A Star Topology to Improve CAN Performance

Gianluca Cena (†) and Adriano Valenzano (‡)
cena@polito.it, valenzano@polito.it
† Dipartimento di Automatica e Informatica, Politecnico di Torino
‡ IRITI – CNR, Politecnico di Torino

***Abstract* - Because of the particular medium access technique they adopt, CAN networks suffer from an intrinsic drawback which severely limits the maximum bus length allowed at high bit rates. In this paper a new proposal is introduced, which is based on a star topology and permits to enlarge the CAN network extension more than ten times. In the proposed scheme, a star coupler takes part actively in the arbitration phase by disabling on-the-fly all the nodes which lose the contention. The new mechanism shows a behaviour which is very similar to CAN, thus a satisfactory degree of compatibility with the original protocol can be preserved.**

I. INTRODUCTION

The main drawback of CAN [1] is that, from the electrical standpoint, the whole communication support has to behave as a single physical point, so that, at any given time, the logical level sensed by the different nodes is the same, wherever they are connected to the bus. The above limitation depends on the fact that, for a correct behaviour of the arbitration procedure, it is necessary that the transmitting nodes be able to stop sending their messages as soon as a collision is detected (e.g. a dominant level is read on the bus but the bit currently written by the node is recessive). All the nodes which lose the contention have to retry the transmission right after the winning node has finished transmitting its message.

The arbitration mechanism can operate correctly only if the propagation delays on the bus are kept sufficiently short. In particular, the turnaround time (that is, twice the end-to-end propagation delay) has to be lower than the *propagation time segment* which, in turn, is strictly shorter than the bit time. This implies that the end-to-end propagation delay must be strictly lower than half the bit time and, often, the maximum bus length is selected so as to be about one quarter of the bit time (since signals propagate at a fixed speed in the communication support, in the following we will measure distances over the network in bit times).

To overcome this limitation and increase the transmission speed and/or the network extension, the two functions mentioned above have to be separated. In other words, distinct mechanisms must be used for exchanging frames and for stopping the transmissions of the nodes which have lost the contention, so that they do not affect the remaining part of the arbitration. In particular, a star-like topology, similar to some extent to fast Ethernet networks [2], can be adopted, where each node is connected to an active concentrator by means of a pair of links. The star coupler is responsible for masking off on-the-fly all the nodes which lose the arbitration, while each node is still responsible for the transmission (and the eventual retransmission) of its messages.

In this way propagation delays can be as long as several bit times, and both the bit rate and the bus length can be increased consequently. As an additional advantage, it should be noted that such a solution fits in well with the adoption of fiber optics, which achieve higher noise-immunity and can be used also in hazardous environments.

In the following a new transmission technique called StarCAN is described, that is derived from the basic CAN protocol but is also based on the concepts introduced above, so that a significantly larger extension of the network can be achieved.

II. STARCAN BASICS

As shown in Fig. 1, the StarCAN architecture consists of a *logical star coupler* (LSC) to which the different nodes are attached. Each node is connected to the LSC by means of two (opposite) unidirectional communication links: a *forward link* (FL), which carries the signal from the node to the LSC, and a *reverse link* (RL) from the LSC back to the node. The basic principle of StarCAN is the following: each node wishing to transmit a message over the network has to send the frame to the LSC on its forward link, by using a frame encoding scheme which is almost the same as in CAN. It should be noted that propagation delays, in this case, can last several bit times, and there is no longer a common level on the network which is sensed by all the devices at the same time.

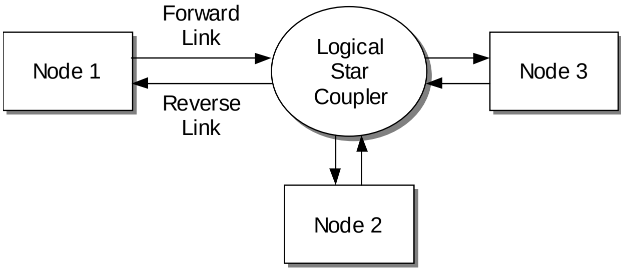

Fig. 1 - StarCAN architecture.

The LSC, on its own, carries out all the operations involved in controlling the access to the network and resolving any (possible) contention. In particular, as soon as a frame reaches the LSC on a given forward link all the other forward links are inhibited immediately, so that they can not interfere with the current transmission.

If more than one node starts transmitting at the same time, a collision takes place, as in CAN. In this case the LSC has to manage the arbitration phase by disabling on-the-fly all the nodes which have lost the contention. Disabled links will be re-enabled after the arbitration phase of the current frame exchange has been completed.

