Cost and energy efficiency for partial / pretended networking on CAN

Roland Lieder, Renesas Electronics Europe

There are two approaches for power saving within the "Efficient Energy Management" context of AUTOSAR, which are related to CAN communication: "Partial Networking" and "Pretended Networking".
When looking for the maximum energy saving potential, both concepts seem to have something in common: The necessity of additional hardware, in order to have highest efficiency.
The newest generation of microcontroller families is offering very low power consumption characteristics. This makes it worth to consider an implementation of either Partial- or Pretended Networking in software, instead of using hardware based solutions.

A competition of concepts?

Both AUTOSAR concepts ([1], [2]) have their advantages and drawbacks. Regarding their target, which is to improve energy efficiency (and thus to reduce power consumption of an embedded control unit (ECU) in the car), both have the same. However, the approaches are different and can even be combined, if the applications are allowing that. Therefore, it is up to the system architect, to find the trade-off (which method to use for which case) and finally to decide about the implementation of either one or both.
Summarized, we can find the following properties of the methods:

Partial Networking [1]

Network nodes are completely switched on and off by dedicated messages. Thus, the power saving intelligence in the network is located in one ECU or a dedicated ECU group.
A switched off node is not supporting the network management (NM) any more, and thus the NM must consider switched-off nodes.
On hardware side, a selective wakeup capable CAN Transceiver is required.
In fact, Partial Networking achieves the highest power saving effect, but requires system redesign on NM side.

Pretended Networking [2]

Network nodes are internally suspending their activities based upon local decisions, so there is local power saving intelligence.
Even a switched off node is still supporting the NM, so that NM and communication matrix will not need significant changes. This also ensures compatibility with other nodes, which are not supporting this feature.
On hardware side, depending on the target of the power saving amount, intelligent communication units (ICOM) may be required.
As a result, Pretended Networking will achieve a significant power saving effect, but its efficiency may have variations, which are difficult to forecast. It will not reach the power saving effect of Partial Networking during its switched-off phases, but in average during a driving cycle (like NEDC), it may become even more efficient than Partial Networking. This would be the case especially, if fast reaction times are required or if the switched-off phases of Partial Networking cannot be used very often. The biggest advantage of Pretended Networking is however, that it ensures almost seamless compatibility within old or existing systems. This allows the exchange of single ECUs without redesigning the whole system, thus improving the system step by step.

The aim of a microcontroller vendor

At first, and commonly for Partial- or Pretended Networking usage, the energy efficiency of the microcontroller itself must be increased as much as possible.
At the same time, special constructions or designs should be avoided, in order to keep the cost to value ratio attractive.
Instead, the emulation of either method or principle by software shall be preferred. The advantages are obvious: software emulations have higher flexibility and less risk during their introduction phases. Changes in the system or identified problems can be solved by software update, and will not require hardware exchanges.
Even more, if a microcontroller reaches a high amount of energy efficiency, the software solutions are in competition with the hardware solutions. Even though the efficiency of hardware solutions is not fully reached by software emulations, the advantages of the software solutions will be taken into account when determining the trade-off. The question typically is: does the amount of additional power saving by hardware means really pay out the risk, cost and limitation of flexibility? We will let decide the customer about that.

Comparing the characteristics

The behavior of Partial- and Pretended Networking during a switch-off phase can be understood easily when looking at a timing chart.

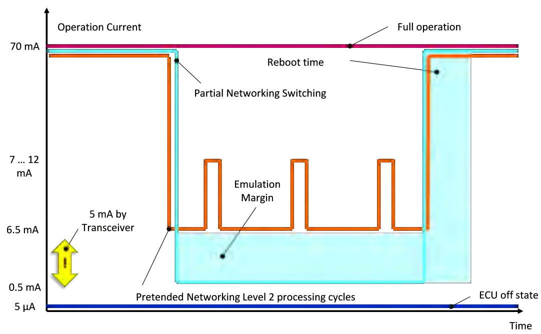

Figure 1: Partial- & Pretended Networking

The figure shows power consumption values of the microcontrollers and CAN transceivers of the current generation. We will come back to these values later.
While Pretended Networking periodically consumes power during its NM maintenance and (in case of emulation) software post-filtering of relevant CAN messages, Partial Networking stays permanently at a very low power level.
As a drawback, Partial Networking will require a significant reboot time, when resuming the operation again.
The low power level of Pretended Networking is determined by the CAN transceiver, which must stay in a receive-only mode, plus a small adder of power of the microcontroller, which is in a hibernation mode.
The difference of the low power levels is indicated as "Emulation Margin". Most of this margin is caused by the transceiver – here, we can see most improvement potential.
Obviously, the situations are different, when Partial- or Pretended Networking can go to the switched-off level. Therefore, when considering the trade-off, such time cycles must be taken into account. The significant questions are how often and how long the switched-off level can be achieved during the relevant usage or test cycle.

