Application Hints

Fail-Safe CAN / LIN System Basis Chips

UJA1061, UJA1065, UJA1066, UJA1069

Version 2.9

Authors:

Thorsten Kauder, Matthias Muth, Martin Wagner

Date: 22nd of September 2010

founded by Philips

Revision History

Date	Version	Remarks
14.03.2003	V 1.0	Initial Version
18.11.2003	V 1.1	Updated
01.01.2006	V 2.0	Completely revised version according to the final SBC functionality
10.03.2006	V 2.1	• UJA1062 deleted from document. This derivative will not be developed • Figures 1-2 to 1-5 corrected. INH/LIMP connected to BAT14 instead of BAT42 • Sections 2.5 and 2.10 updated • Information about internal circuitry added in Pictures 2-10, 2-12, 2-13, 2-15, 2-18 and 2-21 • A.2 Design Checklist – Three new items added H13-H15
08.06.2006	V 2.2	• Corrected Document Header • FAQ 7, 8 and 9 added • Figure 1-2 and Section 1.1.1 warning added concerning SENSE in UJA1061. • Several references to 1.1.1 concerning SENSE in UJA1061 • Section 2.2 and 2.3 updated • Figure 2-2 updated • Design Checklist – Item H02a added
23.06.2006	V 2.2.1	• Document footer added/repaired • Appendix 3 updated for Adobe™ Reader 7.0
1.12.2006	V 2.3	• LIN Ferrite removed • Footnotes Corrected • Appendix B – References Updated • Figure 2.5 Corrected • 6.5.4 Revised • Comment on unused bytes in A1 • Software Design issue S02 changed • Added information in 4.5 • 5.8. References added
11.10.2007	V 2.4	• 4.1.2 Footnote added • A.6 FAQ updated
08.02.2008	V 2.5	• Table A-2 updated • A.4.1 updated
22.05.2008	V 2.6	• Figure 2-11: Figure updated • Chapter 4.6 inserted • Chapter 5.4: Remark about UJA1066 inserted • Chapter 6.5.2: Footnote 6 added
13.08.2008	V 2.7	• Section 5.5.3.4 updated (behavior for V1CMC = 0)
04.05.2010	V 2.8	• Section 2.7, correction of LIN supply voltage • Section 3; note about NC pins added • Section 5.4.1, equations for minimum and maximum trigger point updated • References on 3.0V version and 3.3V version for UJA1069 removed • Section A.7 inserted to clarify Device Identification Codes • References updated
22.09.2010	V 2.9	• Section 9, bypassing of interrupt limitation feature

Preface

Compared to the previous version 1, these application hints version 2 have been updated according to the final silicon behavior of the fail-safe SBC family UJA106x. During the evolution of the SBC family some important optimizations of the fail-safe system approach were made resulting in some changes in the software and hardware interface of this SBC family.

This new release should help all hardware and software designers building safe and reliable systems based on our fail-safe System Basis Chip family. Especially the checklists within the appendix are very useful to walk through the individual application hardware and software in order to make sure that the main application cases are covered by the system.

In case of any questions or suggestions for improvements of this document, please feel free to contact us.

The authors.

Table of Contents:

1. Overview

The UJA106x family is designed to offer standardized, fail-safe System Basis Chips (SBCs) for any kind of automotive body multiplexing application that uses CAN and/or LIN communication.

The UJA106x family members combine one or more transceivers, voltage regulators and system functionality that is explained in this document. The following table gives an overview of the UJA106x family members and the included physical layers.

Table 1-1 UJA106x derivatives

Derivative	LIN	HS-CAN	FT-CAN
UJA1061	X		X
UJA1065	X	X	
UJA1066		X	
UJA1069	X		

All devices are pin and function compatible and only differ in which physical layers are integrated.

All SBC's of this family are as identical as possible in order to allow an easy migration from one SBC to another in one and the same application. Thus, the family approach ensures that even a late change from one Physical Layer to another can easily be performed.

The SBC family is already prepared to operate within 42V environments but also allows operation in conventional 14V and 28V environments without carrying overhead for the included 42V feature.

Within this report, application information is offered supporting ECU developers with respect to hardware and software design issues. Following figure roughly shows, how the UJA106x is embedded in a typical ECU.

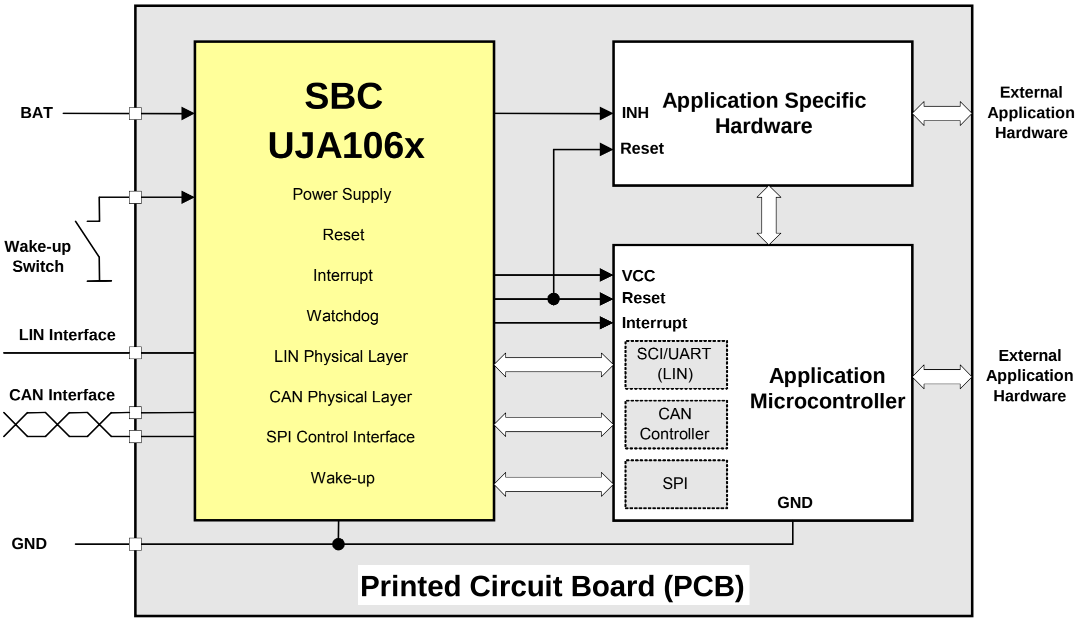


Figure 1-1: General Application Environment of the UJA106x

Integration of the UJA106x into an application is quite easy even though the high level of details in the data sheet might give a different impression. There are several control bits, dealing with general configuration, that only need to be configured to the used application hardware once. After initial configuration of the hardware only a few bits are used during operation of the system.

The main complexity of the UJA106x deals with fail-safe mechanisms, which the user does not need to take care of. This is the reason for many of the state transitions and conditions in the state diagram of the SBC. This document shows that from the user point of view the SBC is quite simple.

1.1. Highlights of the UJA106x SBC family

During development of the System Basis Chip (SBC) UJA106x, special attention was paid to system fail-safe features. Integrating the UJA106x into an ECU should guarantee a system behavior, which very safely prevents erroneous system states with continuous power consumption that would discharge the battery unintentionally.

In addition to the fail-safe system behavior, the UJA106x robustness is already prepared for the future 42V supply environment and offers solutions for power supplying via external DC/DC converters.

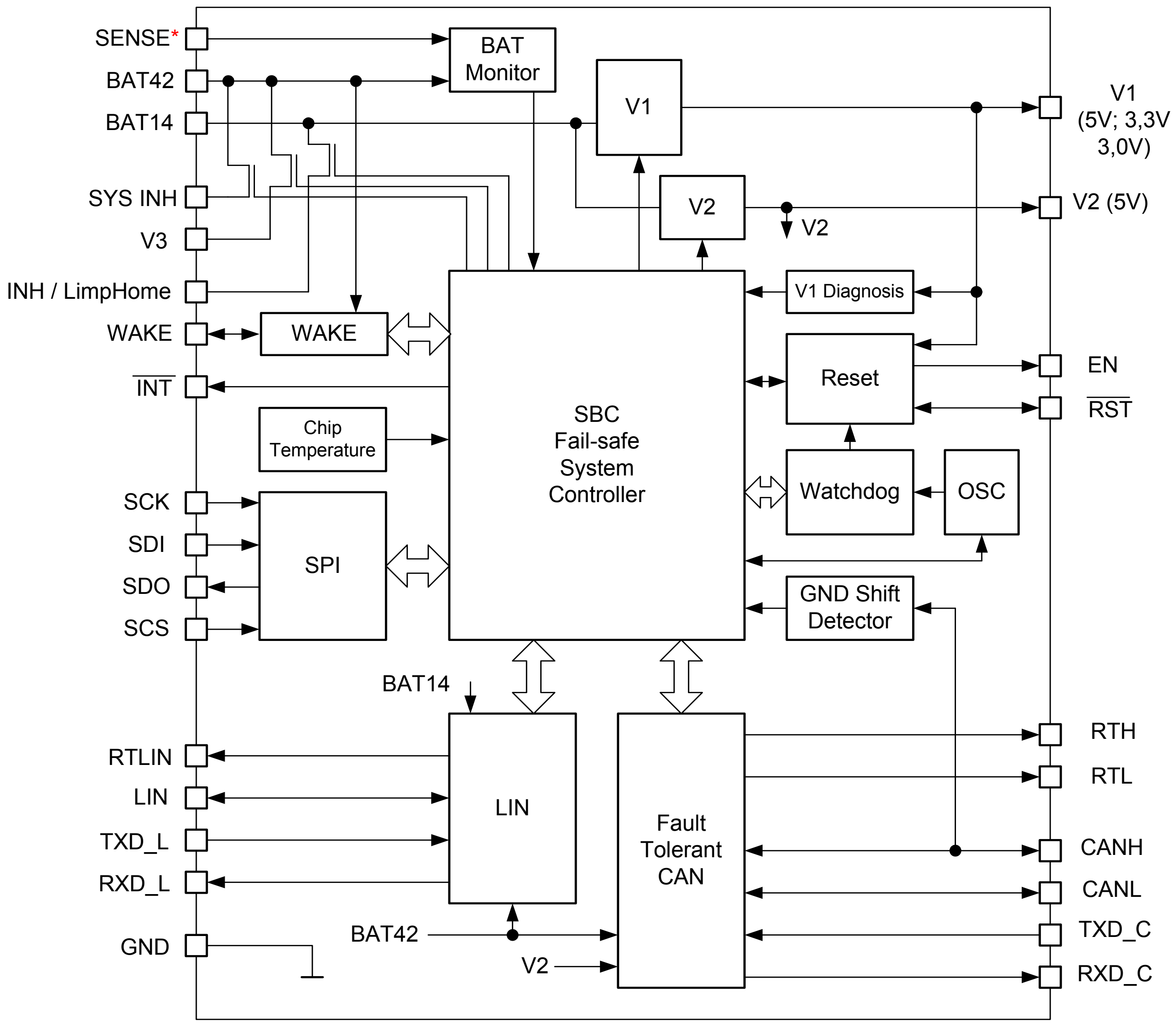


Figure 1-2: Block Diagram of the UJA1061

* Attention! The SENSE pin must not be used with the UJA1061. Please read chapter 1.1.1

