White Paper

Energy Saving in Automotive E/E Architectures

Thomas Liebetrau, Ursula Kelling, Tobias Otter, Magnus Hell

December 2012

Abstract

Reducing electrical energy consumption in the complex E/E-architectures of modern passenger vehicles has increasingly been a topic for discussion over the last couple of years. Most of the electronic functionality is not in constant use while the car is operational. Therefore it is obvious that either entire ECUs or some functionality within one ECU can be deactivated temporarily.

Partial networking, a more radical approach to power saving, introduces partially deactivated nodes into given E/E networks. This functionality requires new bus transceiver devices decoupling a node from an active bus with selective wake capability. The implementation of those devices in the CAN branches of forthcoming vehicle networks is currently in preparation at some carmakers. Usage in further automotive bus systems (LIN, FlexRay, Ethernet) will follow later.

Nodes which cannot temporarily reply to a request from another node would usually lead to a network error. Communication and error handling of the entire network needs to be adapted to this new situation, and proper mechanisms need to be implemented to deal with deactivated nodes. Therefore some carmakers have looked for smoother methods of managing temporarily deactivated nodes without impacting other ECUs in a given network. AUTOSAR concepts of ECU degradation and pretended networking target this aim. ECU degradation uses simple mechanisms to enable the AUTOSAR stack to save power, while pretended networking assists network operations to continue to run. Both mechanisms, working together, enable applications to save power and keep network operations running while facilitating fast ECU wake-up times.

This paper briefly introduces these different methods and respective implementations in semiconductor devices. At the end of the paper, there is a calculation based on some realistic assumptions, what kind of savings could be achieved assuming these methods will be comprehensively introduced in vehicle architectures.

Contents

1 Motivation

The advanced implementation of electrical systems has led to an increased consumption of electrical energy in automotive wiring harnesses. Up to now, there has been no real visible or felt benefit for the customer in using energy saving solutions in the car. If the combustion engine is running, electrical energy has always been considered as a given resource. Therefore energy consumption while the car is in operation has been not the focus of engineers developing electrical systems; but rather they have concentrated on the lowest possible current consumption while the car is in rest i.e. parked.

This focus is now changing. Energy efficiency is no longer an option, but is a prerequisite when defining and developing new ECUs. From 2020 onwards there will be a very challenging threshold for CO_2 emissions with 95g CO_2/km for passenger car fleets sold in Europe. The car industry is currently defining steps towards this target. The introduction of vehicles with pure electrical drive trains will accelerate the transition to energy efficient solutions. For this type of cars energy efficiency directly influences mileage; it turns into a very real and visible benefit to the customer. In the future, the buyer of an expensive E-vehicle will not be prepared to suffer a lower comfort compared to that available in vehicles of today. Therefore, current consumption during car operation is increasingly moving into the focus of automotive engineers.

More efficient use of the given energy supply makes high demands on energy management in the wiring harness, as well as on the implementation of electrical functionality in automotive networks. But there are significant savings to be made. Current methodologies for ECU current consumption, without consideration of load currents, could make energy savings of up to 50% compared to the status quo today. German carmakers currently dominate these discussions [1], but it is certain that those measures will be implemented on a world-wide basis across the vehicle industry within a certain timeframe.

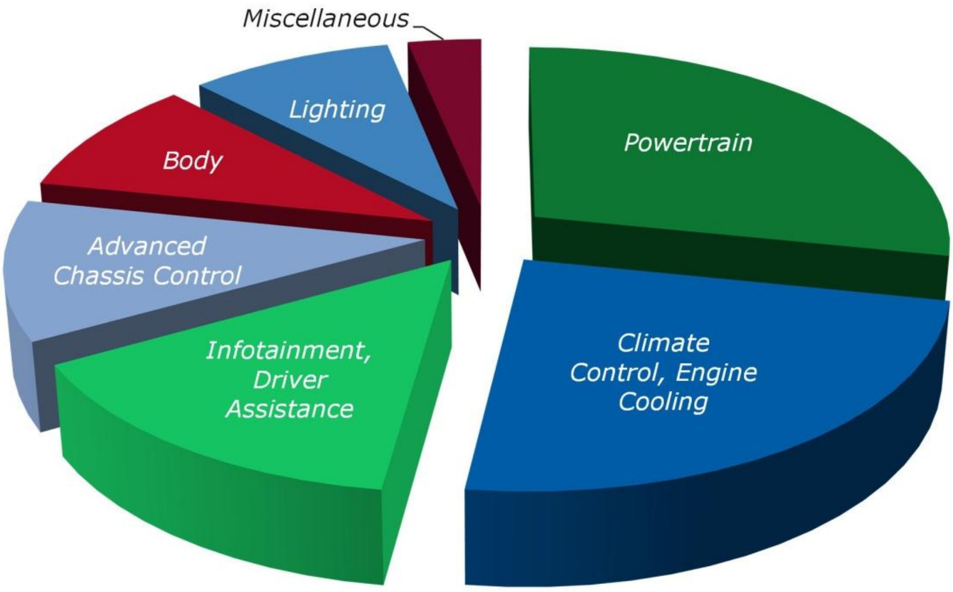

Figure 1: Share of Current Consumption per Function, Source: BMW [2]

2 Bus Systems with Wake Feature – State of Technology

Bus systems in current vehicles do not support partial deactivation of ECUs combined with a selective wake while communication is running. There are indeed several wake mechanisms; but all nodes are woken simultaneously when communication has been activated by a single node. The most popular automotive bus CAN (Controller Area Network) is defined in ISO 11898. ISO 11898-1 describes the data link layer, whereas ISO 11898-5 illustrates the wake capable high speed physical layer. Another popular automotive bus system LIN (Local Interconnect Network) had been defined by a consortium starting in the year 2000. Initiated by some semiconductor vendors, this bus system has been early designed with a wake capability. The specific low cost option of CAN defined by General Motors (GMLAN, Single Wire CAN physical layer [3]) indeed supports selective sleep (one or several sleeping nodes during active bus communication), but only allows global wake based on specific bus voltage level. Separate wake of nodes (Selective Wake) is not support in all these bus systems.

The younger bus system FlexRay also supports global wake, which is used to initialize a FlexRay cluster; but selective wake is not supported. The industry standard bus Ethernet will heavily grow in vehicle electronics within the next years. Ethernet already implements functionality to selectively wake nodes, but the implementation of this Wake-on-LAN feature consumes more energy than it is applicable in automotive environment.