A field project [3]

In a field project using Renesas V850/Fx3, it was proven that Pretended Networking can achieve a reduction of power by 56%, if an additional ICOM hardware is installed in (or beside) the microcontroller. Here, it was clearly identified, that the power reduction was mostly given by the fact that the CPU could be fully stopped while the ICOM was still running, and by the ICOM's capability to filter CAN messages by 100% (including data).
Without ICOM, the power saving could be estimated to be around 46%. Here, the CPU waits in HALT mode, and a certain number of messages must be still processed by software, because the CAN controller cannot filter on data contents.

What has been achieved?

Let’s compare the previous [4] and the actual [5] microcontroller generations of Renesas, regarding their applications for Partial- and Pretended Networking.

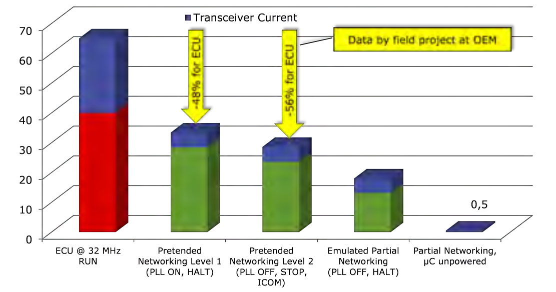

Figure 2: Previous Generation Power (mA)

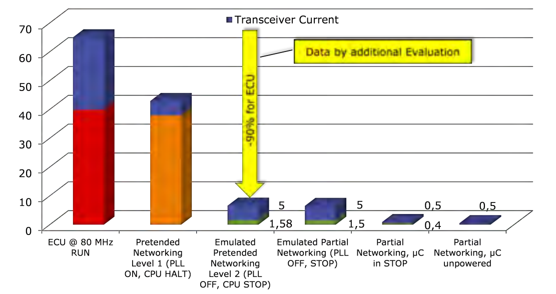

Figure 3: Current Generation Power (mA)

The figures above are the showing improvements of the current consumption characteristics of the previous V850/Fx3 series (130 nm technology) and the current RH850/F1L series (40 nm).
First of all, there is no significant increase of the current consumption, even though the processing speed has increased to 80 MHz.
At the same time, emulation of Pretended Networking has become attractive: The separation of the CAN controller from the CPU operation in conjunction with very low leakage currents. Even though the CAN controller still needs software support to filter data wake up conditions, the ICOM has become unnecessary – the emulated level 2 of Pretended Networking has reached its efficiency.
We could even emulate Partial Networking, if there was a CAN transceiver, which has significantly reduced power consumption in a receive-only mode.
In case that for Partial Networking the reboot time is significant, it is considerable to leave the microcontroller in a powered state during the switch-off phases. Here, the trade-off between power saving and reaction time is has to be considered.
If we are looking at the overall situation, we can say that the recent technology step of the new generation devices is moving the trade-off borders.

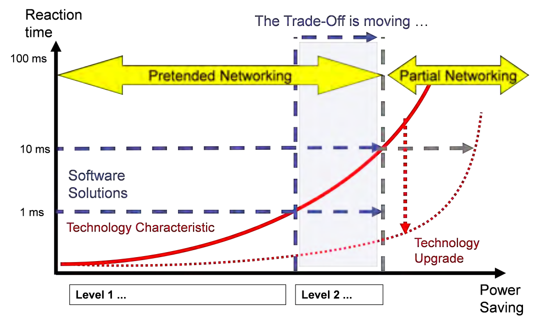

Figure 4: Technology moves the trade-off

Due to the technology step of the actual microcontroller generation, lower current consumption values can be reached, making emulated software solutions more attractive. At the same time, power saving effects of Partial Networking (with transceiver hardware) and software solution are coming closer regarding the absolute power consumption levels.

The evolution of Pretended Networking

Due to the reduction of operation and standby power consumption values, the Pretended Networking concept has performed a considerable evolution.
Starting from before its announcement, until today its potential of power saving has increased.