All the (masked) signals travelling on the forward links are combined inside the LSC so as to obtain a common signal (*common logical level*, CLL), which is equivalent, to some extent, to the signal that can be found on a conventional CAN bus. The CLL signal is then returned to the nodes by means of the reverse links. In order to let the different colliding nodes to know the result of the arbitration phase, an access slot is reserved in the returned frame which is set to the dominant level only for the node(s) which won the contention.

A very important feature of StarCAN is that it does not require any buffering of the frames inside the hub (LSC). This provides a network architecture that is able to satisfy the hard real-time communication requirements often found in a number of production environments, and which behaves almost in the same way as conventional CAN, so that most of the existing devices and control applications can be seamlessly converted to the StarCAN protocol.

III. DELAY EQUALISATION TECHNIQUE

If nodes in StarCAN were allowed to start their transmission immediately upon a user request, or right after the end of a frame exchange is detected on the reverse links, the network would be clearly unfair and hence unsuitable for real-time systems. In fact, nodes which are physically close to the LSC would take an advantage over the others because of the lower propagation delay (T_p), while, on the other hand, remote nodes could be ruled out and become unable to send their messages.

The simplest solution to this problem is to introduce suitable compensation periods of time for the forward and reverse links of each node n_i. In the following we call these delays pre-equalisation ($T_{pre/eq,i}$) and post-equalisation ($T_{post/eq,i}$); they are computed so that all the nodes appear to be connected at the same distance L (*virtual distance*) from the centre of the star. For simplicity we will express L in terms of bit times. Since the forward and reverse links for a given node have the same length, the two delays introduced above are equal, so it is sufficient to consider a single parameter for each node we call *equalisation delay* ($T_{eq,i}$).

The delay equalisation technique works in the following way: a node wishing to transmit a frame must wait for a time equal to the equalisation delay before the frame can be effectively sent on FL. As depicted in Figs. 2b and 2c, this can be accomplished by having the MAC mechanism adding a period of inactivity before the SOF bit, during which FL is kept at the recessive level (*pre-equalisation* field). In the same way, each node assumes that the network is in the idle state only after a time has elapsed which is equal to the equalisation delay from the effective end of frame (third bit of intermission read on RL). This somehow corresponds to a *post-equalisation* field appended right after the frame (see Fig. 2a).

Equalisation delays act so as to "lengthen" the links connecting the different nodes to the LSC, so that they appear at the same distance from it and have the same chances in accessing the shared communication support. In practice, this technique permits all the different nodes to synchronise themselves, so that the arbitration information of the frames to be sent arrives to the LSC exactly at the same time and a quasi-conventional CAN arbitration phase can take place.

In general, the virtual distance L is evaluated as the length of the longest link connecting a node to the LSC. However, a better choice is to define such a parameter (which must be the same for all the nodes) directly in the StarCAN protocol specification. In this case, the maximum length of links only depends on the bit rate adopted for the network. A reasonable choice is to select a value of L equal to 2 bit times. Higher values, in fact, enable a larger network extension, but reduce consequently the communication efficiency because of the longer equalisation and acknowledgement fields.

Every node n_i can evaluate the propagation delay $T_{p,i}$ for its own links very easily and with no effects on the system performance, by measuring the round trip delay

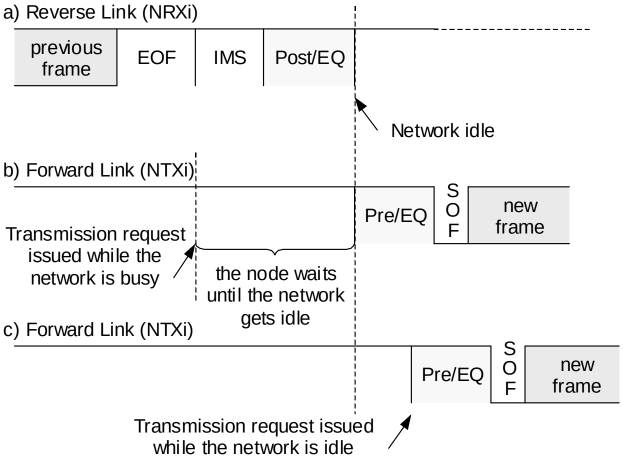
a) Reverse Link (NRXi)
previous frame
EOF
IMS
Post/EQ
Network idle
b) Forward Link (NTXi)
Transmission request issued while the network is busy
the node waits until the network gets idle
Pre/EQ
S O F
new frame
c) Forward Link (NTXi)
Pre/EQ
S O F
new frame
Transmission request issued while the network is idle

Fig. 2 - Frame transmission in StarCAN.

each time it succeeds in transmitting a frame and then by dividing that result by two. The same technique can also be used to update this value periodically, in order to compensate for local clocks tolerances and drifts. The node can then compute its equalisation delay $T_{eq,i}$ as $T_{eq,i} = L - T_{p,i}$.