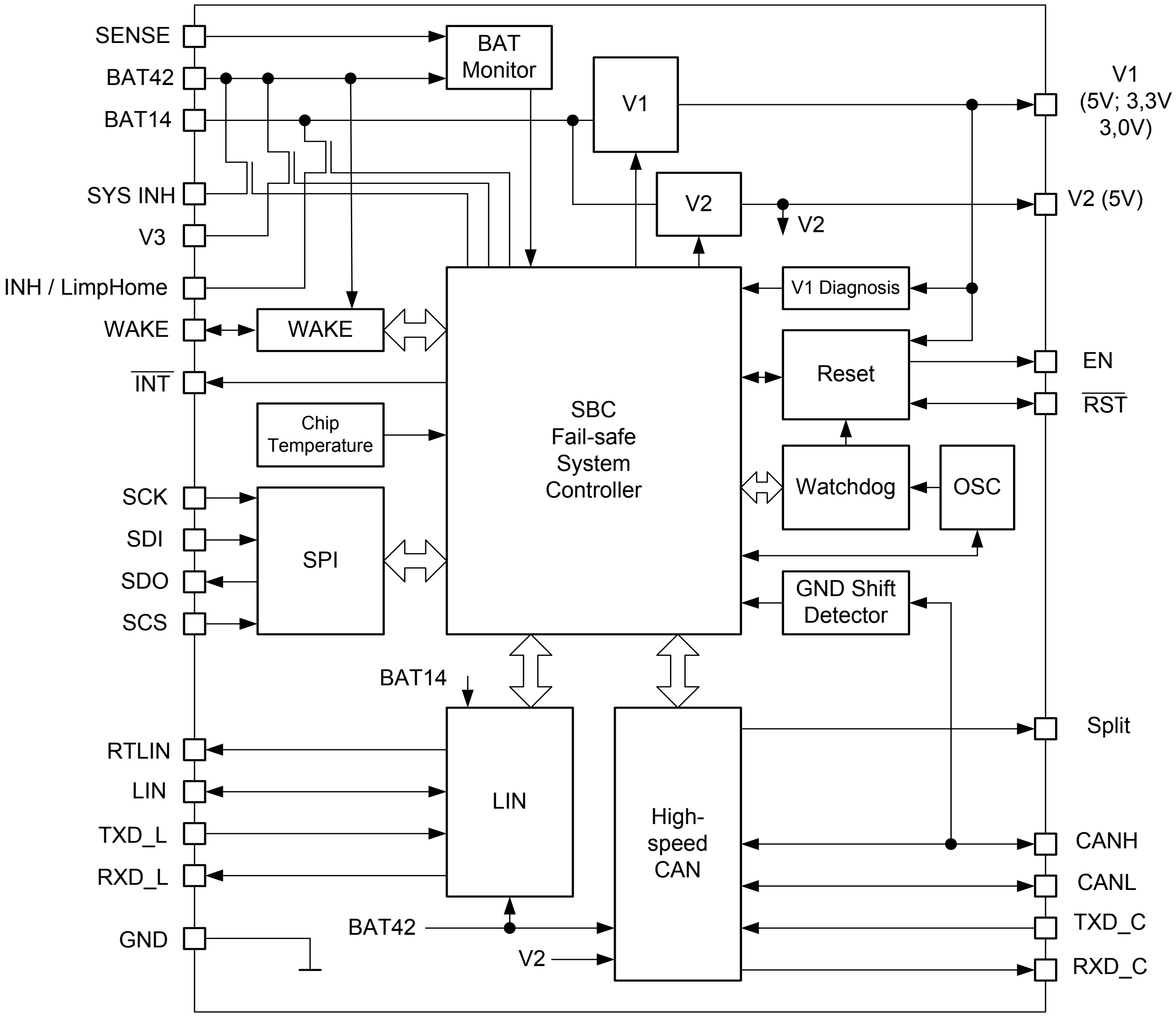


Figure 1-3: Block Diagram of the UJA1065 (without LIN: UJA1066)

The block diagram of the UJA1069 can simply be derived from the above shown diagrams by removing the CAN transceiver and the V2 supply, which is dedicated to the CAN transceiver cells. The CAN pins as well as the V2 pin are not connected within the UJA1069 (32-pin package). Optionally the UJA1069 is available in a 24-pin package and thus, with less non-connected pins. According to the family approach and PCB re-use, all SBC's are available in the same 32-pin HTSSOP package.

1.1.1. Power Supply System

The UJA106x can be supplied directly out of conventional 14V based systems as well as 28V systems. With the high robustness of up to 60V it is prepared for future 42V systems. Separate supply inputs (BAT14 / BAT42) allow easy integration of an external DC/DC converter that can easily be controlled by the SBC via a dedicated output pin (SYSINH) to achieve lowest system power consumption while the system is parked.

A chattering battery contact can optionally be detected with the dedicated input pin SENSE. This allows fast detection of supply problems and thus, allows to safe critical system data into non-volatile memory before the system supply goes down.

Attention: In the UJA1061 correct function of pin SENSE cannot be guaranteed under all environmental conditions. Therefore it must not be used and should be connected to GND.

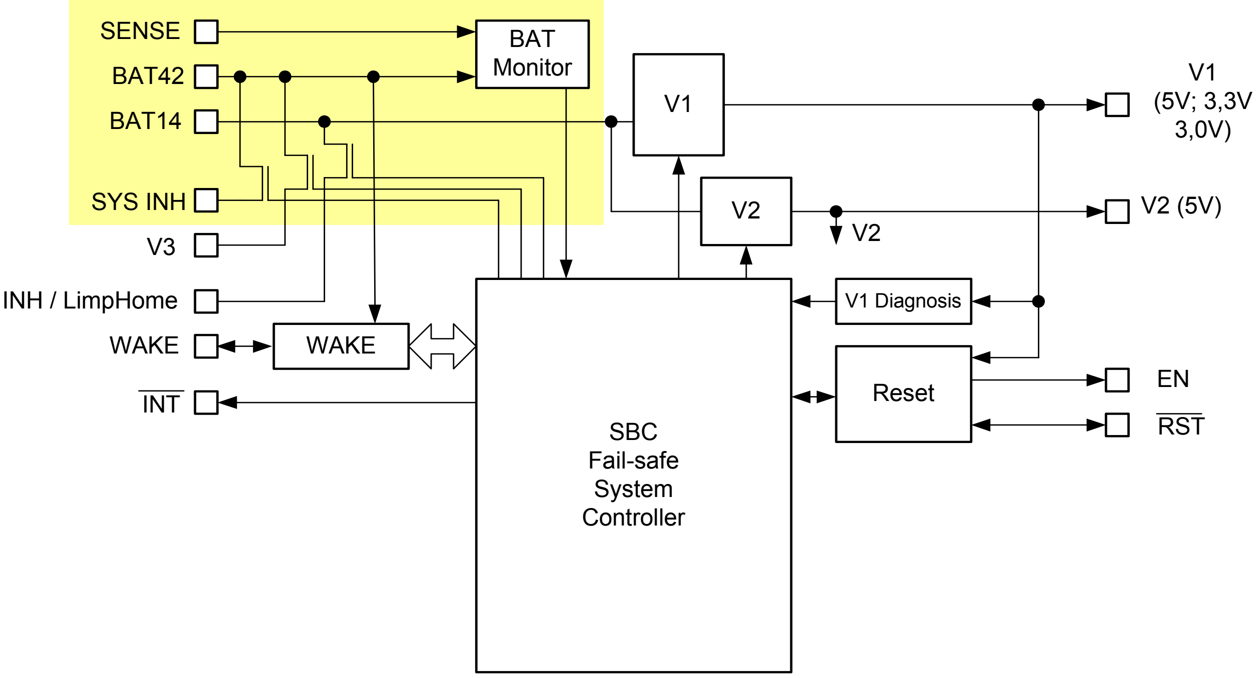


Figure 1-4: Power-supply of the UJA106x

1.1.2. Voltage Regulators

Two independent voltage supplies are integrated within the UJA106x. The main regulator V1 offers the supply voltage for the application microcontroller while V2 is reserved for the CAN Physical Layer.

The maximum output current at V1 depends on the input voltage applied to the power supply pin BAT14. A reduction of the input voltage below a certain threshold automatically increases the maximum output current at V1.

V2 is a dedicated 5V supply for the integrated CAN Physical Layer. The V2 output pin should be used for buffering this supply. In applications that do not use the CAN Physical Layer or in operation modes in which the CAN Physical Layer is not used, V2 can be used as an additional voltage supply for application Hardware.

In case more voltage supplies are needed for the application, the SBC can be extended with external regulators via the INH pin. This INH output of the SBC is included in the fail-safe approach and is disabled in case of unstable system behavior.

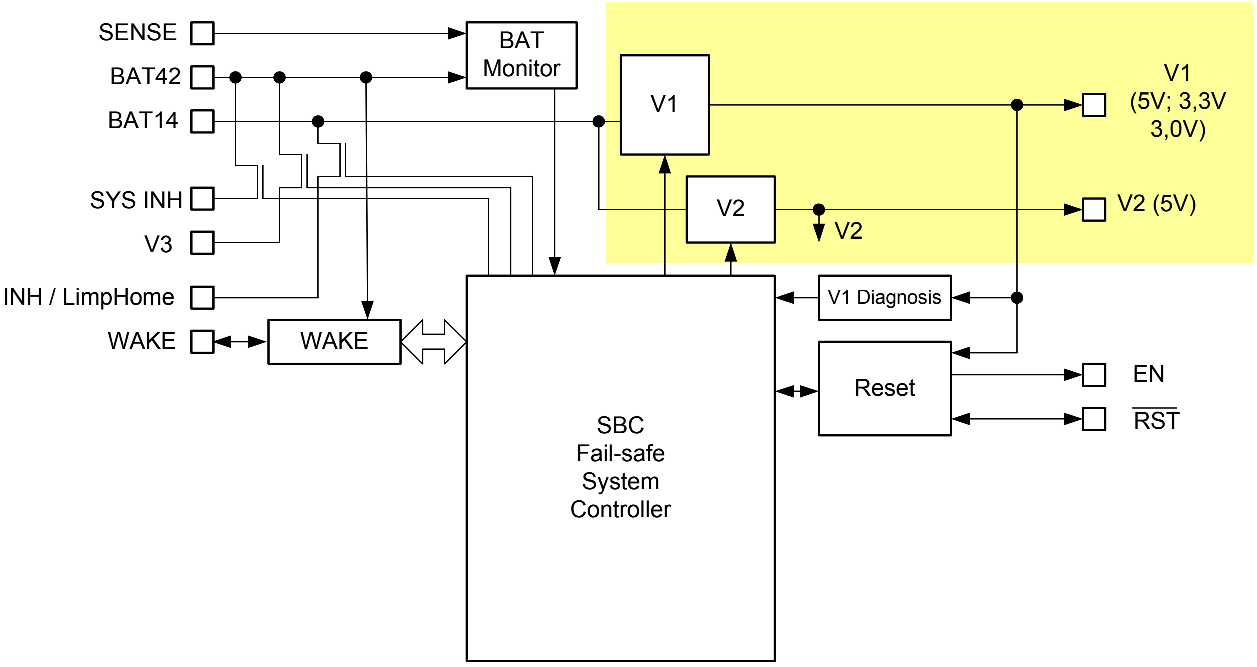


Figure 1-5: Voltage Regulators of the UJA106x

1.1.3. Physical Layers, CAN & LIN

The UJA1061 includes the TJA1054/55 core functionality and thus is 100% compliant to the ISO 11898-3 standard for fault-tolerant CAN networks. It is upgraded with respect to increased EMC / ESD and 42V requirements within the automotive market.

