



Tauria



Simplifying life with embedded control systems engineering

Measurement

Systems

Electronics

Firmware

CONSULTANCY

Ultra Low Power Wired Field-bus

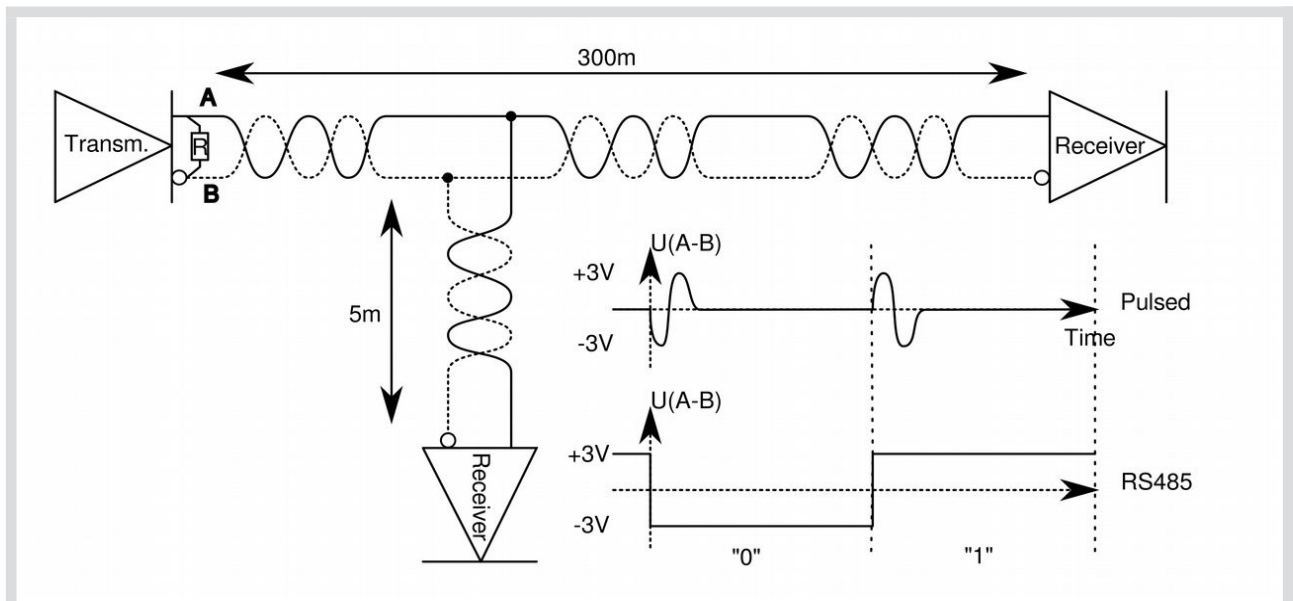
The communication focus for Internet of Things (IoT) has been mainly wireless with battery powered devices. For fixed installations where power consumption is much higher than a battery would provide during a 10 year period, wired power with wired communication would be needed. Example systems are heating, ventilation and air conditioning. Using wired communication would leave wireless space free for non-stationary communication.

On the physical communication layer the world has many wired protocols,

like RS485, M-Bus and so on. We believe that for IoT the power demand for the communication must be even lower than the existing protocols provided. Today's methods require careful wiring because the signal reflections on the bus will make the data reception fail.

We are proposing a physical communication layer protocol that uses short pulsed signals. The data transmission rate will be comparable to RS485. We propose to transmit short pulses of energy on the bus network when the state of the transmitted information bit changes

from "1" to "0" or from "0" to "1". We propose electrical signalling that does have 3 states: "1", "0" and "n". State "1" is transmitted on the raising edge of signal voltage U between wires **A** and **B**. State "0" is transmitted on the falling edge of signal voltage U between wires **A** and **B**. State "n" means nothing is transmitted and the data remains "1" or "0". Voltage at the bus state "n" is in the range -0.5V to 0.5V. The bus requires only one signal terminator **R** which must consume the reflected signal energy before the moment of the next pulsed signal transmission.