IV. STARCAN OPERATIONS

A. Frame format

The format of the data (and remote) frames used in StarCAN is slightly different from CAN. As described in the previous section, two *equalisation* fields have been added before the SOF bit and after the intermission, respectively, which in practice delimit the frame. Besides these fields, an *access slot* is added right after the part of the frame involved in the arbitration phase (hence including the identifier and the RTR and IDE bits). This slot is sent at the recessive level by the transmitting node(s) and is overwritten by the LSC to assign the bus mastership to the winning node. This implies that the bit stuffing mechanism must be temporarily disabled on any node when the access bit is read or written. Moreover, a modified ACK field and a synchronisation (SYN) bit are provided at the end of the frame for acknowledging and synchronisation purposes.

B. Operation of the LSC

As depicted in Fig. 3 the LSC is made up of two kinds of blocks: a *stream decoding logic* (SDL) and several *node control logic* (NCL) blocks. Each NCL block is devoted to the communication from and to a specific node and consists of two different parts: an *input control logic* (ICL) and an *output control logic* (OCL) block.

ICL blocks take care of managing the signal received from the forward links on the *star receive* (SRX) ports. When a node is transmitting a recessive bit and the level sensed on CLL is dominant, the related ICL masks off the signal coming from that node, so that the arbitration phase can be correctly carried out. It should be noted that the logic structure of an ICL is extremely simple (i.e., a SR-type flip-flop and a few logic components to detect a collision and to disable the incoming signal), as depicted in Fig. 4. The masked *node logical level* (NLL) signals produced by the different ICLs are then merged together by means of a big AND gate, which carries out the same operation accomplished in CAN by the wired-AND connection scheme. This implies that the resulting signal on CLL is almost the same as the one which can be observed in a conventional CAN network.

The signal on CLL is propagated (with almost no modifications) on the *star transmit* (STX) ports, where it reaches all the different nodes by means of the reverse links. In particular, all the parts of the incoming (winning) frame are repeated unchanged on the outgoing links, except for the access bit, which is set at the dominant level for the winning node(s) and at the recessive level for the nodes which have lost the contention. This operation is carried out by OCL blocks, that overwrite the access slot in the frames which are sent back to the different nodes. Also the structure of OCLs is very simple.

The SDL block implements a subset of the functionality of the receiving part of a CAN controller. In particular, it extracts the timing information from the bit stream read on CLL and generates the control signals for ICLs in Fig. 4, i.e. the *bit sampling strobe* (BSS, a strobing pulse which is activated each bit time), the *access slot select* (ASS, which is set at the high logical level when the access slot is read on CLL) and the *link enable strobe* (LES, a strobing pulse which re-enables all the incoming links). The structure of SDL is very simple and, most important, only one SDL block is needed for each LSC. This makes the implementation of LSC quite inexpensive.

During the arbitration, the LSC has to disable a forward link as soon as it detects that the connected node has lost the contention (or it is not currently transmitting any frame). This can be accomplished very easily in the NCL by comparing the levels on SRX and CLL: when a recessive state is read on SRX while CLL is dominant, the link is disabled by resetting the SR flip-flop. This operation is triggered by the BSS signal, which is generated by the SDL control logic.

Propagation of the incoming signals on the forward links is re-enabled in the LSC (by means of the LES line) $2 \cdot L$ bit times after the access bit is read on CLL. At that time, only the winning node is still transmitting, while all the other stations (including the loosing transmitters) have switched to the receiving state. This ensures that there are

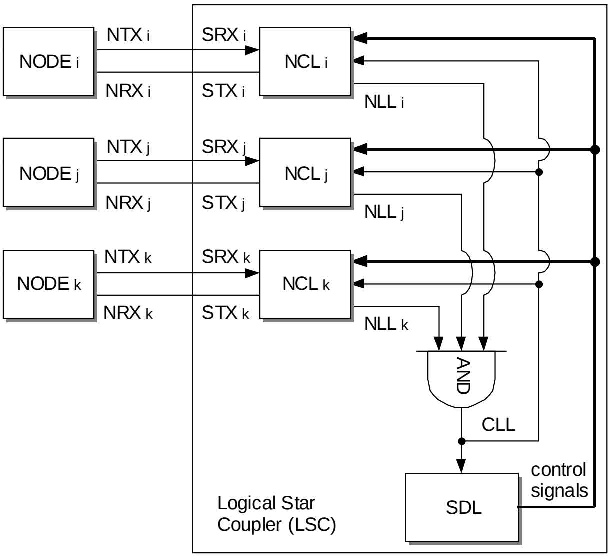

Fig. 3 - Structure of the LSC.

no longer garbage bits which are still travelling on the links and that may affect the transmission of the winning frame.