Bus System	Maximum Data Rate	Used Data Rate (Automotive)	Industry Standard OEM Standard	Physical Layer	Wakeup Capability	Remark
CAN	1 MBit/s *(High Speed CAN)*	500 kBit/s	ISO 11898-1 ISO 11898-2/5 *www.iso.org*	Unshielded Twisted Pair (UTP)	Active Bus (Global Wake)	Dominating CAN option
	125 kBit/s *(Fault Tolerant CAN)*	125 kBit/s	ISO 11898-1 ISO 11898-3 *www.iso.org*	UTP	Active Bus (Global Wake)	Currently Replaced by High-Speed CAN
	83.33 kBit/s *(Single Wire CAN)*	33.33 kBit/s	GM LAN (General Motors Local Area Network)	Single Wire	Higher Supply Voltage Level (Global Wake)	Defined by General Motors; No ISO Standard
LIN	20 kBit/s	19.2 kBit/s	LIN Consortium *www.lin-subbus.org*	Single Wire	Active Bus (Global Wake)	Low Cost Bus, Broad Usage
SAE J2602	10.4 kBit/s	10.4 kBit/s	SAE J2602 *www.sae.org*	Single Wire	Active Bus (Global Wake)	Based on LIN, Deviations on Data Link Layer
FlexRay	10 MBit/s	10 MBit/s 5 MBit/s 2.5 MBit/s	FlexRay Consortium *www.flexray.com*	UTP	Active Bus (Global Wake)	Usage in Real Time Systems (e.g. Chassis Control)
MOST	150 MBit/s	25 MBit/s 50 MBit/s 150 MBit/s	MOST Cooperation *www.mostcooperation .com*	Fibre Optical or UTP (MOST-50 only)	-	Focused on Multimedia Applications
Ethernet	Up to 10 GBit/s (40+100 GBit/s in definition)	100 MBit/s 1 GBit/s aim	IEEE 802.3 *www.ieee.org*	STP,UTP,Coax, Fibre Optical	Wake-on-LAN IEEE 802.3az	Industry Standard, first Usage in Vehicles

Figure 2: Overview of Vehicle Bus Systems with Wake Capability

3 Solutions for Selective Wake

Carmakers today demand the capability to selectively wake sleeping nodes on a given bus during active communication in order to enable energy efficient E/E networks. A global wake feature as described in chapter 2 is not sufficient. The majority of network nodes today are CAN nodes; therefore the biggest savings can be expected for CAN networks.

In this section, two basic methods of supporting energy efficiency in automotive networks are described; Partial networking and ECU degradation in conjunction with pretended networking.

The implementation of these mechanisms into a given vehicle environment is under investigation in several carmaker working groups, in close co-operation with semiconductor vendors. Results can be reviewed in some recent publications; see references [2] [4] [5].

3.1 Partial Networking with Selective Wake CAN Physical Layer

Partial networking has been discussed by German car manufacturers since 2008. The basic idea is to independently switch single or multiple nodes within an E/E network into current saving mode called Standby or Sleep mode (Selective Sleep) and be able to wake those nodes up on pre-defined wake messages/CAN frames (Selective Wake).

Mid of 2010 the German car manufacturers founded a working group called SWITCH (Selective Wake-up Interoperable Transceivers for CAN High-Speed Networks) in order to define CAN transceivers with selective wake capability. The intention is that this definition should lead to an extension of the existing CAN standard ISO 11898, and the working group has provided a draft document (ISO 11898-6) to the ISO community. In addition, a supporting conformance test (ISO 11645-2) and a supplemental OEM requirement specification, similar to documents for existing transceiver solutions for CAN, LIN, FlexRay, have been defined. For more details, refer to [6].

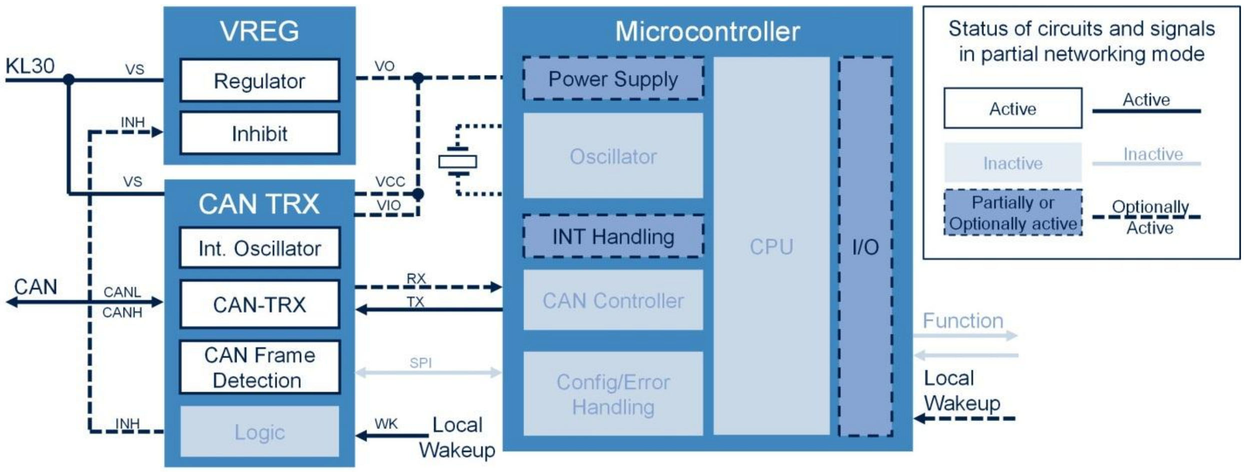

Figure 3: Basic Principle of Partial Networking

During bus sleep the transceiver will wait for a ‘wake-up pattern’ (WUP), which can be part of one or more CAN frames. This mechanism had been already introduced in ISO 11898-5. As soon as bus traffic was detected, the device in selective wake mode autonomously scans CAN frames on the bus for a ‘wake-up frame’ (WUF) in order to switch the ECU back into the normal mode of operation. The microcontroller on the ECU could either be switched-off during selective wake mode (the assumed mode for most applications), or remain partially active.

When implementing one or more nodes which could be deactivated while normal bus communication is running, the possible effects on application software must be considered. If one ECU requires data from another ECU which is in partial networking mode, it is essential to send a wake message (WUF) to that ECU before sending the information request. Therefore advanced network management needs to be implemented in order to manage sleeping ECUs to prevent malfunction within the network.

The AUTOSAR community has been working on a concept dealing with this constraint. Called partial networking it will be the basis of ECU software for CAN nodes in E/E-architectures with selective wake capability. It is therefore limited to vehicles with AUTOSAR software architecture.

Partial networking means partial deactivation of subnets within a given network. There is no intention to individually deactivate a single node, but clusters of ECUs simultaneously, in order to reduce the additional number of possible different vehicle states. The car manufacturers define these ECU clusters based on a common behavior in various driving situations; the partial networking wake scheme enables an individual wake of any cluster or even a combination of two or more clusters. This is seen as the only reasonable way to limit the verification effort of E/E networks during vehicle development (see references [2] [4]).

3.2 Microcontroller Focus of ECU Degradation and Pretended Networking

The selective sleep/wake of ECU nodes is a very important method of saving energy in automotive networks. However it does not cover the control of power consumption based on the dynamic requirements of functionality and performance. Operational ECU features are usually targeted to the maximum required performance. For an engine control unit for example, it covers functionality at the highest possible crankshaft speed; a door control unit with window lift control is optimized for demanding window lift function with anti-pinch. The required performance is reserved permanently but is only rarely required. As already mentioned, energy saving has not been an important constraint during the system definition. Sometimes peripheral functions on ECUs are deactivated while temporarily not in use, but the functionality of the microcontroller remains untouched; the microcontroller is consuming energy while running in wait loops.