The UJA1065/66 includes an upgraded TJA1040/41 core function and thus it is 100% compliant to ISO 11898-2 and ISO 11898-5 High-speed CAN standard. It is upgraded with respect to EMC / ESD and 42V system requirements.

From system point of view it is very important that the CAN physical layer is active in all ECU's. Availability of the UJA106x's CAN Physical Layer is guaranteed because it operates autonomously without the need of a running microcontroller. This is different from Stand-alone Transceivers, which need to be pulled into "Normal Mode" by the microcontroller first.

A dedicated "Selective Sleep Bit" features "CAN Partial Networking" scenarios with the UJA106x where parts of the CAN network may be shut down, while others are still communicating.

Besides the CAN physical layer, an enhanced, LIN 2.0 compatible transceiver block is included in the UJA1061/65 and UJA1069. The slope control function is upgraded according to the LIN2.0 specification and thus supports the full range of LIN applications up to 2% oscillator tolerance at 20kBit/s. A dedicated low-slope mode allows further reduction of emissions in case lower bit rates are used (10.4kBit/s according to SAE-J2602).

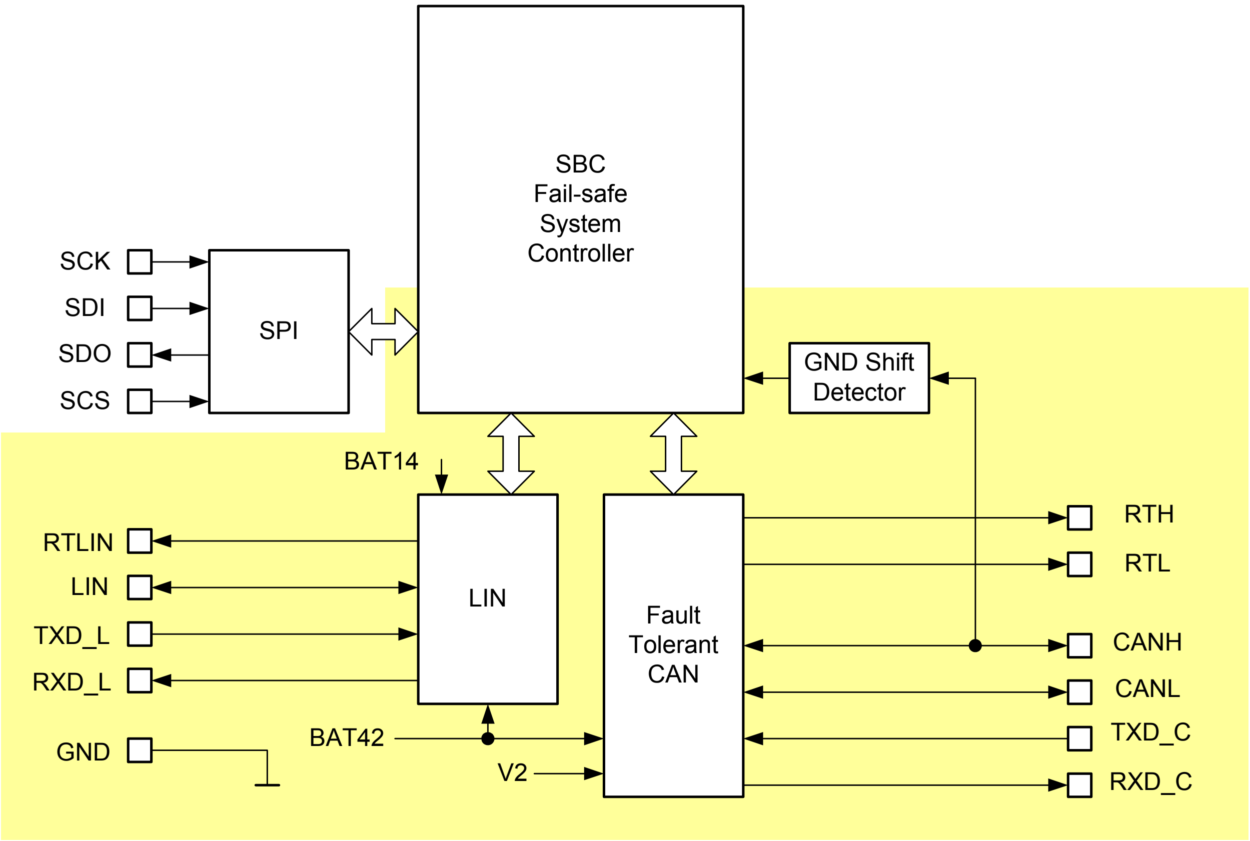


Figure 1-6: FT-CAN and LIN Physical Layer within the UJA1061

1.1.4. System Safety Features

An integrated Window Watchdog provides supervision of the application microcontroller. An independent on-chip oscillator within the SBC is used as clock reference. Even this internal oscillator is observed by the SBC and forces a fail-safe low-power system state in case of oscillator failure. The oscillator offers maximum reliability because it does not require any external components.

Several monitoring functions of the SBC prevent deadlocks with high current consumptions. The SBC system reset line safely resets all external hardware upon serious system conditions. Besides resetting external hardware this line is observed by the SBC for clamping and open-wire conditions and the system is forced into a safe condition whenever the system reset signal does not operate correctly.

The UJA106x can detect poor GND connections of an ECU. Two different detection levels can be programmed to provide a warning to the application if the GND connection is getting worse.

A special flag provides supervision of the microcontroller supply voltage. It is set whenever the microcontroller supply V1 has fallen below a certain threshold before a hardware reset is forced.

The SBC detects if its interrupts are not served within a reasonable time. This ensures that the application software is running correctly and that the interrupt line is intact.

The UJA106x provides a global enable signal for safety critical hardware (EN Pin). This output is integrated in the fail-safe system and can only be activated if the application is running correctly, triggering the watchdog in a correct way. Whenever the system gets instable, the EN signal is immediately set LOW to shut down critical external hardware.

The UJA106x is accessed via a 16-bit SPI interface. Any critical access is coded in a fail-safe way using redundant bits. In addition to redundancy, the clock cycle count is monitored and any erroneous access is denied.

Two special registers within the SBC are designed to support detection of cyclic software failures like damaged FLASH Memory cells. The UJA106x is the first device, which supports detection of such system failures using a failure statistics function.

The chip temperature of the UJA106x is monitored and a warning is offered in case it exceeds a certain limit. This allows the application to reduce the load.

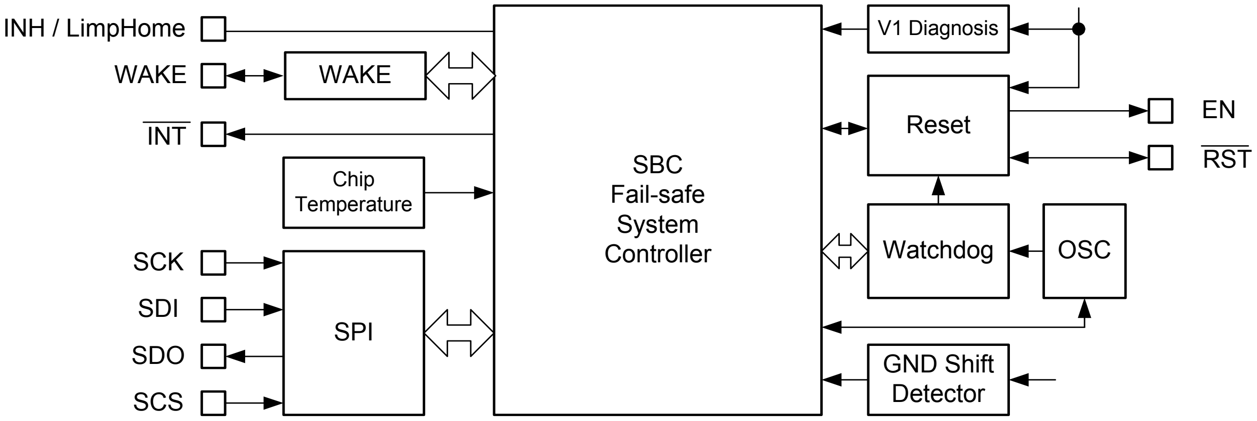


Figure 1-7: Overview of fail-safe features within the UJA106x

1.1.5. Low-power Support

The UJA106x provides two levels of Low-power operation.

In Standby Mode, the application microcontroller is still supplied but may be set into a low-current mode (Stop Mode of the micro). A fail-safe method allows disabling the watchdog during Standby Mode based on a V1 supply current monitoring feature. The watchdog can only be disabled if the system supply current of the microcontroller is below a certain threshold.

Within Sleep Mode, the application microcontroller supply is shut down completely. The only supplied device in the entire ECU is the UJA106x monitoring the interfaces for a wake-up condition.

Waking up an ECU is possible via CAN traffic, LIN traffic, falling edges on the dedicated wake-up port or a power-on condition at the battery inputs. Furthermore it is possible to program the SBC’s watchdog supporting cyclic wake-up events. This would autonomously wake-up the ECU from time to time with a programmable period even out of Sleep Mode with un-powered microcontroller.

In case external wake-up switches are used, the V3 output pin can be used to cyclically bias the external switch with some current towards BAT42. This allows minimizing the bias current in the switch in case of a clamped contact. The wake-up port of the SBC is of course synchronized with the cycle time of V3.

1.1.6. Flash Memory Support

A special operating mode of the UJA106x is implemented in order to support FLASH memory reprogramming within an ECU. This mode has a secure entry mechanism and provides a defined system reset upon entry and exit. Within the so-called Flash Mode, the time consuming flashing is still supervised by the Watchdog but the SBC’s Watchdog can be triggered with relaxed timings.

1.1.7. Software Development Support

Software development is supported through a dedicated mode. Software Development Mode can be entered by either setting a special control bit after first battery connection or by supplying a certain voltage at pin Test before first battery contact. In this mode the watchdog does not have to be triggered and interrupts do not have to be served. I.e. wrong watchdog triggering and not served interrupts do not result in resets. This allows hardware debugging with breakpoints.