Most of the time, the messaging bus would not be powered in state "n" contrary to the existing systems. Therefore, less energy is spent for the data transmission. The transmitter and the receiver can be designed to consume energy only when the short pulse is transferred.

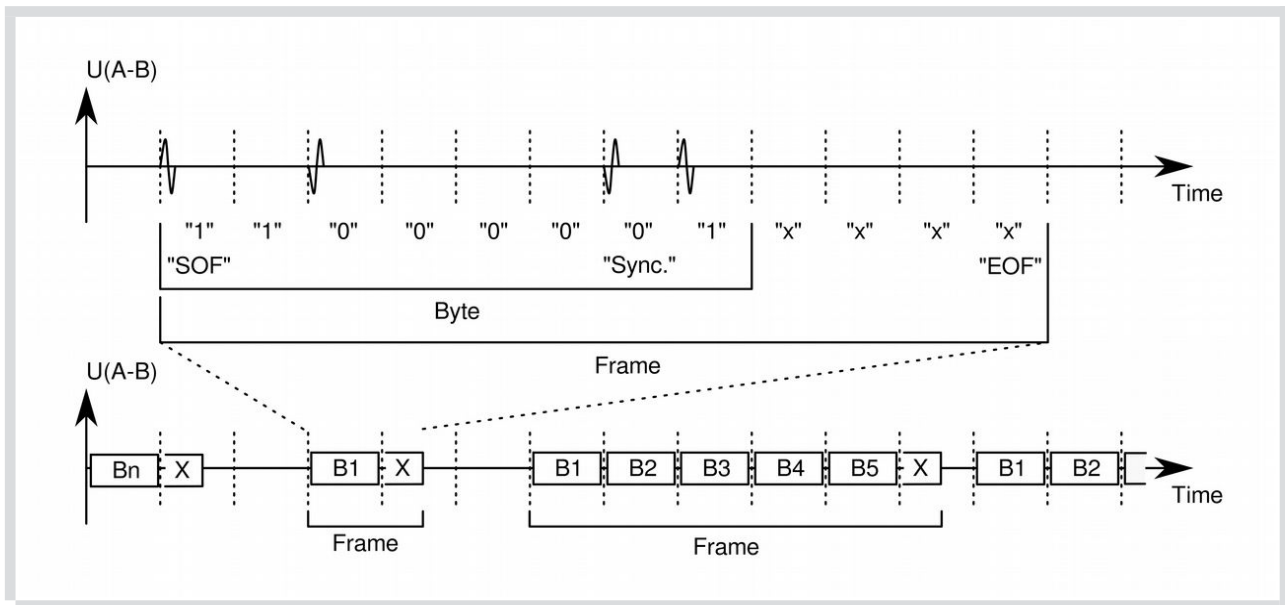
The reflections on the bus network do not interfere with the data transmission because no reflections

will arrive to the receiver before the first edge of the data pulse. The wiring becomes more robust in comparison to RS485.

The bus may be around 300m in length. Short branches in the range of 5 meters are permitted. Estimated half-duplex data rate would be 15.6 kB/s.

When the transmittable data does not

change during the time of 4 bits, then the pulse of the last state will be re-transmitted for synchronisation of the receiver with the transmitter. The end of data frame (EOF) will be formed by keeping the transmitter in the state "n" during the time of 5 bits. The first pulse for bit "1" or "0" after the EOF is at the same time as the start of the data frame (SOF). Data is transmitted in multiples of full Bytes (8 bits).



Partially transmitted Byte **X** must be discarded because it contains no data. Before ending the transmission with EOF, multiple Bytes can be transmitted in tight sequences. The Bytes transferred together will behave as a single stream of bits and there is no need to transmit a pulse at the beginning of every following byte.

The described frame forming requires a Byte transmission time of 8 bits. In RS485 usually the UART frame is used. UART requires a Byte transmission time of 10 bits. The pulsed method has therefore a 25% higher data density at the same baud rate.