C. Operation of StarCAN nodes

Each node is provided with two distinct ports, that is a *node transmit* (NTX) and a *node receive* (NRX) port, connected to the forward and reverse links, respectively. The behaviour of StarCAN nodes is similar to conventional CAN nodes, however each node transmits on FL and receives on RL. Nodes can know the result of the arbitration phase carried out by the LSC by examining the bit stream coming on the RL. As in CAN, when the bit of the identifier which is being read differs from the corresponding bit of the transmitted frame, the arbitration is lost. In this case, however, a non-negligible phase lag may exist between the signal written on the output port and the one received on the input port. In addition, the transmitting nodes must also check the access bit: a dominant level notifies the node it is the winner of the current arbitration, while a recessive value means that the arbitration has been lost. This is needed to cope with the situation when two or more nodes start transmitting frames with the same identifier nearly at the same time (this is possible when the data field is empty).

As soon as the node finds out it has lost the contention, it must immediately stop transmitting on FL. In the same way, a node which is going to transmit must be forced to abort the transmission also when it has already started to "send" the pre-equalisation field and a dominant (SOF) bit is read at the same time on RL.

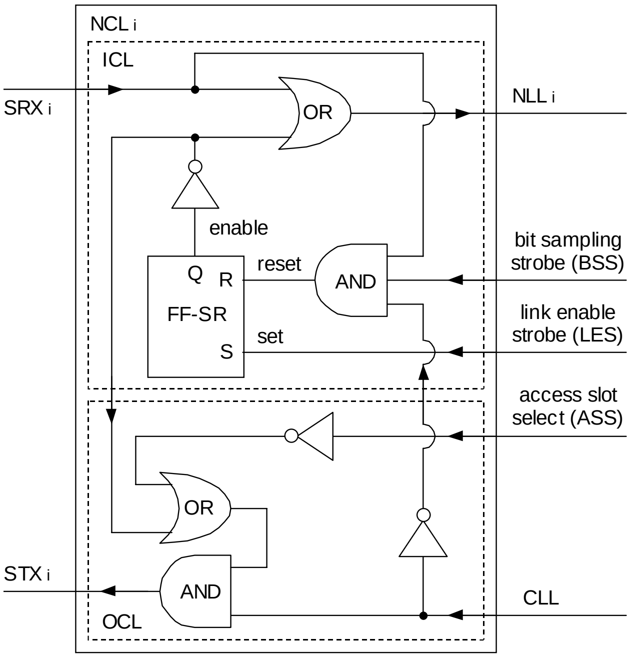

Fig. 4 - Structure of NCL blocks.

V. CONCLUSIONS

CAN suffers from a serious limitation concerning the bus extension at high bit rates [3], which may role out this network in the next future as a high performance solution in a number of application fields. In this paper a new kind of protocol is introduced, which is based on the principles of CAN but which relies on a star topology and enables a noticeably larger network extension. StarCAN breaks the size-speed trade-off of CAN.

If L is set equal to 2 bit times, the maximum extension (diameter) of a StarCAN network is up to 16 times higher than a conventional CAN network operating at the same bit rate. For example, if a 1Mbit/s bit rate is selected, the diameter of a StarCAN network can be as high as 800m (as opposite to a CAN bus which can extend up to 40m). In the same way, it is theoretically possible to conceive a 100m-wide StarCAN network operating at 8Mbit/s.

StarCAN frames are longer than corresponding CAN frames, and hence the efficiency is somehow reduced. However, since only 10 additional bits are needed when L is set to 2, the achievable throughput is worsened on the average by no more than 15% with respect to CAN, if the same bit rate is considered. Even though StarCAN is not directly compatible with current CAN devices, it shows a behaviour which is almost the same as the conventional CAN protocol. This means that implementing StarCAN devices is a relatively simple and inexpensive issue.

REFERENCES

[1] International Standard Organisation, "Road vehicles - Interchange of digital information - Controller area network for high-speed communication", ISO 11898, November 1993.

[2] Institute of Electrical and Electronics Engineers "Local and Metropolitan Area Networks: Media Access Control (MAC) Parameters, Physical Layer, Medium Attachment Units, and Repeater for 100 Mb/s Operation, Type 100BASE-T (Clauses 21-30)", IEEE Std. 802.3u-1995, Supplement to ISO/IEC 8802-3:1993.

[3] G. Cena, C. Demartini and A. Valenzano, "On the Performances of Two Popular Fieldbuses", in *Proc. WFCS'97 IEEE Workshop on Factory Communication Systems,* Barcelona, Spain, October 1997, pp. 177-186.