1.1.8. Packaging

The UJA106x comes in a very small outline power package, which allows the design of very smart printed circuit boards. A heat sink on the back of the HTSSOP32 Package makes sure that the SBC gets rid of the internal power dissipation. Following figure shows a comparison of the well known SO14 Package of standard fault-tolerant transceivers compared to the UJA106x package including CAN / LIN / Power Supply / Watchdog and more.

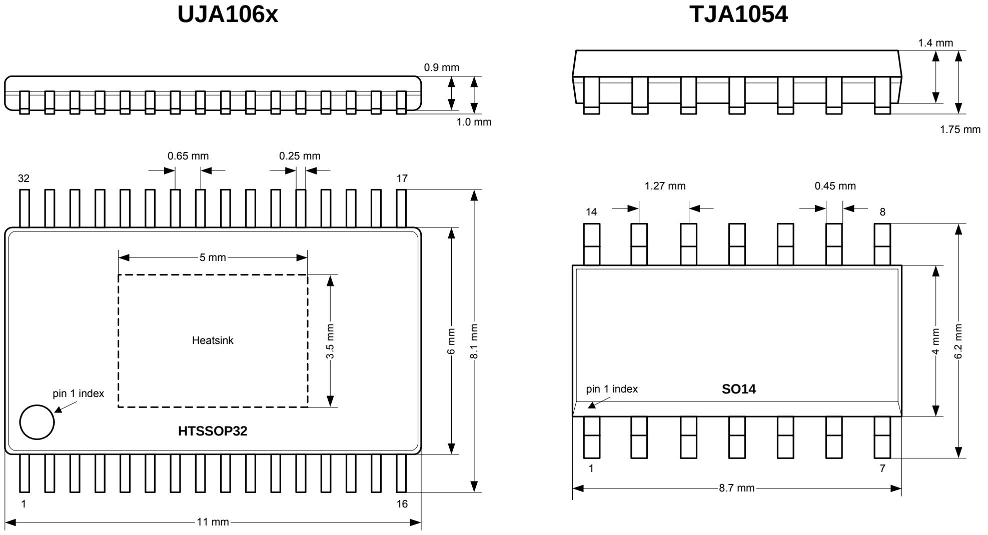


Figure 1-8: Comparison of Packages, UJA106x vs. TJA1054

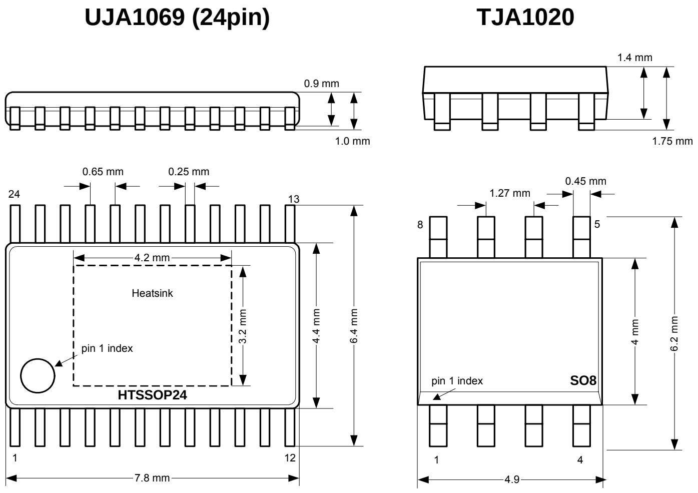


Figure 1-9: Comparison of Packages, UJA1069 (24pin) vs. TJA1020

2. Hardware Design

In this chapter typical application circuitry for the UJA106x is shown in detail. For detailed characteristics please refer to the respective Datasheet.

2.1. 14V ECU Supply Environment

In conventional 14V supply systems the two supply input pins of the UJA106x may be short-circuited and connected to the battery supply via the ECU polarity protection diode. Some buffering should be applied to suppress voltage spikes, drops and noise.

Optionally the SENSE[1] pin can be connected to the ECU battery supply pin to support the battery contact-monitoring feature of the UJA106x as shown in Figure 2-1. If pin SENSE is used, it has to be connected to the Battery supply via a separate polarity protection diode and a 2kΩ series resistor. In case SENSE is not used it is recommended to connect this pin to GND. In 14V environments the SYSINH output pin is typically not used and can therefore be left open.

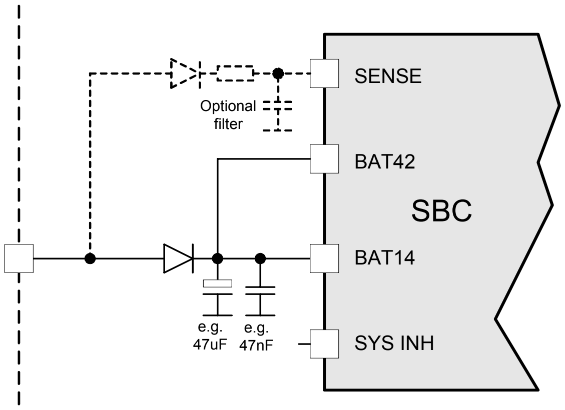


Figure 2-1: Typical Application 14V environment

2.2. 28V/42V ECU Supply Environment

In case of a 28V or 42V-supplied ECU, an external DC/DC converter is recommended to step-down the BAT-supply to a suitable voltage. Since the CAN physical layer operates on a 5V basis, the DC/DC converter should provide approximately 6V to the BAT14 pin of the UJA106x.

The SYSINH output of the SBC can be used to control the external DC/DC converter. Whenever V1 or V2 needs to be active the DC/DC controller is enabled with an active HIGH signal. As soon as the system enters a low-power condition the SYSINH pin becomes floating, thereby shutting down the external DC/DC stage to minimize the current consumption of the entire system. Depending on the control input of the DC/DC converter, a pull-down resistor may be required.

To guarantee safe start-up and shutdown behavior, BAT14 and BAT42 have to be applied with separate protection diodes and an additional diode between them (see Figure 2-2).

Optionally the SENSE[1] pin can be connected to the ECU battery supply pin to support the battery contact-monitoring feature of the UJA106x as shown in Figure 2-2. If pin SENSE is used, it has to be connected to the Battery supply via a separate polarity protection diode and a 2kΩ series resistor. In case SENSE is not used it is recommended to connect this pin to GND.

[1] The SENSE pin must not be used with the UJA1061. Please see 1.1.1

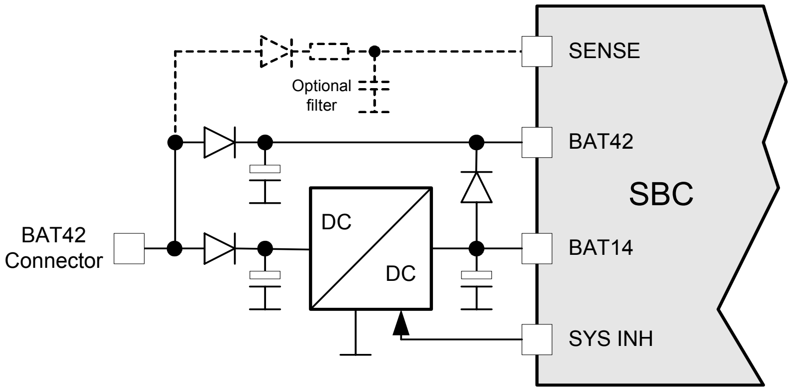


Figure 2-2: Typical Application 42V Environment

Hint: The SYSINH can also be used to disconnect the buffer capacitors applied to the BAT14 input of the SBC, while the system is in Sleep Mode. Instead of using the DC/DC converter within 14V applications, a simple SYSINH-controlled switching transistor in front of the BAT14 pin could be used getting rid of the leakage current of big buffering capacitors at BAT14.

2.3. Test Pulse 4 and 4b

One Test pulse that carmakers require ECUs to withstand is test pulse 4 defined in ISO 7637 that simulates supply voltage during engine start. The pulse form with its worst-case values is depicted in Figure 2-3. With a common diode for BAT 14 and BAT 42, as suggested in 2.1, the voltage at BAT42 would fall below the Power-On threshold of 4.25V…5.0V and a Reset would be forced.

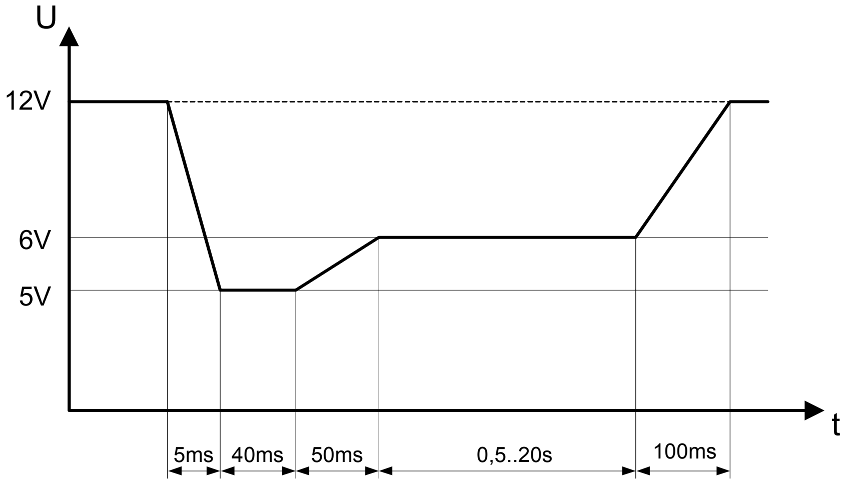


Figure 2-3: Test pulse 4 in a 12V system

To withstand this pulse without resetting the microcontroller some measures can be taken. Schottky diodes can reduce the diode drop to typically 0,3V. If a 5V microcontroller is used the V1 under voltage reset threshold should be set to 70%. BAT14 and BAT42 should be supplied via separate diodes and buffered independently with sufficient capacitors (BAT42 at least47uF) as shown in Figure 2-5. The relatively small current running into BAT42 can then be supplied by the capacitor until the supply voltage returns to its 6V value (Figure 2-3).

A variation of the test pulse 4 is the 4b pulse as introduced by VW (TL 820 66 1997–05) and depicted in Figure 2-4. Here the supply voltage falls below the power-on threshold for more than one second and even below 3 Volts for some tens of milliseconds.