Energy spent for forming the signal on the bus is proportional to the voltage in square of the signal and the time length of the signal pulse inside the wire and inversely proportional to the impedance of the signal cable. When signal voltage and line impedance are the same compared to RS485 set-up, then the time length is going to be the only difference. In the case RS485, the

voltage is constantly applied over the entire time of the frame transmission. In the pulsed method, the voltage is applied in short slices. As shown before, the amount of pulses depends on the message transmitted.

At the worst case, the binary message **10101010** or **01010101** requires one pulse for every bit. If the pulse duration is 1 micro second and the bit time is 8 micro seconds, then sending pulses on the bus requires 8 times less energy per data bit. UART transmits 10 bits per Byte and pulsed method transmits 8 bits per Byte. Sending pulses on the bus requires 10 times less energy per data Byte than with RS485 at the worst case.

At the best case, the binary message **00000000** or **11111111** requires 2 pulses: Start and Sync pulses. It is 4 times less than at the worst case scenario. The bit which requires a pulse is drawn in bold. At the best case the pulsed method requires 40 times less energy than RS485.

The real energy savings would be

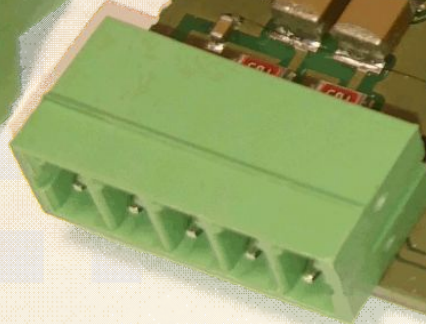
somewhere in between the best and the worst case scenarios. For example, the message "Hello World!" transmitted in the ASCII binary encoding (**01001000** **01100101** **01101100** **01101100** **01101111** **00100000** **01010111** **01101111** **01110010** **01101100** **01100100** **00100001**) would require 51 pulses in total. This is on average 4.25 pulses per Byte and the savings per message is 18.8 times compared to RS485.

The shown signal encoding with short pulses is easy to implement in the hardware. It is possible to implement more complex encoding, where at the worst case scenario the energy savings would be 16 times compared with RS485. The best case scenario would stay almost the same and data density would be the same as for UART. Transmission of the "Hello World!" message would be 23.9 times more energy efficient than on RS485. Because of the high complexity of the mentioned encoding, it is left out from this article.



Evaluation Board

... for ULP Field-bus



Introduction:

Today Internet of Things is wireless. Wireless is good for many situations, and it is good to keep wireless ether space silent as Silence is Golden. Therefore Earth, Moon and Mars are going to need a WIRED NERVE SYSTEM of smart buildings and machines. Benefits that Wired Internet of Things brings:

- No batteries and no chemical waste management – power over wire.
- Higher bandwidth– the signal path is narrow line in space.
- Open and easy to understand protocol.
- Suitable for low end and ultra cheap electronics.
- World's record on Low power consumption on wired communication.
- Physical barrier against DoS attack.