Observing solutions outside the automotive industry, the fast changing and innovative consumer electronics field already uses very intelligent systems involving microcontroller/-processor performance adaptation [7]. There are sophisticated techniques in place, e.g. in smartphones, notebooks and tablet computers. In these kind of battery supplied devices, energy saving is of direct benefit for the user since it affects the available operating time when there is no additional power supply. Another example would be the importance of energy saving in commercial IT systems (server/cloud). ‘Green IT’ is more or less required to be fulfilled in today’s market. The technical realization is complex in both cases, but highly efficient, dynamic adaptation of current consumption to the needs of the application can be achieved.

Physical constraints in vehicle electronics are of course the same as in a mobile phone or computing system. However there are also important differences that need to be considered. Vehicle electronics need to fulfill extremely high quality and reliability requirements; any efficiency feature must neither decrease the level of reliability, nor influence any ECU functionality, nor impact the response time to the customer. Vehicle safety requirements are also increasing. Additionally, current E/E networks represent a kind of distributed intelligence; functionality is widely spread into many different ECUs from more than a handful of suppliers. Energy saving techniques must not influence the interaction of this complex hardware/software system structure, and any impact to field failure rates must be prevented. As a result it is therefore highly recommended to limit the number of device states with an easy-to-use structure of transitions between low power and normal modes supported by the ECU operating system.

The current AUTOSAR *ECU degradation* concept is targeted to cover all of these constraints. Degradation of functionality is applied only to the microcontroller. Certain functionality can be switched-off; low power states are possible in a single core as well as in multi-core devices for each CPU separately, including the main CPU. It must be certain that the operating system is running without any impact at all times. Further measures to increase the level of current saving, such as reduced processor clock or reduced supply voltage levels, have not been considered so far. Increasing needs of energy efficiency casts doubt on this limitation, especially when considering vehicles with a pure electric drive train, where any increase of mileage means a substantial benefit for the customer.

Microcontrollers already offer various current saving modes, which are only rarely used in today's ECU designs. Since it is possible to switch off parts of the peripheral units of the microcontroller while keeping the CAN controller alive, results are already substantial [7][8]. The AUTOSAR community is currently working on a concept using the measures defined in *ECU degradation*, but also to enable the network to remain running at the same time. This was proposed in a concept called *pretended networking* in early 2010. As with *partial networking*, these methods can only be used in AUTOSAR software architectures. However there are differences in usage of *pretended networking* compared to *partial networking*.

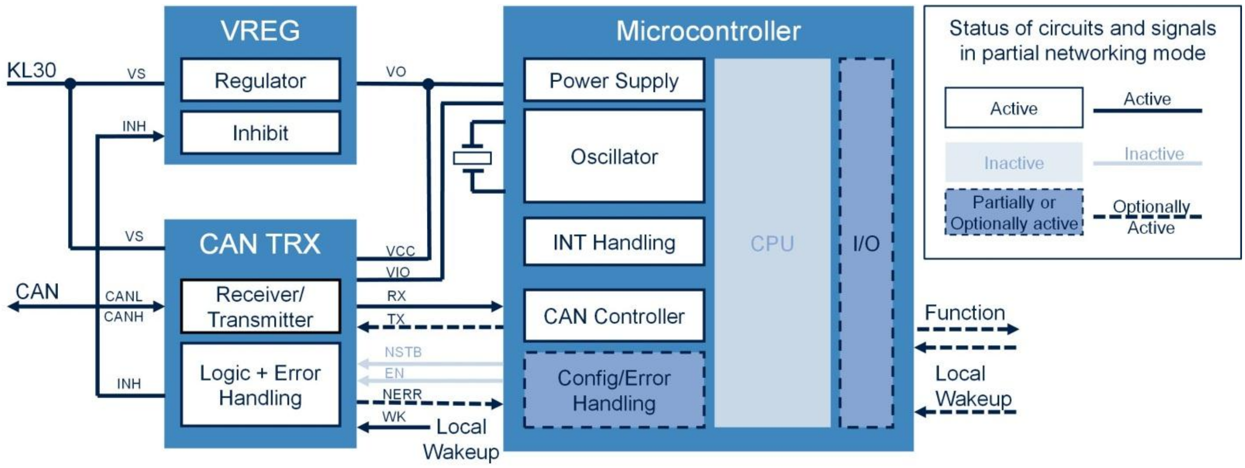

Figure 4: Basic Principle of Pretended Networking

Pretended networking can be used in networks where an ECU can switch into a self-determined low power mode returning to normal operation dependent on received CAN frames or signals. This feature enables transparent behavior in a given network. An ECU can take action at any time and with a very low response time on incoming events and vehicle states. It does not require one special wake-up frame. Additionally, cyclic transmission of predefined CAN messages is possible without using the microcontroller CPU, although this is not an absolute requirement to ensure that previous generation controllers can also run *pretended networking*. This is one basic constraint for the transparent implementation of such an ECU into a given network.

Based on these features, it is possible to integrate *pretended networking* in existing networks, for example in given car platforms without affecting other nodes in the network. This is a very important consideration for the development of new car models based on a given (and not changing) platform configuration. In summary, the basic advantage of *pretended networking* is that it supports the transparent migration of current saving ECUs into a given car platform [5]. However the absolute value of current saving is smaller compared to *partial networking*.

4 Implementation of a Selective Wake CAN Physical Layer

The first implementation of a selective wake CAN transceiver from Infineon is the TLE9267 system basis chip. Figure 5 shows the block diagram of this IC developed for door control units. System basis chips (SBC) have been used for many years particularly in control units of the body electronics. In addition to the typical supply and communication functions (voltage regulator, watchdog, SPI, CAN and LIN transceiver), they often incorporate other, application-specific functions, such as high-side and low-side drivers, for example. To assure the required downward compatibility, Infineon offers an otherwise function and pin-compatible SBC with a CAN cell according to ISO11898-5 (TLE9266). Both ICs provide the basis for further products of a new SBC generation and are currently (as at October 2012) in development. Within this new family, all the derivatives will be pin and software-compatible [9].

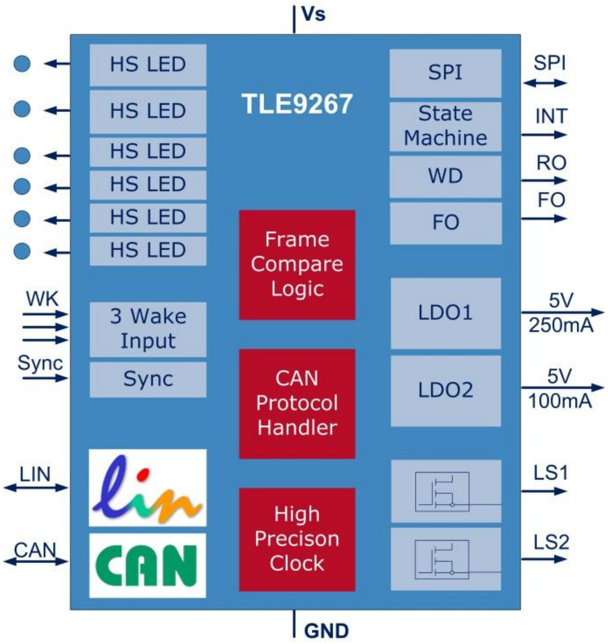

Figure 5: Block Diagram of Selective Wake Capable System Basis Chip TLE9267

4.1 Activation of Selective Wake Mode

For *partial networking*, individual nodes are deactivated or switched into a low power mode. This changeover is realized by means of normal CAN message. This can ensue via a central control unit (power master or gateway) or also locally. Here, as described above, the formation of clusters is favored so as to switch multiple control units simultaneously into the low power mode and also to wake these up again all together (see also [2],[4]).