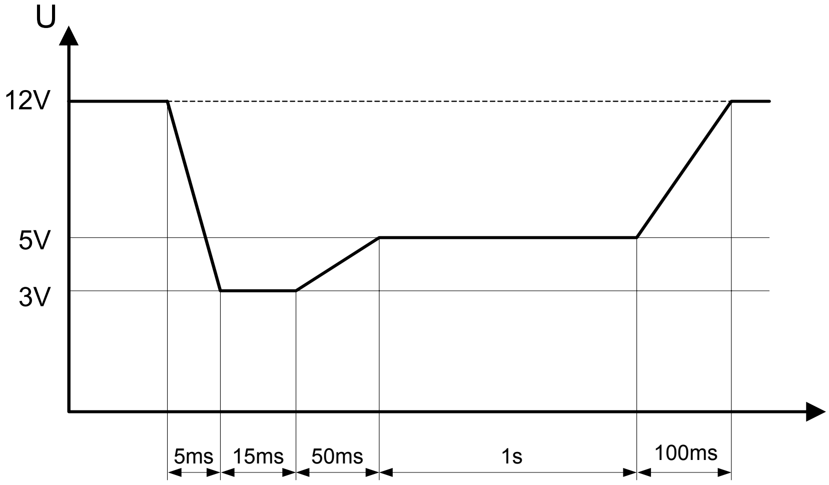


Figure 2-4: 4b pulse in a 12V system

Even this pulse can be withstood with a 5V microcontroller provided that it works at voltages down to 3.5V. As proposed above, two separate diodes have to be used for BAT14 and BAT42 and both pins have to be buffered separately, the V1 under-voltage threshold has to be set to 70%. Even with a non-Schottky diode the overall drop at BAT14 is low enough for V1 to stay above the 70% under-voltage threshold. It is recommended to deactivate both the CAN and the LIN transmitter to minimize current consumption as soon as the low supply voltage is detected. The buffer capacitor at BAT42 has to be sufficiently big to supply the SBC for just over one second.

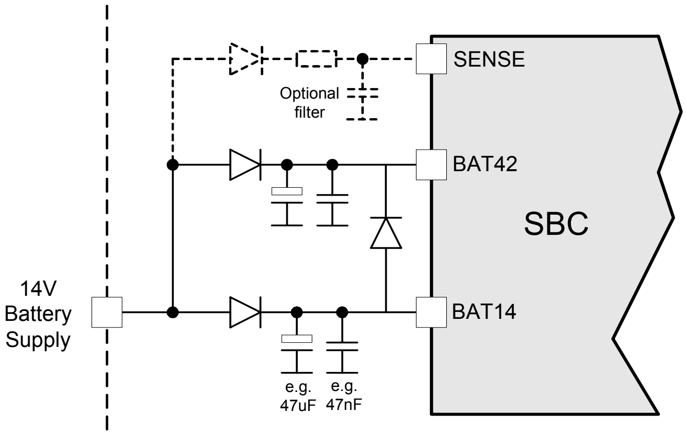


Figure 2-5: Application to withstand test pulses 4 and 4b

To guarantee correct operation of the V1 voltage regulator, the digital supply of the SBC always has to be active whenever V1 ramps up. Therefore it is highly recommended to apply an extra diode between BAT14 and BAT42 in case these two supplies are connected to the BAT supply of an ECU via separated polarity protection diodes (as shown in Figure 2-5).

2.4. Application Voltage Regulators

The UJA106x offers two linear voltage regulators V1 and V2, which are both supplied out of the BAT14 pin. The voltage output V1 is dedicated for the main application microcontroller, V2 is internally applied to the integrated CAN physical layer.

In addition to these two integrated voltage regulators the SBC supports a flexible system extension via an INH output pin. This pin is fully accessible from the application micro controller via the SPI interface and is integrated into the fail-safe system approach of the UJA106x. Thus, whenever extra supplies are required for the application, it is highly recommended to control them with the INH output feature of the UJA106x. The INH output is automatically disabled by the SBC in case Fail Safe Mode is entered, e.g. if the watchdog of the SBC is not triggered as required.

2.4.1. Voltage Regulator V1

V1 is the main voltage regulator of the Application and is fully integrated into the fail-safe system of the UJA106x. In case of serious ECU problems, V1 will automatically be shut down, thus making sure that the application does not consume current unintentionally.

After power-up or a wake-up event V1 becomes active. V1 is only inactive during Sleep- and Fail-safe Mode.

Depending on the application, some buffering capacitance has to be applied at the V1 output pin. The voltage regulator works stable even with very low ESR capacitors such as Tantals. All SBC's are available with 5V output voltage at V1. In addition the UJA1061, UJA1065 and UJA1066 are also available in a 3.3V version (different order codes, see data sheet).

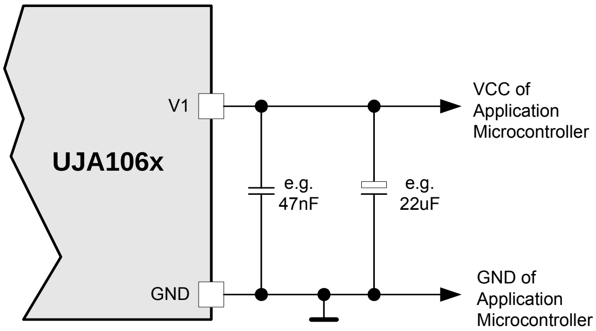


Figure 2-6: Typical V1 Application

2.4.2. Voltage Regulator V2

The voltage regulator V2 is dedicated for the integrated CAN physical layer. This ensures that the noisy CPU environment (V1) is electrically separated from the CAN interface. V2 is switched on and off automatically by the SBC fail-safe system. This guarantees 100% CAN bus support even if the microcontroller of the ECU is down due to some hardware failure or damage. The external V2 pin is intended to apply some external buffer capacitance for the CAN transceiver supply as shown in Figure 2-7.

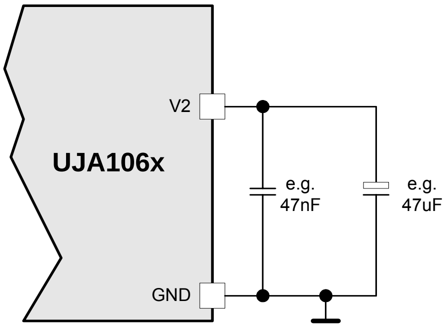


Figure 2-7: Typical V2 Application

In order to achieve optimum EMC and bus performance of the CAN interface it is normally not recommended to use V2 for any other external supply purposes. However if the CAN bus is not used in an application or if it can be guaranteed that it is not used at certain times V2 can be used to supply external hardware. V2 can be turned on continuously, regardless of the CAN transceiver state using a dedicated control bit.

Remark: V2 should only be used to supply other hardware blocks than the SBC's CAN block if CAN is definitely not used (e.g. when in Off-line Mode or if not connected to the CAN bus).

2.4.3. Further Voltage Regulators, System Extension, Limp Home Feature

The UJA106x is designed to offer a flexible extension with external voltage regulators using an INH output pin. This INH pin allows the integration of an extra voltage regulator into the fail-safe system of the UJA106x, if required by the application. The INH signal is accessible via the SPI Interface of the SBC and can be activated or deactivated as desired as long as the SBC's watchdog is triggered correctly. The external voltage regulator can also stay active if V1 is turned off and the system has successfully entered Sleep Mode.

Whenever the system is forced into Fail-safe Mode, because e.g. an ECU hardware or software failure occurred, the INH output is set LOW in order to disable all external supplies. This is to guarantee lowest power consumption of the system and to achieve maximum fail-safe behavior in case of serious application problems.

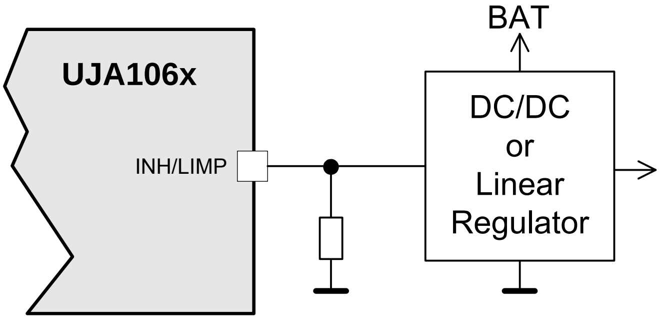


Figure 2-8: Inhibit Application

Depending on the control input of the external supply, a pull-down resistor may be required as depicted in Figure 2-8.

The second possible application of this pin is as a limp home output. This way it can be used to activate "emergency" hardware whenever the SBC enters Fail-safe Mode. Regardless of the configuration the Limp Home pin is set to LOW on entering Fail-safe Mode. A simple circuitry as shown in Figure 2-9 can then be used to provide a BAT related HIGH signal to control an emergency Application.

The INH/LIMP pin can be switched on and off by software any time. After Power-on the INH/LIMP pin is always floating.

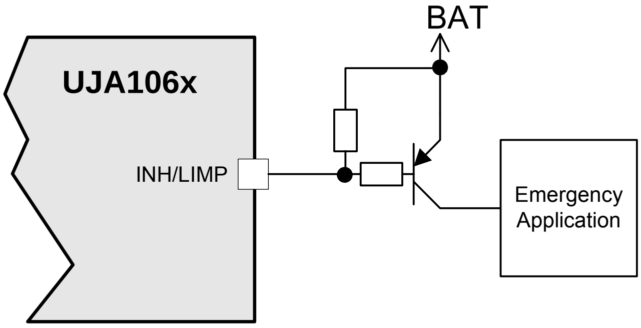


Figure 2-9: Limp Home Application

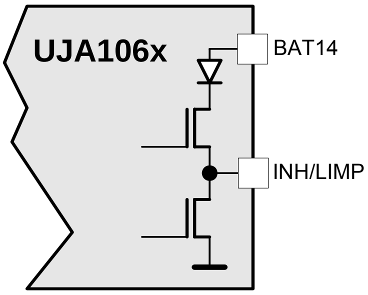


Figure 2-10: INH/LIMP Internal Circuitry

2.5. Microcontroller Interface

The application host microcontroller has to be supplied out of V1 as described in the previous paragraph. The SBC offers the system Reset signal via the active LOW pin RSTN upon power-up or any condition that requires a system reset. The pin RSTN does not require any external components for signal conditioning. However an external pull-up to V1 can be applied, if required by the application. A small capacitor (e.g. 47 pF) can eliminate potential cross-talk from other signals of the PCB.

The reset pulse duration can be programmed to be short or long depending on the application requirements. For fail -safe reasons, the long Reset length is automatically set in restart or Fail-safe Mode.

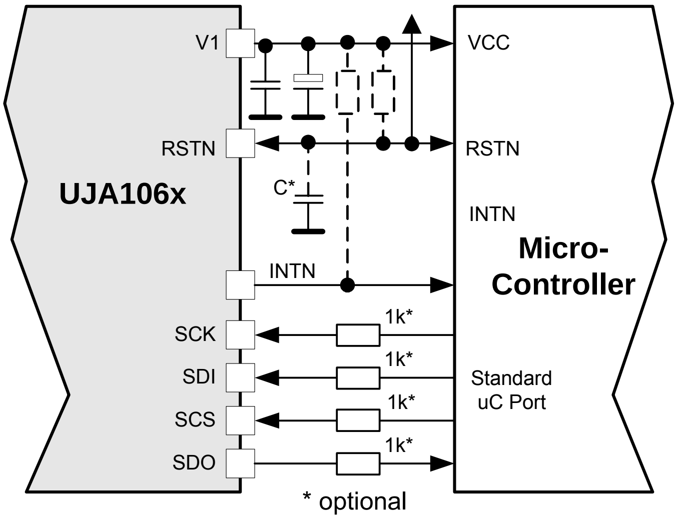


Figure 2-11: Typical Microcontroller Interface

Main control of the SBC is handled via the 4-wire SPI interface. Typical microcontroller ports can be used to drive this interface with 16-bit SPI commands. For decoupling reasons 1kΩ series-resistors can be inserted in all SPI lines.