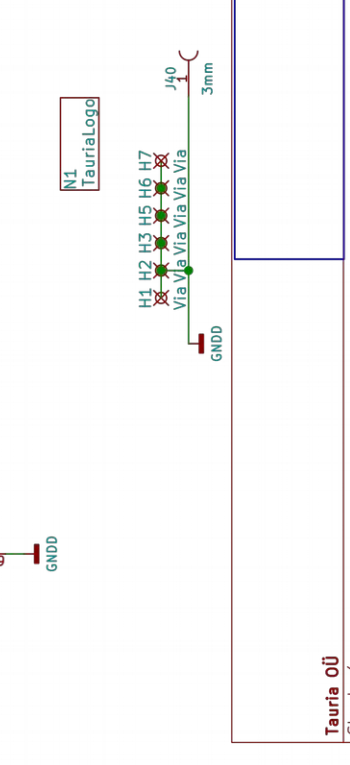
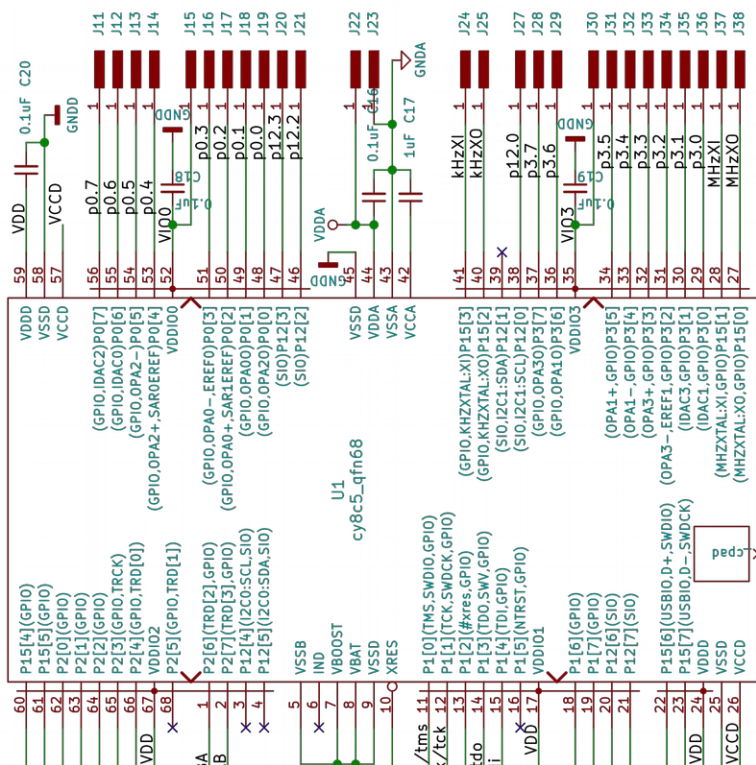
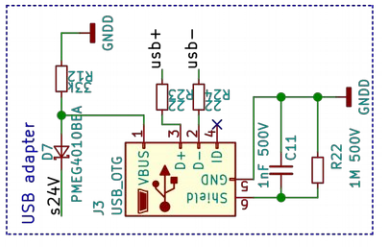
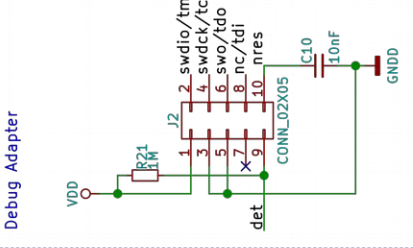
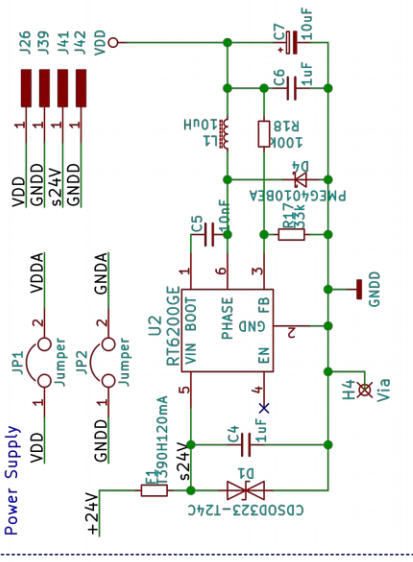
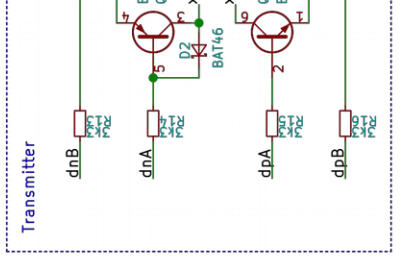
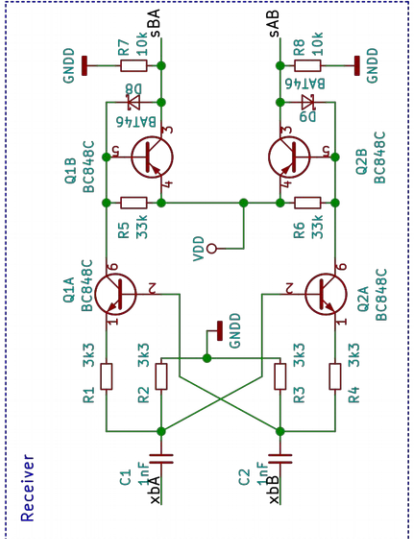
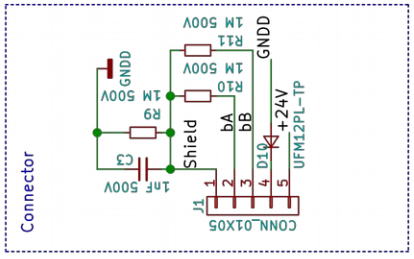
This evaluation board offers Transceiver circuit built from few transistors. With this evaluation kit you can participate in World Changing transformation of technology of things communication. The target is:

- Open standard for building Wired Bus Network in true IoT spirit.

- magnitude times lower energy consumption than RS485.
- State of art energy saving on message transfer with Short Pulses.
- Automatic power-on power off during single pulse transfer.

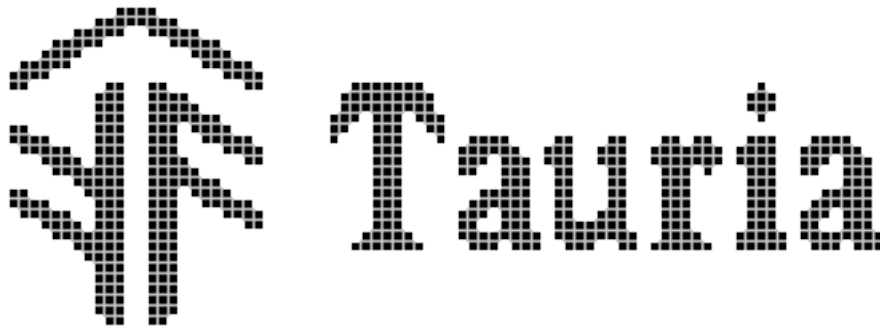
Board Info:

- Cypress 8x8mm QFN-68 PSoC 5LP MCU with programmable logic and ARM Cortex M3 CPU.
- USB connection with PC.
- 5 Wire terminal for 24V bus power, twisted-pair signal and shielding.
- Switched DC down converting power supply.
- 36 IO pin-hole terminals for MCU pins 24V power, 3.3V power for interfacing with your application.
- Cypress UDB and Datapath components for transmitting and receiving (does not require specific libraries), supports DMA and hardware CRC calculation.
- We target the same UDB and Datapath components for PSoC 4 too.



Tauria OÜ
 Sheet: /
 File: TauBusModule.sch
Title: TauBus Evaluation Module
 Size: A4 Date:
 KiCad E.D.A. kicad 4.0.7

Rev: 1.0.1
 Id: 1/1



About Tauria

Our Mission:

"Simplifying life with embedded control systems engineering"

Our Services:

We can help you from idea to production with:

- Requirement research, analysis and documentation.
- Proof of concept evaluation.
- Firmware and software development for various micro-controllers 8 & 32 bit and for Linux and servers. (ASM, C, C++, PHP, Python, Javascript, Cypress PSoC UDB components, Verilog)
- Electronic & PCB design. (Eagle, KiCad)
- Mechanical & 3D design. (Varicad)
- Product management.
- We understand also physics. For firmware development it is good to know how the nature works. We have partners in Estonia who do assembly of the devices in small and large series.

It is possible to help you in many ways:

- Consultancy – intellectual property of developed product will belong to our customer.
- Sales contract – we provide required amount of devices or license to customer with agreed price.

Our Contacts:

WWW: <http://www.tauria.eu>

E-MAIL: tauria@tauria.eu