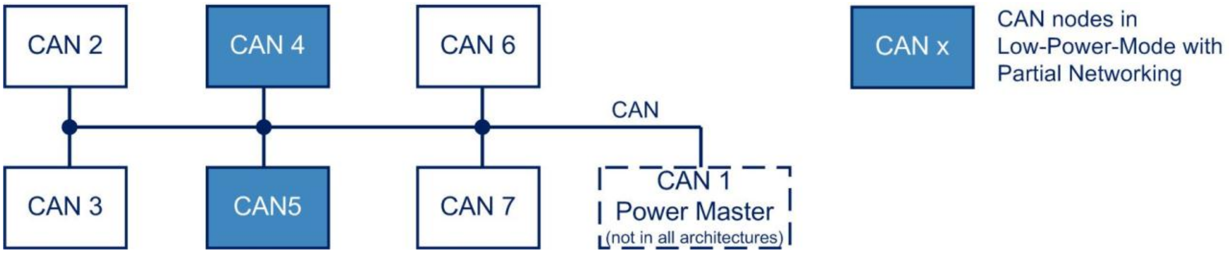

Figure 6: CAN network with Partial Networking and Power Master

When low power mode is requested, the information must be available in the control unit regarding with which CAN message the node or nodes are to be woken up again, and which mask the nodes are to use for evaluating the CAN messages. Depending on the network architecture, this information is transferred once only during the initialization/configuration of the network or, how-ever, each time partial networking mode is activated, e.g. via the power master.

The microcontroller in the corresponding ECUs receives this information and, to do this, uses the protocol handler found in the microcontroller, the ISO data link layers and, if necessary, the corresponding AUTOSAR drivers. The transceiver is used in normal mode for the physical bus connection to the CAN bus. Once the microcontroller has received and processed the information, it configures the transceiver and/or the SBC by transferring the wake-up mask by SPI and, by reading back, ensures that the information has been transferred correctly.

As such, the requirements are satisfied for activating partial networking in the SBC and/or transceiver and for switching into low power mode. Once the microcontroller has delivered the corresponding SPI message, it switches itself off or goes into a low power mode. The changeover is critical, since no CAN wake-up message is allowed to be lost during this transition. Consequently, the start machine is implemented in TLE9267 in such a way that the evaluation of the CAN messages can be carried out in parallel for a time by both the SBC and the microcontroller.

4.2 Overview of Low Power Modes

Particularly when using an SBC, distinctions are made between different power-saving states, which can be selected depending on the function of the control unit as well as however depending on the operating state of the vehicle. In each of the low power modes, the CAN bus allows the user, depending on the configuration, to use a selective wake either for an active bus with a ‘wake up frame’ (WUF) or, however, for an inactive bus, to use the wake-up function according to ISO 11898-5. To improve its disturbance immunity and to minimize incorrect waking on the bus due to spikes, a series of dominant levels have been defined for this in ISO 11898-6, otherwise known as the ‘wake-up pattern’ (WUP).

Sleep Mode

In SBC sleep mode, the voltage regulator is deactivated, the microcontroller is no longer supplied, and waking can now only be triggered via CAN or other waking sources in the SBC, e.g. monitor inputs. If a wake-up is detected, the SBC activates the voltage controller, the microcontroller starts, re-initializes itself and proceeds to process new incoming CAN messages. In sleep mode, it is possible to achieve the lowest power consumption at the expense of a long wake-up time and highly restricted functionality.

Stop Mode

In SBC stop mode, also referred to as standby mode by some manufacturers, the voltage supply of the microcontroller remains active, although the voltage regulator integrated in the SBC may only show a minimal current consumption of <30 µA. While the SBC is in stop mode, a corresponding low power mode can and must be set in the microcontroller so as to permit low current consumption. This makes it possible to connect further wake-up sources to the microcontroller, to use polling in the microcontroller and to realize a fast wake-up time.

4.3 Processing of Wake-Up Frames

Every incoming CAN frame is sampled by the CAN transceiver, is checked for framing errors and compared with the mask. This requires a precise oscillator clock and the receiver section of a CAN protocol handler in addition to a logic that compares the received CAN message with a mask. Figure 7 shows a simplified block diagram of the implementation of a selective wake CAN cell. This block can be used in a standalone CAN transceiver, a system basis chip, and of course also in a larger system-on-chip.

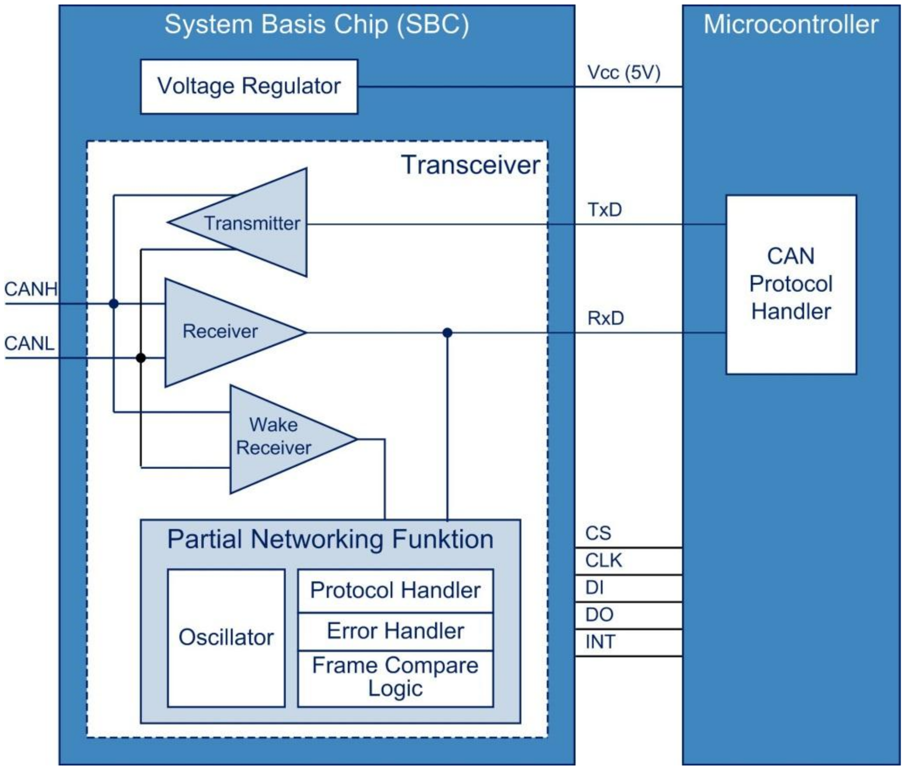

Figure 7: Simplified Block Diagram of Selective Wake CAN Transceiver Cell

Sampling with oscillator

Sampling is performed with a bit-time logic according to ISO 11898-1 that defines at which point the respective CAN bit is sampled and with which maximum jump displacement a phase error can be corrected. For a CAN baud rate of 500 kBit/s, a sampling point of 80 % and a maximum jump displacement of 400 ns are recommended.