It should be noted that the SCS pin of the fail-safe SBC should always be on a defined level. Any HIGH-to-LOW transition is interpreted as the start of an SPI communication cycle. Therefore it has to be guaranteed that SCS does not unintentionally change during operation. Background: When the reset pin is down the pins of many microcontrollers are floating. If the SCS pin is not well defined at this moment a chattering on SCS might occur when the reset pin is released. This is interpreted as an SPI failure and results in another reset of the SBC. The SCS pin of the SBC contains an internal pull-up resistor to V1 with a typical value of 130k. If necessary within a certain application this internal pull-up should be strengthened with an external resistor.

The open-drain interrupt signal line INTN is used to forward system interrupts to the host microcontroller. All interrupt sources of the SBC can separately be enabled. It is guaranteed that the INTN line stays cleared (floating) for at least 2µs after the interrupt register was read out in order to support edge-triggered microcontroller interrupt input pins.

If no other components provide a pull-up resistor at the interrupt line, a pull-up resistor (e.g.2kΩ) towards V1 has to be provided.

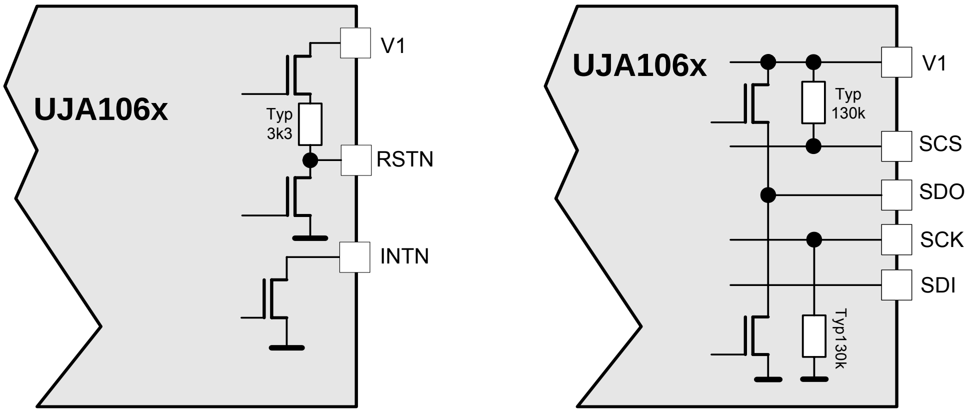


Figure 2-12: Microcontroller Interface Internal Circuitry

The output voltage levels as well as the input voltage thresholds of all microcontroller related pins are fully adapted to V1. Thus, if a 3.3V derivative is used, the complete interfacing between the SBC and the microcontroller operates on a 3.3V basis.

2.6. CAN Transceiver Interface

The CAN Transceiver interface of the UJA1061 is compatible to the ISO11898-3 fault-tolerant CAN standard. The UJA1065/66 includes a High-speed CAN block according to the ISO11898-2 and ISO11898-5 standards.

The CAN Transceiver is supplied by V2 which has to be buffered to achieve optimum EMC and system behavior. For details, please refer to 2.4.2

As known from stand alone transceivers the TXDC and RXDC wires are used for the serial communication between the CAN protocol controller and the UJA106x. If the CAN protocol controller generates unacceptable noise, these connections can optionally be applied with series resistors (about 1k).

The UJA106x is designed to withstand ESD pulses up to 8kV according to the human body model at the bus pins CANH, CANL, RTH, RTL and pin SPLIT and thus typically does not need further external measures. Nevertheless, if much higher protection is required, external clamping circuits can be applied to the CANH and CANL line. In Figure 2-14 and Figure 2-16 optional external ESD protection is realized with the PESD1CAN protection diode at CANH and CANL to GND.

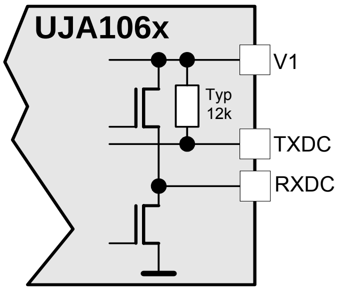


Figure 2-13: CAN Transceiver Interface Internal Circuitry

The clamping voltage of the protection devices should be above the maximum battery supply voltage of the system in order not to damage the devices, in case there is a short to battery on the bus lines. It is recommended to apply the ESD protection circuitry close to the connectors of the ECU as shown in Figure 2-14 and Figure 2-16.

A common mode choke provides high impedance for common mode signals and low impedance for differential data signals. Due to this, common mode signals, either produced by RF noise or by the transceiver itself, get effectively attenuated while passing the choke. In fact, a common mode choke can help to reduce emission and to improve immunity against common mode disturbances and ESD.

2.6.1. UJA1061 – Fault Tolerant CAN Transceiver

A dedicated SPI register that offers more detailed bus failure analysis replaces the former NERR pin of stand-alone transceivers. The EN output of the UJA1061 can optionally be configured to output a TJA1054 compatible active LOW error signal (see special modes register).

Within typical applications, the final required ECU termination resistance has to be applied to the dedicated RTL and RTH pins as known from the TJA1054 / TJA1055. Due to EMC considerations (emission) it is recommended to choose resistors with a matching of at least 1% between the RTH and RTL resistor within one ECU.

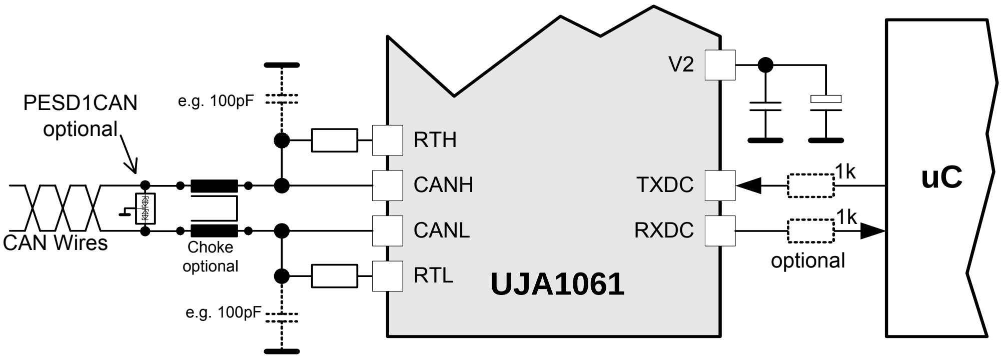


Figure 2-14: Typical Application of Fault-tolerant CAN Transceiver Block

2.6.2. UJA1065/66 – High Speed CAN Transceiver

The UJA1065/66 supports the internal termination of CANH and CANL to $V_2/2$ in operating mode and towards GND during low-power mode as known from the TJA1040/41 devices (see also ISO11898-5).

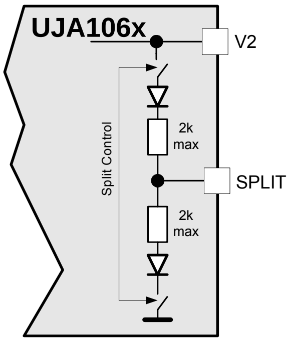


Figure 2-15: Split Pin Internal Circuitry

To further reduce EME by stabilizing the recessive bus level a split termination network can optionally be employed. Pin SPLIT provides an output voltage of $V_2/2$. By simply connecting the pin SPLIT to the centre tap of the Split Termination as shown in Figure 2-16, DC stabilization of the common mode voltage is achieved.

Especially in case of un-powered nodes leakage currents from the bus into the transceiver may force the common mode voltage to drop below $V_2/2$ during recessive state. The DC stabilization aims to oppose this degradation and thus helps improving the emission performance. In case of no significant leakage currents from the bus, the pin SPLIT can be left open. (It has to be checked with the carmaker whether this option is applicable or not)

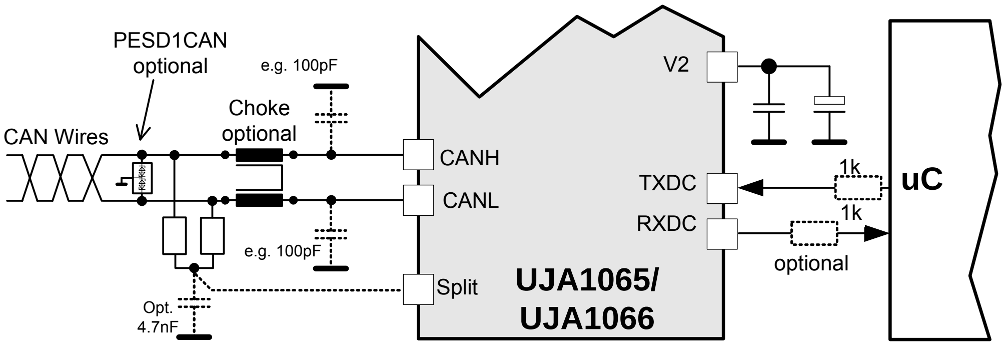


Figure 2-16: Typical Application of High-speed CAN Transceiver Block

2.7. LIN Transceiver Interface

The integrated LIN Transceiver Interface is directly supplied out of BAT42. The slope timings and propagation delay symmetry is adapted to the LIN 2.0 standard in order to allow maximum bit rate tolerance in the system while offering first-class EMC performance.

As known from the TJA1020, the TXDL and RXDL wires are used for the serial communication between the LIN protocol controller and the UJA106x. These connections can optionally be applied with series resistors (about 1k) for filtering.

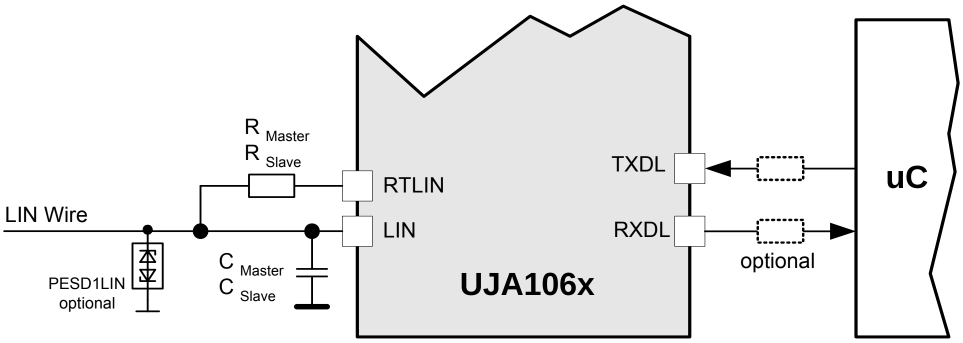


Figure 2-17: Typical Application of LIN Transceiver Block

Since the UJA106x typically operates as a LIN master node, no 30k slave termination resistor is integrated. Instead, a dedicated pin RTLIN allows connecting an external master or slave resistor of 1kΩ or 30kΩ respectively. The dedicated termination pin allows switching off the low-ohmic path towards BAT in case a short circuit between LIN and GND. This fail-safe feature ensures that the battery is not discharged because of a short circuit between LIN and GND.

