN slave requires to derive full benefit from the LIN. Hardware implementation of LIN protocol simplifies the LIN driver code, increases system reliability and simplifies application validation.

3. Switch panel as LIN Slave

The requirements for switch panel MCU are similar to those described in the chapter above. It has been assumed, that no switch functionality is required in battery powered operation mode. To get a good reaction speed on pressed switches, the contacts have to be polled in a loop that is not synchronized with master polling of the LIN bus. The current state of various contacts will be stored in RAM and transmitted immediately after receiving corresponding commands from the master.

![Diagram](Figure 5 Driving functions for a high-end mirror, showing significant level of harness complexity)
4. The Mechatronic mirror

Mirrors have the highest lead count and hence offer greatest savings potential for a Mechatronic approach. Up to 19 leads are necessary to provide all functions, implemented in the mirror. Because the hollow spindle of the mirror joint has a limited capacity this can easily become a problem. Figure 5 shows a typical case for maximum level of mirror equipment, as well as the effect on central door module, which naturally also suffers from the multiple options.

The way out of this dilemma is a Mechatronic mirror concept, in which all signals and actuator functions are already pre-processed within the mirror. Only one bus connection and the two supply leads remain (see figure 6). Due to limited space in mirror housing only very compactly built electronics are possible, which in the end eliminates a concept with several packages.

For the transformed monolithic solution, a 0.6µ BCD technology was selected, which also allows integration of an “embedded controller”, whose architecture is like that of the ST7 family from STMicroelectronics. The intrinsic controller monitors and controls all sensors and actuators. Operational software can be tailored to specific requirements of car manufacturer. For the development stage an EPROM version is available. Communication with the central control device runs over a LIN-bus interface.

Because of the multiplicity of power factors, the overall power management was a particular challenge in the development. With SMD power package HIQUAD64 no additional cooling is needed, allowing maximum flexibility of mounting.

![Diagram of Mechatronic mirror solution](image)

**Figure 6** Block diagram of the Mechatronic mirror solution, based on the SoC L9913
4.1 L9913 actuators and its special features

4.1.1 Consideration of power dissipation
The mirror driver L9913 includes a total of 9 actuator drivers, voltage regulator and position sensor supply. All drivers generate in active operation a power dissipation, which in sum, taking into account the package, ambient temperature and heat removal possibilities may not exceed a specific limit. This limit was set at 1.5 watts.

It must be considered that not all actuators must work simultaneously. Therefore a scenario has been worked out, showing which actuators are given priority under which circumstances. Whether less important actuators are blocked for a short period by targeted activation of specific functions, is an exclusive decision of the system developer implemented by means of the software. For this purpose the chip informs the µC about current junction temperature. If the thermal reserves are sufficiently high, there are no resulting limitations. Briefly occurring higher power dissipation, will have no effect as the thermal impedance of the HIQUAD64 package is still better than 15°C/W after a minute.

4.1.2 The EC mirror
The minimization of power dissipation without introducing EMC critical switched mode regulator solutions has lead to the choice on a shunt controller, lying in parallel to the EC glass. A 6 bit D/A converter steered by a controller builds reference for the shunt control. Its maximum power dissipation lies below 500 mW. An external resistor determines the peak current that has an influence on dipping time. At maximum transparency, that is zero volt control voltage, no unnecessary current flows into the saturated shunt controller. The intrinsic high side driver switches the current completely off under these operating conditions. Summed-up, the resulting power dissipation is removed from the chip and occurs in the external resistor, which by virtue of is mounting position contributes to the mirror heating.

4.1.3 Heating of the mirror
A significant advantage for L9913 is its ability to control heating power independent of onboard voltage and actuator type. A high side driver registers the load current to this purpose and transmits this information back to the controller via internal AD converter, which at the same time measures the supply voltage. Using a software controlled simple low frequency PWM in sub Hz range, any desired heating time profile is possible.

4.1.4 Control of the mirror motors
The L9913 contains four half bridges for mirror adjustment and folding motors. Each half bridge is separately controllable from the controller allowing simultaneous operation. The central half bridge, and also the half bridge branch for the fold in motor offers 200mOhm per transistor (25°C) and is hence capable of driving all known motors with very low power dissipation. In order to safe the otherwise normally used mechanical end switches, the central half bridge allows bi-directional current measurement for stall operation detection. The current level can be read out from controller over ADC.

4.2 Assembly and connection technique
In the following section, the construction is taken from an actual Mechatronic exterior mirror implementation, running in production within BMW 3 series.

Starting with a modified drive unit, that allows the implementation of an additional electronic. Within the mirror there is a not to be under estimated connection task, due to the complexity of actuators and sensors. The alternative to a space eating plug connector technique is a flexible circuit board, also known as FPC. It consists of a one side electrically conducting copper layer mounted on base foil and covered with an additional foil. Thus the resulting FPC has a total thickness of less then 150µm, providing best mechanical flexibility for connection inside the mirror chassis. The FPC technique is the key reducing the number of contact points to a minimum. All connected actuators using new sealed
plug connections specially developed for FPC with MQS plug contacts. An additional FR4 substrate is employed to fix the FPC to the mirror.

It is well known that fitting individual contacts in a plug housing is a production step having a high failure rate, which is made avoidable by a flexible lead print. The contact order is given by layout, preventing mixed contacts. Furthermore, the probability of a cable break in the leads due to a movable actuator (fold in drive) is reduced to a minimum. Especially at low temperatures the fold in motor is released of strain, since the leads remain flexible.

Together with various connectors the final Mechatronic drive unit evolves by mounting the FPC in the drive unit and encapsulate this for water protection.

Further objective of a Mechatronic drive unit is multiple uses in various carmaker mirrors. Neglecting variation in respective car color (handled by the carmaker itself), the number of different mirror variants due to mechanical features is still very high. For that, it's a big advantage to have only one Mechatronic drive unit, programmable for driver or passenger side via an external EEPROM.

4.3 Conclusion
With the expected increase in mirror standard configuration, the L9913 provides an innovative approach for such a unified module platform. From an economical point of view, a strongly reduced variety of semiconductor SoC devices will be required. This gets the maximum cost impact due to cumulated quantities, as well as for reduced hard- and software development efforts. One main condition that will affect the acceptance of Mechatronic drive unit platforms is the standardized bus interface. With LIN bus, the precondition for a wide distribution of Mechatronic system solutions has been created.

STMicroelectronics
Design and Application GmbH
Werner-von-Siemens-Ring 3-5
85630 Grasbrunn, Germany

Phone: +49 89 46006 2405
Fax: +49 89 46006 72405
Mailto: martin.barth@st.com