The requirements on the oscillator result from the signal propagation delay in the CAN network including all parasitic effects and the oscillator deviation of the transmitter. An active CAN node has a maximum oscillator tolerance of 0.5 %. A passive CAN node in selective wake mode permits a greater tolerance of the order of 1 %, since only receiving take places during bus arbitration and nothing has to be sent. The first 65 % of the bit time is reserved for network propagation delays including arbitration. The timing is schematically represented in.

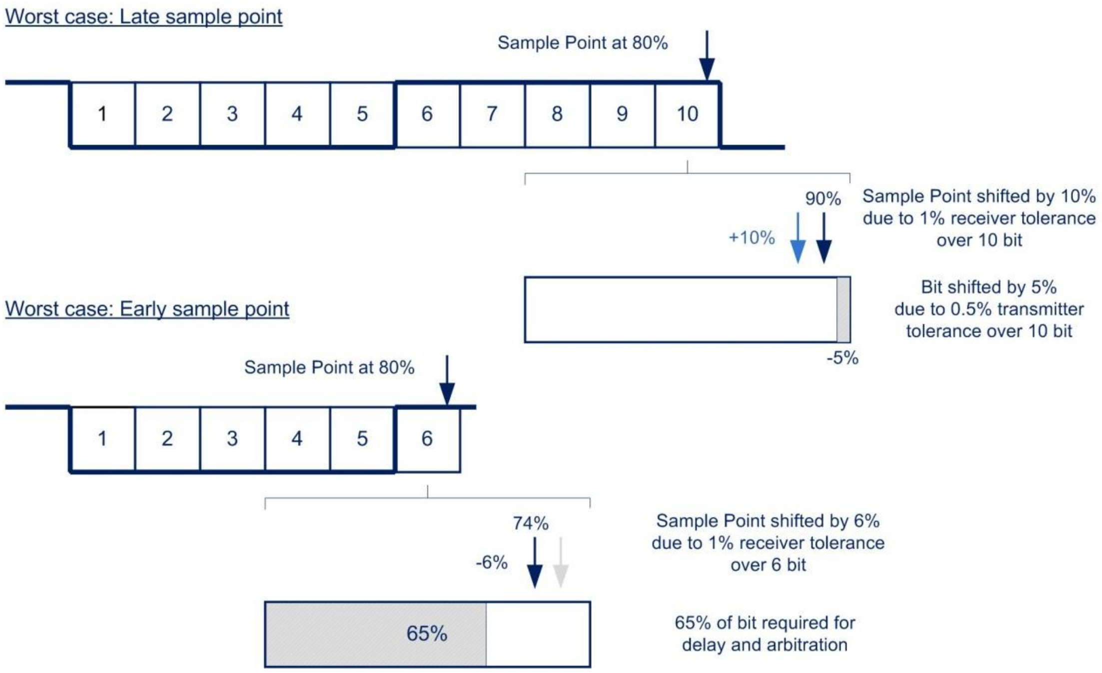

Figure 8: Timing of Bit Scanning

For active nodes, a quartz oscillator is typically used at the microcontroller. In selective wake mode, however, so as to keep the current consumption and costs low, an integrated RC oscillator is used in the transceiver and/or SBC. The particular challenge here involves balancing production fluctuations as well as temperature and long-term drifting. In addition, methods are available for calibrating the internal clock with the aid of the CAN data stream.

Troubleshooting

To prevent incorrect waking due to a disrupted CAN communication or that nodes can no longer be woken up, the CAN protocol handler incorporates an error detection facility in accordance with ISO 11898-1, chapter 10.9. This processes the errors Stuff Error, CRC Error and Form Error. Errors that occur downstream from the CRC delimiter are not processed. Incorrect CAN messages are not evaluated as wake-up messages, neither does any signaling of a detected error take place in the form of an error frame. Instead, an error counter is integrated that counts up the CAN errors detected and wakes up the local node in the event of a counter overflow. Accordingly, the microcontroller and/or control unit software is given the opportunity to take over of the troubleshooting.

Comparison of the wake-up messages with the mask

Every incorrectly received CAN message is compared with the stored wake-up mask. If the criteria for the comparison are satisfied, the microcontroller is woken up by the SBC and/or transceiver. For the comparison, the ID (CAN address), the DLC (data length) and the number of data bytes defined in the DLC are evaluated. For the ID, a mask (ID mask) and a comparative value (ID config) are used, the DLC must match the stored value. The data bytes are compared bit by bit, with a matching ‘1’ at a bit position leading to a wake up. This method makes it possible to use one ID to group up to 64 CAN nodes correspondingly into clusters and to wake them.

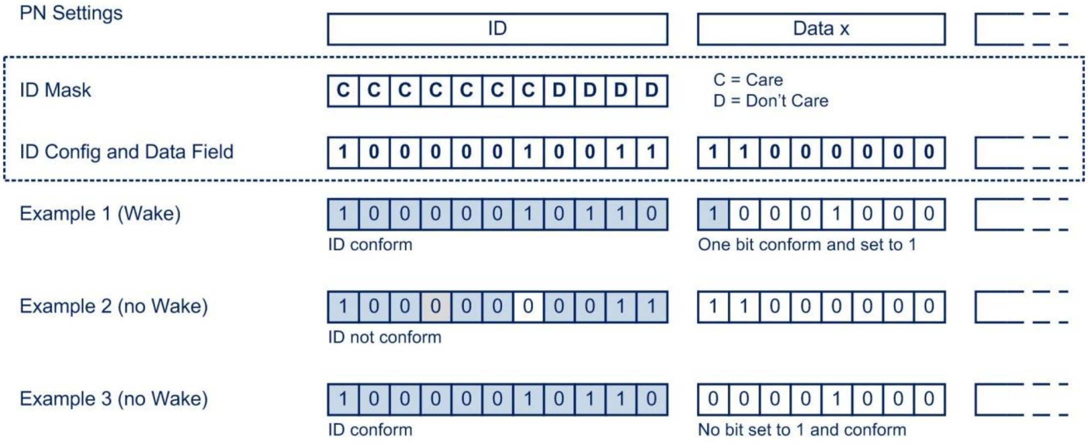

Figure 9: Basic Principle Masking of Wake Message

Time Out Function

To reduce the current consumption even further in parked vehicles and also have the ability to wake nodes selectively, a time out function has been implemented. If selective wake is activated in a node, but the bus remains inactive for a long period of time (typically 1 s), the SBC and/or transceiver deactivates the selective wake function. As a result of activity at the CAN bus, this is however reactivated at the latest after five CAN frames and is able to evaluate CAN messages. In selective wake mode, biasing of the CAN line is also activated, that is, the CAN lines are kept at 2.5 V mean voltage so as to prevent disturbances on the active CAN bus. In this mode, the node behaves as if it were an active node with a recessive output signal. In global wake mode with an inactive bus (detection of the WUP), biasing is deactivated, however.