To achieve optimum EMC performance the LIN pin should be applied with a 2.2nF capacitor (master) towards GND as described in the LIN standard. In slave applications it is recommended to at least apply a 150pF capacitor. Depending on carmakers requirements, the capacitance can be chosen differently.

The on-chip ESD protection of pin LIN of the UJA106x is designed to withstand Vesd(HBM) = ±8 kV according to the Human Body Model (HBM) JESD22-A114-B (100pF / 1.5 kΩ). External ESD protection on the LIN bus connection is recommended if the UJA106x is subjected to higher ESD-pulses. Figure 2-17 shows such an external ESD protection employing the ESD protection diode PESD1LIN. The Suppressor diode should be placed as close as possible to the connectors, whereas the LIN node capacitor $C_{MASTER/SLAVE}$ should be placed close to the LIN transceiver's bus pin.

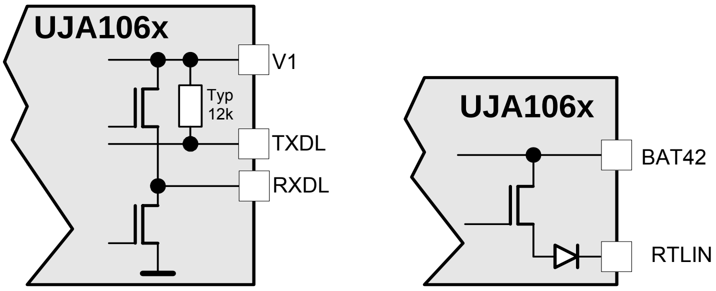


Figure 2-18: LIN Transceiver Internal Circuitry

The curve shaping of the LIN bus signal in Normal-slope Mode is optimized for the maximum specified LIN transmission speed of 20kBaud. Thus for low speed LIN applications (≤10.4kBit/s) the curve shaping in

Normal-slope Mode has unnecessary steep slopes. Therefore the UJA106x provides a special Low-slope Mode with reduced slopes (see Figure 2-19). These reduced slopes result in further reduction of EME.

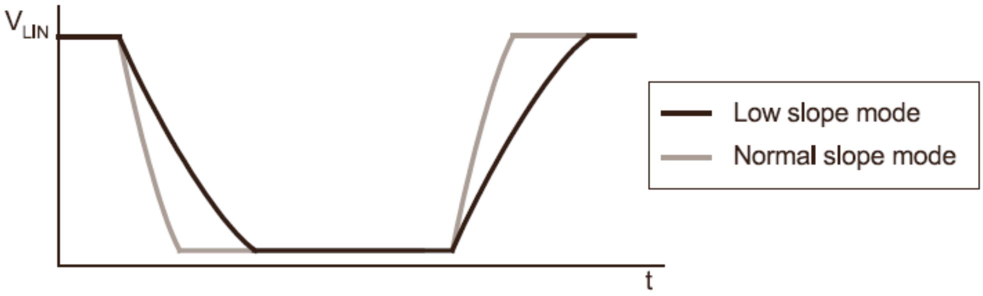


Figure 2-19: Reduced LIN bus slope in Low-slope Mode

Whenever the LIN Transceiver of the UJA106x is in Offline-Mode the RTLIN pin is driven with a 75µA current source (unless a bus short via GND is detected). If a stronger termination is required, the LIN master termination can be split as depicted in Figure 2-20, at the cost of a higher short-circuit current in case of a dominantly clamped LIN bus.

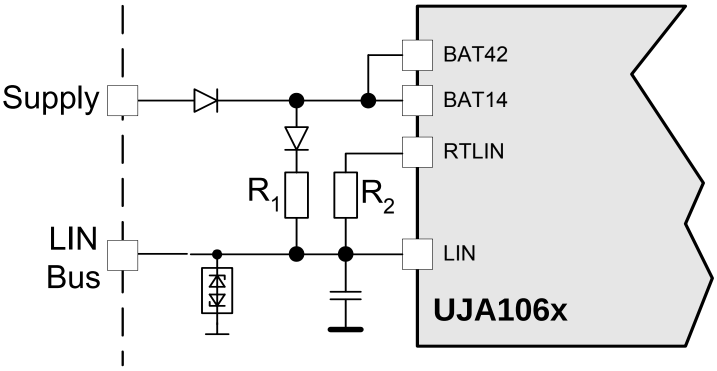


Figure 2-20: Master Termination Split

The Resistors R_1 and R_2 in parallel determine the master termination while the LIN transceiver is in Active-Mode. In Off-line-Mode the termination is mainly determined by R_1. Therefore the maximum LIN bus short-circuit current $I_{SC,max}$ can be trimmed by R_1 according to Equation 1.

$$R_1 = \frac{V_{BAT,\max}}{I_{SC,\max}} \ ; \ R_2 = \frac{R_1 \cdot R_{Master}}{R_1 - R_{Master}} \text{ with } R_{Master} = 1k\Omega$$

Equation 1: Split Termination Resistors

2.8. Local Wake-up Interface, V3 Output

The UJA106x offers one local dedicated wake-up port. In most applications this port is connected to an external wake-up switch to GND. The wake-up port can be configured to be sampled either continuously or cyclically with two different periods. The cyclic sampling is synchronized to the V3 output and offers significant current saving if switches are supplied via V3.

A dedicated edge capture register allows identifying the wake-up source and distinguishing between CAN, LIN and wake port triggered wake-ups.

The V3 output can alternatively be used to drive loads.

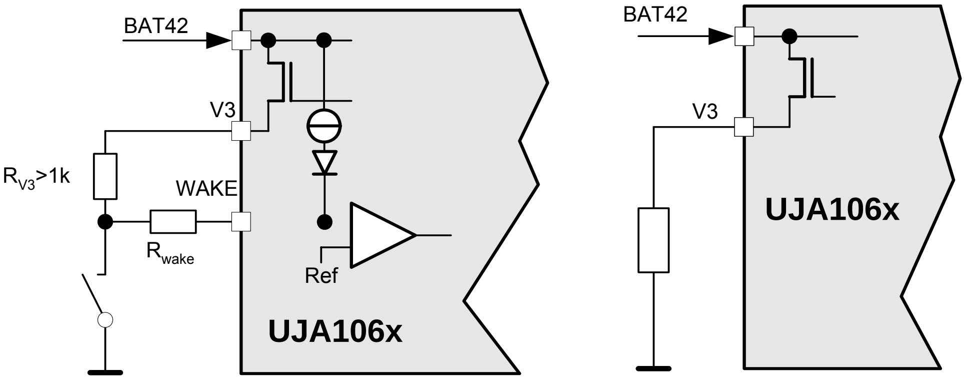


Figure 2-21: Typical Application of WAKE Inputs and V3 Output

For ESD Reasons V3 should always be applied with at least 1kΩ series resistance. For EMC and ESD Reasons a series resistor should also be applied at wake. To ensure safe detection of wake signals this resistor has to be smaller than $R_{wakemax}$. $R_{wakemax}$ is calculated depending on the expected maximum ground shift at the ECU according to Equation 2.

$$R_{Wake\max} = \frac{V_{th(wake)\min} - V_{gndshift}}{I_{wake(pu)\min}} = \frac{2V - V_{gndshift}}{25\mu A}$$

Equation 2: Maximum Wake Resistor

In case the ECU's GND is lost, a closed wake-switch towards an off-board GND can lead to significant currents out of pin WAKE. This could result in the destruction of the ESD protection diode at pin WAKE. Therefore a minimum resistance is needed at pin WAKE to limit this current. This resistance is calculated according to Equation 3.

$$R_{Wake\min} = \frac{V_{Bat\max}}{I_{wake\max}} = \frac{V_{Bat\max}}{15mA}, e.g. \frac{60V}{15mA} = 4k\Omega$$

Equation 3: Minimum Wake Resistor

2.9. Global System Enable

To prevent safety relevant application hardware from being unintentionally activated, the UJA106x provides a V1-related push-pull enable output pin (EN) that can be controlled by the application software via the SPI interface.

For fail-safe reasons the EN output function is linked to the internal watchdog of the UJA106x. EN can only be set HIGH if the watchdog is served properly. EN is immediately set LOW as soon as RSTN is forced LOW. Immediately after power-up and in Sleep and Fail-safe Mode EN is also set LOW.

In those SBCs that include a CAN transceiver (i.e. UJA1061 / 65 / 66) the enable pin can be programmed to output the SBC-internal failure register content of the CAN transceiver, similar to the NERR pin of the TJA1054 / 55 / 41. This can be useful for existing software drivers that are optimized for stand-alone transceivers.

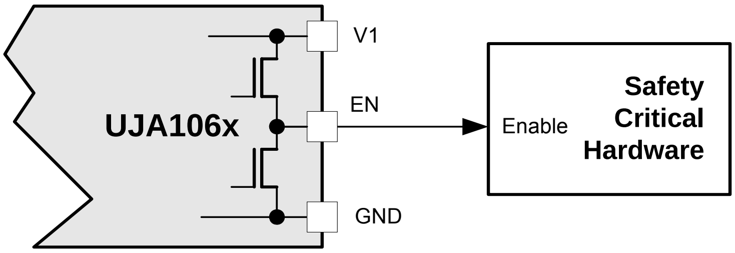


Figure 2-22: Typical Application of EN Output

2.10. Test Pin

The TEST pin is used to enter special modes for software development and for initial programming in mass production. To ensure that these modes cannot be entered by accident it is highly recommended to connect this pin to GND via a Pull-down resistor (e.g. 1kΩ). Additionally the layout has to make sure that this signal is routed very locally and is kept away from BAT related signals. For ESD protection a 10nF capacitor close to the test connector is recommended.

In Applications where this pin is not used at all it is recommended to connect it directly to ground.

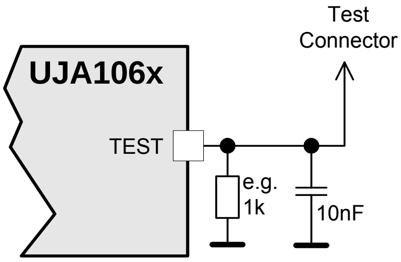


Figure 2-23: Typical Test Pin Application

Two special modes can be entered by applying certain voltages at this pin before switching on the supply:

- Software Development Mode (SDM)
- Forced Normal Mode (FNM)

The Forced Normal Node, in which almost the entire Fail Safe system of the SBC is disabled and all access to the SPI registers is blocked, is entered by applying between 13,5V and 15V at pin TEST before supplying the SBC. In this mode all regulators are permanently switched on and the CAN and LIN transceivers are always in Active Mode [1] - [4]. This mode is intended for initial programming of Hardware in mass production.

The Software Development Mode offers the SBC’s entire functionality but with relaxed watchdog and interrupt supervision. This mode is very useful when developing software for the SBC and allows hardware debugging with break points.