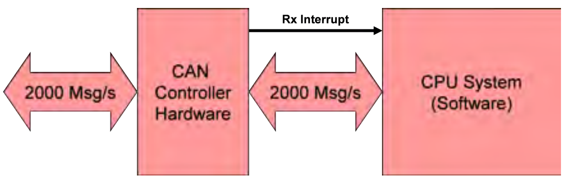

Figure 5: Before Pretended Networking

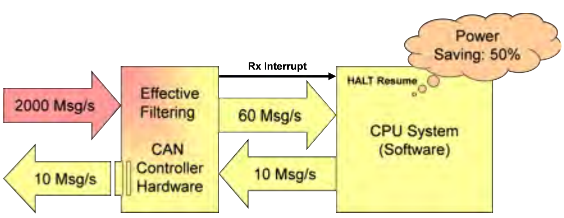

Figure 6: Pretended Networking Level 1

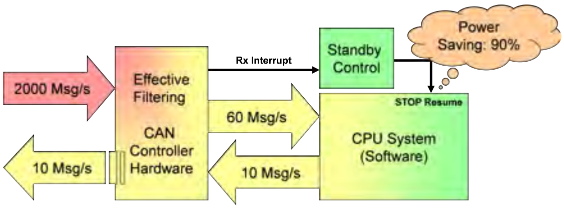

Figure 7: Emulated Pretended Networking Level 2

When considering this evolution, there was no ICOM hardware needed for it.
At this point, therefore it was concluded that an ICOM hardware could improve this evolution only in a minor way (when fighting against the last 10% of power that were remaining).

Conditions of our evaluation

Pretended Networking was evaluated for its power consumption during a switched-off phase, in order to have comparable results. In the switched-off phase, we are assuming the following:

- There is no application job to be executed by the CPU. Therefore, it enters a hibernation mode (STOP), where its clock is suspended until an interrupt occurs. Other parts of the microcontroller are powered off, which are not related to CAN and CPU execution.
- The ECU has to send a NM maintaining message every 100 ms.
- The ECU receives 6 filtered messages by the CAN controller within 100 ms, where it has to verify the data content and decides that a wake up is not required.
- When checking the data content, the CPU is running with a degraded clock of 8 MHz only.
- All other messages on the CAN bus can be filtered by the CAN controller.
- The CAN controller stays clocked with a communication clock of 8 MHz.

Figures 8 and 9 are showing the shapes of the current consumption of the microcontroller.

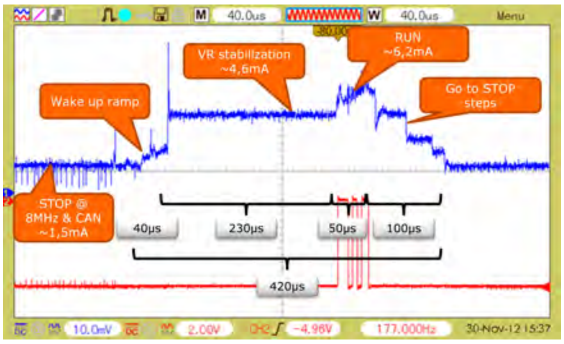

Figure 8: One of 6 reception cycles

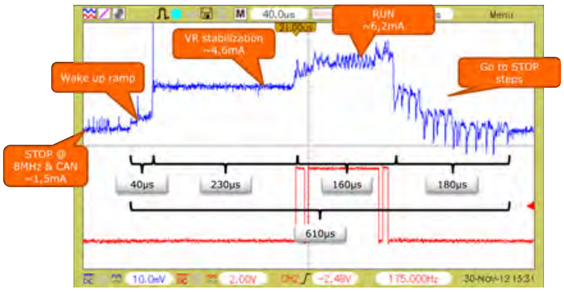

Figure 9: The transmission cycle

For our overall power consumption value of the system, we have calculated the averages of the current waveforms and added the transceiver current of 5 mA during reception and an average of 25 mA during the transmission phase.
The average current consumption of the Renesas RH850/F1L microcontroller itself is at 1.58 mA.
The microcontroller is ready to run after a wake up after 300 μs at latest. As the CPU and RAM stays powered, for this restart only the unpowered sections of the device need to be reinitialized.

Where's the commercial cost? – The chain of efficiency

Besides the cost of electrical power consumption in a vehicle during its lifetime (which is fuel cost), from 2015 onwards, the homologation tax will increase significantly.
Electrical power is coupled with CO_2 emission, on which the tax is put on.
Figure 10 shows the relationship between CO_2 emission and power consumption; which is in fact simplified as linear relationship of energy transformation.

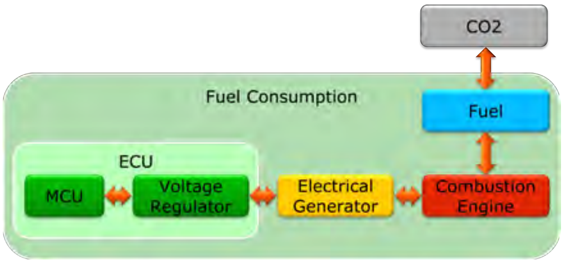

Figure 10: Energy transformation in a car

For each transformation step, a worst case and a best case assumption can be made:

Table 1: Transformation Efficiency

Step	**Unit**	**Best Case**	**Worst Case**
Fuel combustion CO_2 emission	g CO_2 / l	Gasoline 2360	Diesel 2650
Fuel combustion energy	kWh / l	Diesel 9.8	Gasoline 9.0
Combustion engine efficiency	%	Diesel 30	Gasoline 25
Generator efficiency	% at 14.4V	77	70
ECU voltage regulator efficiency	% from 14.4V to 5V	DC-DC 93	Linear 35

When looking at this table, at first it's becoming obvious that there are many other items in a car, where energy efficiency can be improved. Even more, this is once more raising the question why an ECU should save much power, while there are still other components which are consuming it. But the answer here is: the number of ECUs in a car increases more and more, and therefore their influence on the overall efficiency of a car increases similarly.
For our further investigations, we have selected an assumption for a "standard" car. This car would have a gasoline engine with low generator efficiency and use linear voltage regulators within its ECUs.
At the end of the calculation, the outcome is that for every 1 mA of ECU current consumption, 14 mW of power is consumed, resulting in 0.7 mg CO_2 emission per km.
For every g CO_2 per km, a homologation tax of 95 € has to be paid per car.
In the end, if at least 20 ECUs could save about 60 mA each during the NEDC driving cycle, about 80 € tax could be saved per car.

Below the line:
savings, targets and strategy

The total tax for every new homologated car will be much higher – the CO_2 emission is mostly related to other reasons than the electrical power consumption of the ECUs.

However, on every component the focus must be applied, otherwise the target becomes unreachable.

Besides that, ecological design is a strong factor for advertisements.

Each development has risks and cost factors, which are in competition with political requirements or forces as tax. Therefore, the big equation always has to be put on the table for every ECU:

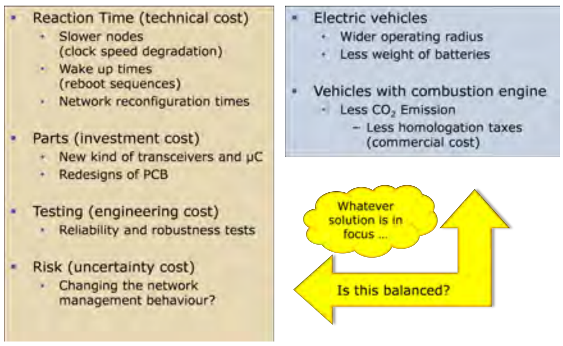

Figure 11: Keeping the balance

The new generations of microcontrollers of Renesas now are allowing new aspects to be highlighted:

- Some part of the risk of low power concepts can be moved to software by emulation.
  Software allows updating and fixing of errors without exchanging components.
- Harsh changes in sensitive system properties like NM can be avoided by choosing alternative power saving concepts.
  Pretended Networking can reach a very high efficiency level.
- Flexible adaptation on individual needs.
  When choosing software based solutions (emulation) of either

principle, conditions and efficiency can be adapted to the individual ECU requirements, for example the reaction time.

- Smooth introduction schedules. When using Pretended Networking, it's not required to exchange or update all ECUs at a time. Instead, the behavior and benefit of individual ECUs can be done step by step during running series.

These new aspects have been made feasible with the new RH850/F1L microcontroller generation of Renesas. Efficient energy management is no longer limited to sophisticated hardware solutions, but now also reachable by software means.

Even more, software solutions are offering less risk and higher flexibility of implementation.

For this reason, Renesas is recommending the software emulated level 2 of Pretended Networking as a best choice solution for efficient energy management on CAN.

In a car, quite a lot of ECUs can be put into this focus. Therefore, the list of potential optimizations is long.

Some candidates can be mentioned:

- Keyless Entry
- Trailer Control
- Climate Control
- Door Module
- Sliding Doors
- Seat Control
- Boot Cover Control
- Convertible Top Control
- Sunroof Control
- Park Heating Control
- Parking Assistant
- Dynamic Headlamp Control
- Rear View Camera
- Charging Manager

... and most probably many more.

Roland Lieder
Renesas Electronics Europe GmbH
ABG / M&E / CT / N&P
Arcadiastraße 10
DE-40472 Düsseldorf
Tel.: +49-211-6503-0
application_support.can-eu@lm.renesas.com
www.renesas.eu

References

[1] Support of energy efficient technologies by AUTOSAR – Partial Networking & Co. by Dr. Marcel Wille, Volkswagen AG Product Day 2011, Fellbach, November 16th 2011
[2] ATZ Elektronik 8/2012: “Migrations-fähiger Teilnetzbetrieb – Pretended Networking“ by Jörg Speh, Volks-wagen AG
[3] NEC Electronics Data Sheet 2008 V850ES/FJ3 U18567EE1V3DS00
[4] Renesas Electronics Preliminary Data Sheet 10-MAY-2013 RH850/F1L R01DS0170EJ0040 Rev. 0.40