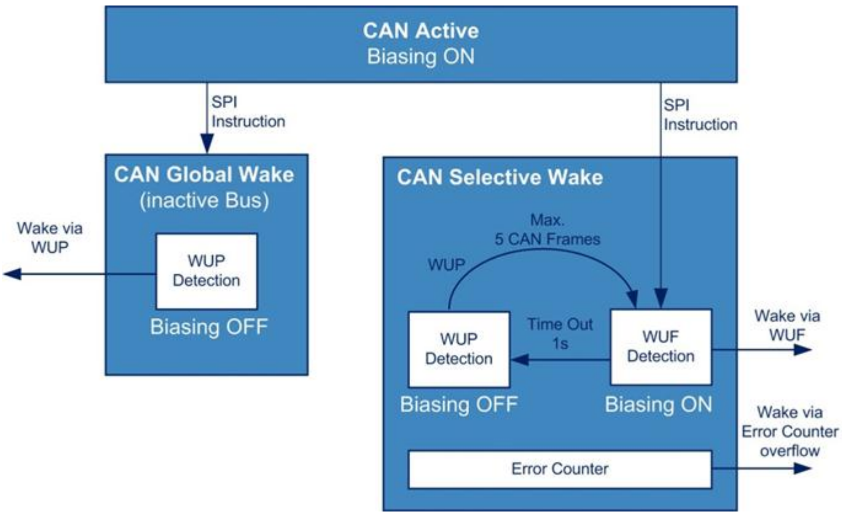

Figure 10: Selective Wake Mode Diagram

5 Implementation of Pretended Networking & ECU Degradation

5.1 Introducing Power States with ECU Degradation

ECU degradation includes different measures to ensure power saving, but in essence it enables AUTOSAR to introduce a power saving state. The ability to switch off modules during the operation of an AUTOSAR stack is now defined as a standard feature. The corresponding function also has free parameters for the integrator, enabling further module power saving functions. For example in the free parameters implementations are enabled to include features such as clocking down a module, which is currently used, but that is not needed in full operation. This is the very first step for saving power.

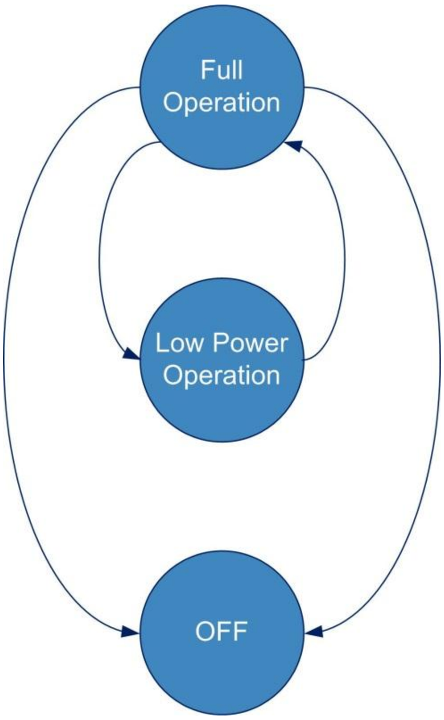

Figure 11: Power states of an ECU

ECU degradation also defines that single CPUs can be set to HALT mode (sometimes also referred to as IDLE). Why single CPUs? Firstly, an ECU may not run only one application in the future. By using mechanisms initially intended for safety features, modules can be protected against unwanted access from other CPUs, therefore it becomes possible to partition one microcontroller among applications. With low power states and reduced operation, one CPU might be sufficient to run all remaining operations. Therefore this mechanism is therefore essential for a power saving state.

The concept also includes 'switching runnables'. Even though runnables could be switched off before *ECU degradation*, the influence on software was re-examined and also rechecked in terms of switching single CPUs into HALT mode.

Even though each of these measures sounds simple, many side conditions had to be checked to enable operation of an AUTOSAR stack. For non-AUTOSAR systems, which are either already designed to incorporate power saving states or which are less complex, the described mechanisms should be easier to integrate. For systems with a similar complex environment, not designed for power saving, these measures are also advised for power saving, but integration may be difficult.

One measure was skipped during the definition phase of the *ECU degradation* definition. Central frequency scaling could enable ECUs to have two fixed points, one ECU full operation state, and one ECU power saving state, with two different frequencies globally distributed to the whole system. As in current systems, baud rates of all communication controllers need to be switched to a different value, as long as no additional clock for the communication part exists.

As yet, the fear to integrate this concept into AUTOSAR has been too big so far. Even though on microcontroller families like the AURIX™ [10], central frequency scaling and stable baud rates are supported. Central frequency scaling has the highest power saving potential, but having local frequency scaling on modules is a good first step.

5.2 Network Management with Pretended Networking

Pretended networking had the following requirements as a starting point: Save power, no influence on network architecture and fast wake-up times. The power saving features have been moved to *ECU degradation* to avoid duplicate definition of the same mechanism. Therefore the disabling of modules and the definition of how to get into HALT mode are not part of the concept. Instead, the concept includes a description to enable *pretended networking* mode itself and the wake-up procedures.

The *pretended networking* approach is split into two levels. The first level applies the *ECU degradation* approach while *pretended networking* serves as the communication backpack. The second approach assumes that the microcontroller has two power domains. One power domain includes the communication modules and another that includes the CPU and the rest of the modules. Both concepts have in common that during the power saving state, communication shall continue. The communication itself shall run on a reduced catalogue and a set of wake-up sources shall be switched active. The wake-up sources are not necessarily hardware events, but could also be a set of hardware/software events. This allows for more complex wake-up masks, with for example special bits being set within the data segment or data length violations. Therefore a CPU wake-up therefore does not necessarily mean that the complete software stack needs to be started, in case a wake-up event is declared invalid.

In level 1, where it is only necessary to leave the HALT mode, an extremely short wake-up time can be achieved. As the CAN module (or FlexRay in the future) is running all the time, the actual messages are still buffered and can be transferred to higher software layers. To enable the network management to run without waking the CPU up all the time, network management supervision is required. A timeout on certain CAN message objects is therefore used. The timer shall expire in case no network management message has been received for a certain time, to ensure the microcontroller is in bus sleep if no further network management is running. To avoid this timeout on other network ECUs, the network management of the ECU in *pretended networking* needs to continue running. Messages therefore need to be triggered also within *pretended networking* mode. An internal or external timer needs to trigger those messages to prevent any influence on the network.

In level 2, no CPU is available. As a result, all referenced operations including all wake-up source validation needs to be performed in hardware. With the main domain of the microcontroller shut off, the microcontroller wake-up time is dramatically increased. At wake-up the complete software stack needs to be reloaded with the requirement of resetting the microcontroller, but communication modules shall be read out first.

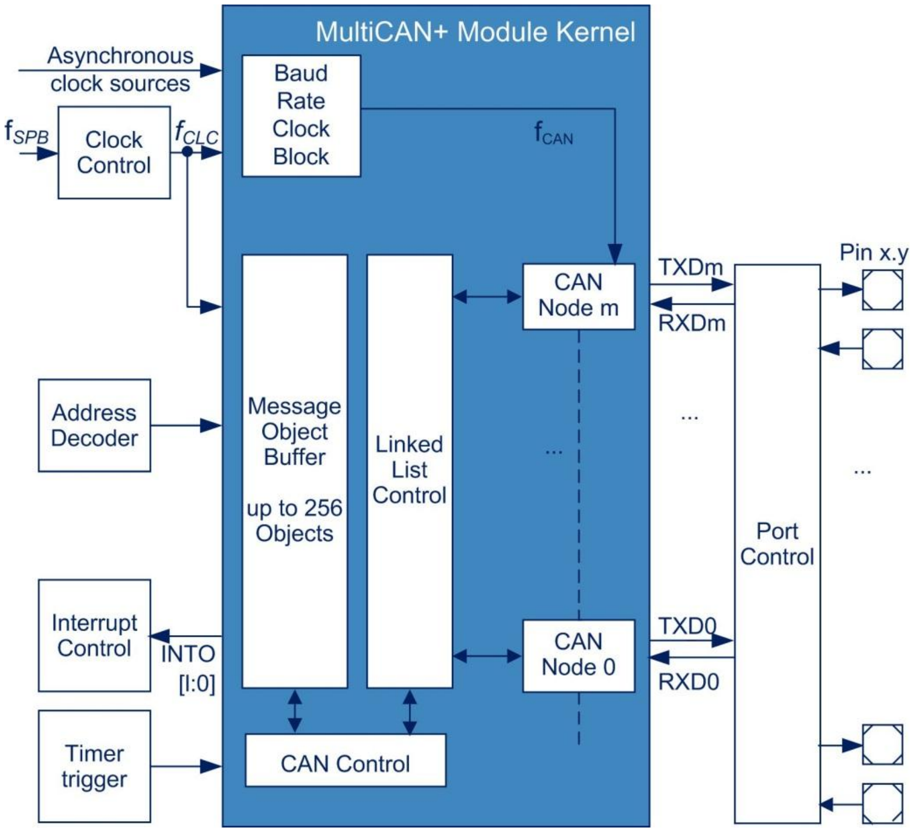

Figure 12: Block diagram MultiCAN+

Infineon has opted to support level 1, which has been further developed in the definition of the AURIX™ family of products. The definition of *pretended networking* had a direct influence on the definition of the incorporated CAN module. MultiCAN+ is an evolution of the already well-known MultiCAN module. As with recent MultiCAN implementation, this is now a module having two clock domains, but it includes the option to have separately configurable interrupt for each single message object, and the possibility to overwrite message objects.

Some further improvements now support *pretended networking*. Network management (NM) timeout is introduced. For each node, one timeout is available, which allows multiple message objects to be connected to it. In case none of the selected message objects have received a message for the programmed maximum timer value, a timeout will occur and an interrupt will be issued. A similar mechanism is used for sending NM messages. Three timers have been introduced. The timers are programmable and are started on request. With the three timers, three message objects can be programmed, which will be sent without any software interaction. By shifting the NM supervision and the operation to send NM messages into the CAN module, less modules need to be active and the sleep time of the ECU is extended.

The mechanisms introduced into the MultiCAN+ module, which is available in the AURIX™ family and the possibilities provided by *ECU degradation* and *pretended networking*, enable power savings for ECUs which need to be partly awake for safety reasons or which cannot use *partial networking* because the wake-up time is far too long to be compliant to on-board diagnosis (OBD) regulations. As a result, a current car platform can be changed to consume less power without having to make changes to the car architecture. The only remaining question then is: What is the power saving potential?

6 What savings can be achieved?

This is surely the crucial question for architects of electrical networks. Based on uncertainties and the additional effort to implement the measures discussed, there must be a substantial impact on energy balance in order to force an introduction into a car architecture. Saving potential is strongly dependent on the question of how many nodes are suited to enter energy saving modes while the car is in operation. Deactivated engine control units or airbag systems are not usually an option. However deactivation of the electrical seat control while driving (beyond a certain vehicle speed for example) or a sleep mode for a door control unit (as the door functions are rarely used while driving) can be seriously considered.

The example of the door control unit shows the complexity of this topic. In case of an accident the door lock needs to be unlocked without a tangible delay; therefore a fast wake of a sleeping door module seems to be necessary. Furthermore, driver interaction has to be taken into account, for example opening the window or moving the seat while driving, which requires a response without a noticeable delay for the customer.

The concepts of partial and pretended networking can co-exist in a car, so in the future both concepts can contribute to the total power saving. A sample calculation based on a typical E/E network of a mid-size car in the premium segment with 40 ECUs has been presented for the first time in 2010 [7], but has been a little updated (see Figure 13) in the meantime in order to show potential of co-existence of partial and pretended networking. This calculation covers average current consumption without consideration of external loads.

Even higher value savings have been reported. A German OEM has mentioned a target value of savings in the range of 2.6g CO_2/km for introduction of *partial networking* into new network architecture [1]. However in order to achieve such a high value, additional measures are necessary, such as the deactivation of entire subnets (LIN) or incorporation of deactivated peripheral functions into the calculation.

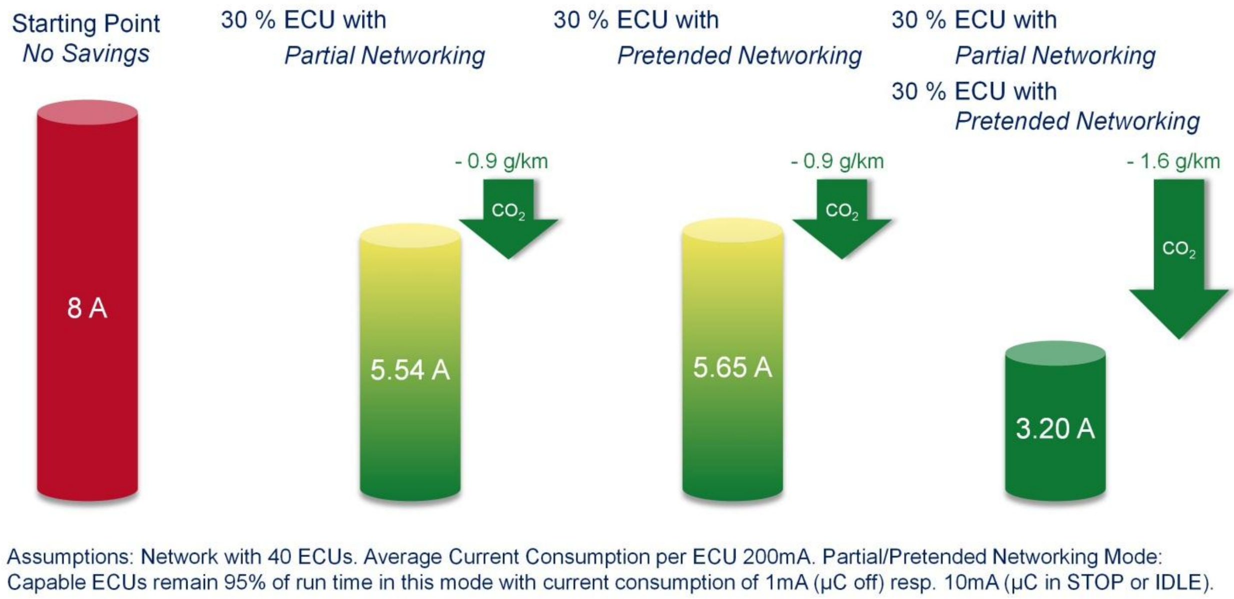

Figure 13: Results of Sample Calculation of Achievable Current Savings

7 Conclusions

A first implementation of *partial networking* has been announced for 2013/2014 [4], where the entire E/E network and hardware/software architecture will be adapted accordingly.

Although the saving potential of *pretended networking* is smaller than *partial networking*, it is much easier to integrate into existing networks and can be used in ECUs which need a very fast response at any time. One car manufacturer has announced an introduction scenario in a real car platform for 2015 onwards [5]. The Tier1 community still has concerns regarding the impact on reliability of given ECUs, and this needs to be seriously considered.

Taking a mid-term view, all the described energy saving methods will probably be implemented; parallel occurrence in one car architecture is possible. *Partial networking* requires special transceiver devices or SBCs. Those devices have already been announced by several semiconductor vendors, even though ISO 11898-6 is still not finalized. *Pretended networking* can be used to start power saving right now, without changes to the network architecture, enabling power saving without any special hardware. All measures taken in new microcontroller devices further increase the power saving potential.

The joint effort of carmakers, suppliers of electronic control units and semiconductor vendors will result in highly efficient, energy saving solutions with the level of quality and reliability demanded by automotive applications. These solutions will help to support the integration of conventional comfort and safety functionality into vehicles with pure electrical drive trains.

References

[1] Frickenstein, Wilhelmi, Hudi, Michael, Tanneberger, Milke
New Automotive Standards
15th Int. Conference 'Advances in Automotive Electronics' · Ludwigsburg · 2011

[2] Scheer, Fuchs, Grzemba, Liebetrau
Selektiver Teilnetzbetrieb und Clusterbildung – Erprobung unter realen Fahrbedingungen
Elektronik im Kraftfahrzeug 2011 – 15. Internationaler Fachkongress des VDI
VDI-Berichte 2132 · ISBN 978-3-18-092132-7 · VDI Verlag · Düsseldorf · 2011

[3] *GMW3089 - GMLAN Single Wire CAN Physical & Data Link Layer Specification;*
General Motors Corporation

[4] Meyer, Esch, Linn
Teilnetzbetrieb – Abschaltung inaktiver Steuergeräte: Anwendung, Standardisierung und Absicherung
Elektronik im Kraftfahrzeug 2011 – 15. Internationaler Fachkongress des VDI
VDI-Berichte 2132 · ISBN 978-3-18-092132-7 · VDI Verlag · Düsseldorf · 2011

[5] Speh, Dr. Wille
Pretended Networking - Migrationsfähiger Teilnetzbetrieb
ATZ Elektronik 04/2012 (August 2012) page 264 ff.

[6] Butzkamm, Bollati
Partial Networking for CAN bus systems: Any saved gram CO2/km is essential to meet stricter EU regulations
13th International CAN Conference · Hambach Castle · 2012

[7] T. Liebetrau
Wege zur Energieeinsparung im Fahrzeugnetzwerk - Teilnetzbetrieb und skalierbare Funktionalität
Elektronik automotive congress · Ludwigsburg · 2010

[8] T. Liebetrau
Teilnetzbetrieb in komplexen Netzwerkarchitekturen – Migration in bestehende Netzwerke
Elektronik im Kraftfahrzeug, Günter Schmitz (Hrsg.)
Haus der Technik, Fachbuch Band 123, expert Verlag · ISBN 978-3-8169-3110-2

[9] Monetti, Otter, Ulshöfer
Spritverbrauch senken und Reichweite erhöhen - System-Basis-Chip für den Teilnetzbetrieb am CAN-Bus
Elektronik Automotive · WEKA Fachmedien · November 2011

[10] *Infineon Introduces Microcontroller Multicore Architecture for Automotive Applications*
Infineon News Releases, October 2011

List of Figures

List of Abbreviations

AUTOSAR	*Automotive Open System Architecture; www.autosar.org*
CAN	*Controller Area Network; www.can-cia.org*
CPU	*Central Processing Unit*
E/E	*Electrics/Electronics*
ECU	*Electronic Control Unit*
IEEE	*Institute of Electrical and Electronics Engineers; www.ieee.org*
INH	*Inhibit*
ISO	*International Organization for Standardization; www.iso.org*
IC	*Integrated Circuit*
IT	*Information Technology*
LAN	*Local Area Network*
LIN	*Local Interconnect Network; www.lin-subbus.org*
MOST	*Media Oriented System Transport Protocol; www.most-cooperation.com*
NM	*Network Management*
OBD	*On-Board Diagnosis*
OEM	*Original Equipment Manufacturer*
RX	*Receiver*
SBC	*System Basis Chip*
SPI	*Serial Peripheral Interface*
STP	*Shielded Twisted Pair Cable (single or multiple pairs)*
SWITCH	*Selective Wake-Up Interoperable Transceivers for CAN High-Speed Networks (OEM Working Group)*
TRX	*Transceiver*
TX	*Transmitter*
UTP	*Unshielded Twisted Pair Cable*
WUF	*Wake-Up Frame*
WUP	*Wake-Up Pattern*

Ask Infineon. Get connected with the answers.

Where you need it. When you need it.

Infineon offers its toll-free 0800/4001 service hotline as one central number, available 24/7 in English, Mandarin and German.

Our global connection service goes way beyond standard switchboard services by offering qualified support on the phone. Call us!

- Germany…..................... 0800 951 951 951 (German/English)
- China, mainland............ 4001 200 951 (Mandarin/English)
- India.............................. 000 800 4402 951 (English)
- USA…........................... 1-866 951 9519 (English/German)
- Other countries .... 00* 800 951 951 951 (English/German)
- Direct access…............. +49 89 234-0 (interconnection fee, German/English)

* Please note: Some countries may require you to dial a code other than "00" to access this international number, please visit www.infineon.com/service for your country!

Where to Buy

Infineon Distribution Partners and Sales Offices

Please use our location finder to get in contact with your nearest Infineon distributor or sales office.

www.infineon.com/WhereToBuy

Infineon Technologies – innovative semiconductor solutions for energy efficiency, mobility and security.

Published by
Infineon Technologies AG
85579 Neubiberg, Germany

Visit us:
www.infineon.com

ATTENTION PLEASE!
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie"). With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

INFORMATION
For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office (www.infineon.com).

WARNINGS
Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office. Infineon Technologies Compo-nents may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable
to assume that the health of the user or other persons may be endangered.