The Software Development Mode (SDM) is entered by applying a voltage at pin TEST prior to supplying the SBC. The minimum voltage that has to be applied to enter the SDM depends on chip temperature and production spread and lies between 1V and 8V. Since there is also an upper threshold, above which no special mode is entered, the optimum TEST pin voltage has to be found by the following iteration process:

Apply 1V at pin TEST, supply the SBC and check whether it is in SDM. If it is not, turn-off the SBC supply and increase the voltage at pin TEST by e.g. 0.5V. Turn-on the SBC Supply and check again. Repeat until SDM is entered. This procedure ensures that the SDM is entered with the lowest, device-specific TEST-pin voltage.

Since the threshold can only decrease with temperature, the SBC then stays in the SDM even when the Chip temperature increases.

If one of the special modes is entered via pin TEST, the voltage at pin TEST is continuously sampled and therefore has to be very stable. As soon as it falls outside the specified window, the special mode is left. Therefore it can for example be necessary to apply an additional Capacitor at pin TEST to eliminate voltage spikes.

Note, that SDM can also be entered without the help of the TEST-pin. A write access to the Special Modes Register before initializing the SBC the first time after battery power-on allows once initializing the SDM with a dedicated control bit. This initialization has to be done before the SBC is set into Normal Mode and thus, before the Software Start-up time has been expired. This option can only be used, if the development environment (software emulator etc.) starts-up fast enough after power-up of the system or within an early prototype phase with embedded software inside of the application program memory.

3. Pinning / Layout / FMEA

During definition of the UJA106x pinning, special attention was paid on pinning FMEA aspects. High voltage pins are consequently separated from low-voltage microcontroller pins and the 42V related pins are separated from the 12V pins. Non-connected pins (nc) are increasing the safety against erroneous shorts between these different voltage systems.

3.1. Layout Issues

During PCB layout this separation should be taken into account in order to prevent possible damages to the ECU by simple shorts on ECU level.

Although the heat sink on the back of the package is not connected to the GND level of the UJA106x, the connection of the Heat sink to the PCB's ground layer is crucial for the heat dissipation of the SBC and EMC reasons. Therefore the Heat sink has to be soldered directly onto the PCB and a good thermal connection to the GND layer has to be ensured e.g. by a sufficient amount of vias. In order to act as a good heat sink, the GND layer has to be of sufficient size. Additionally it shall be kept in mind that an outer GND layer can dissipate heat better than an inner layer.

One possibility to test whether the SBC is soldered on correctly is by injecting a backward current of some 100mA into V1 with BAT14 and BAT42 connected to GND as depicted in Figure 3-1. The quality of the thermal connection can then be assessed by measuring the reverse diode characteristic of the V1 voltage regulator and comparing it to a reference measurement. A bad soldering result show a significantly changed diode characteristic due to higher chip temperature compared to a well soldered device.

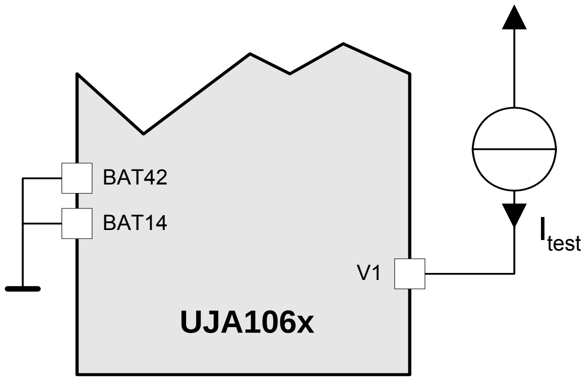


Figure 3-1: Test Application For Proper Heat Sink Connection

Further on it is highly recommended to place buffer capacitors very close to the supply and GND pins of the UJA106x in order to achieve optimum behavior of the device. This is particularly important for the smaller capacitors (e.g. the 47nF) shown within chapter 2.

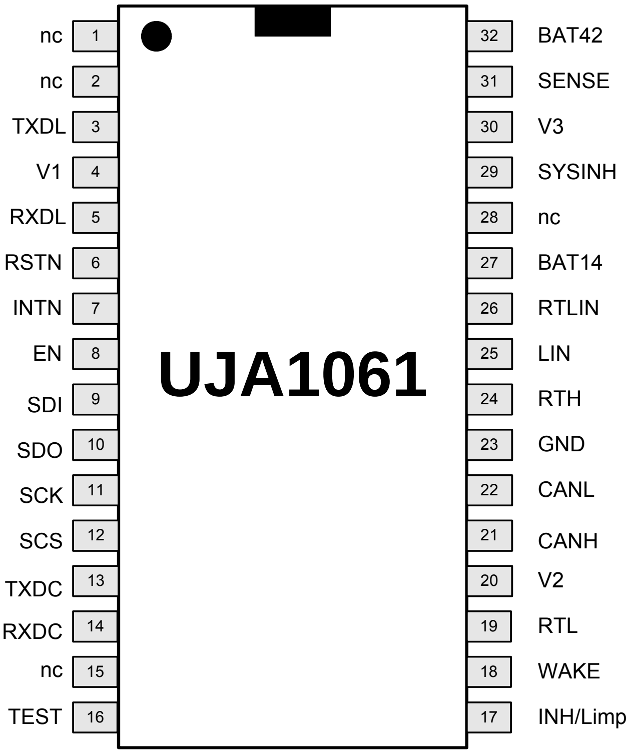
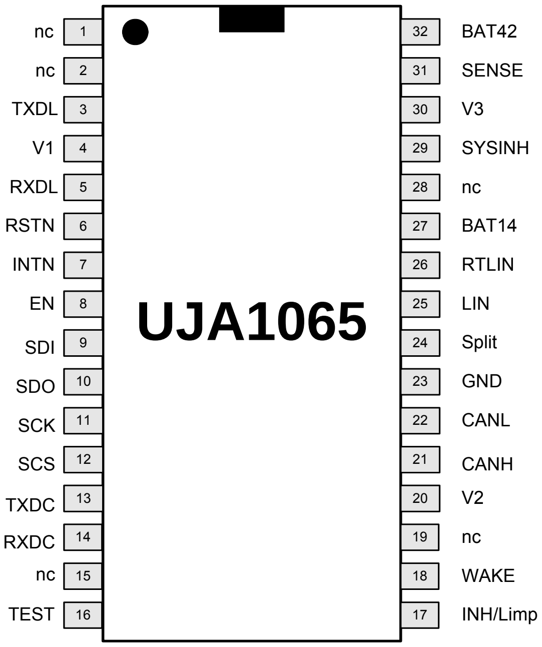


Figure 3-2: Pinning of the UJA1061



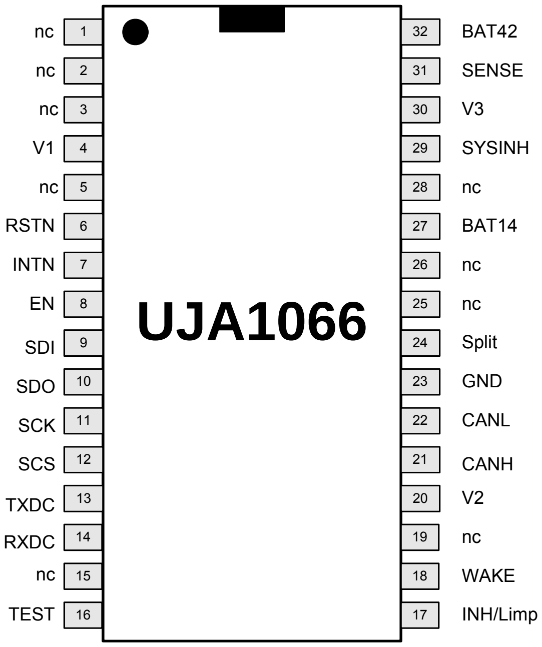


Figure 3-3: Pinning of the UJA1065 and UJA1066

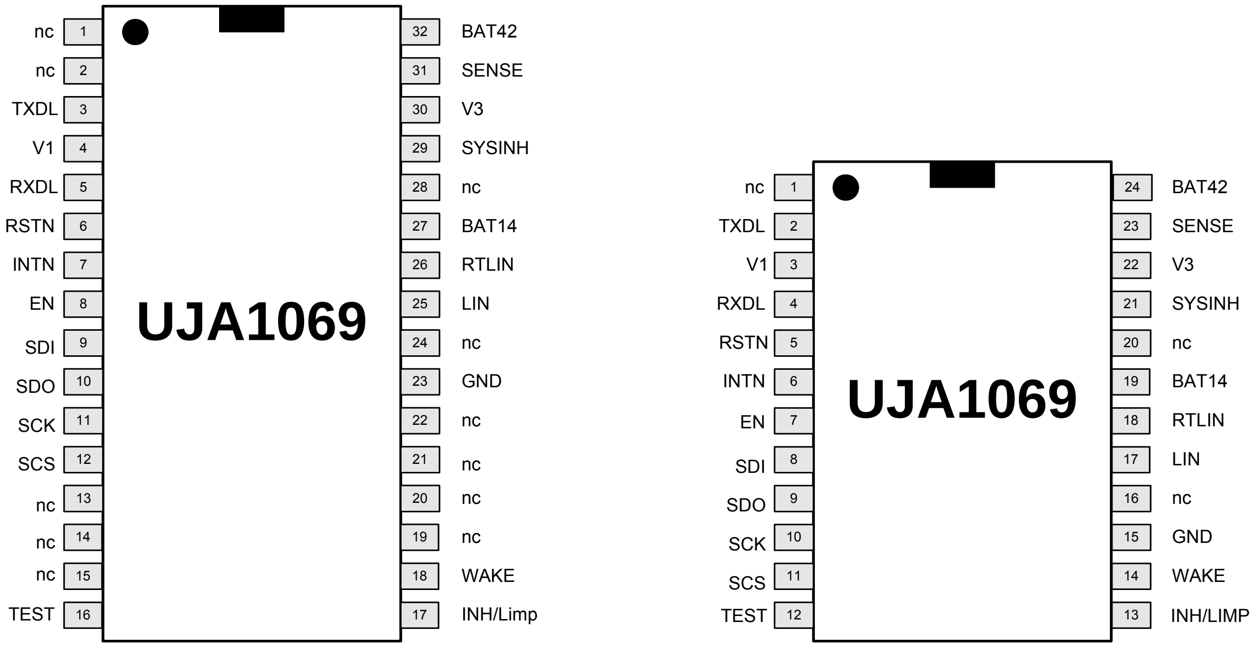


Figure 3-4: Pinning of the UJA1069 in the 32 an 24-pin package

3.2. Pin FMEA

The pinning of the UJA106x together with the integrated fail-safe features ensures, that shorts between adjacent pins do not result in a system dead lock.

Following classes of severity are distinguished:

Severity Class	Effects
A	- Damage to device - Serial communication on CAN and/or LIN may be affected globally
B	- No damage to device - Serial communication in the overall system not possible (global problem)
C	- No damage to device - Bus communication of other nodes in the system possible - Corrupted node not able to communicate (local problem) - Application might shut-down due to fail-safe behavior of SBC
D	- No damage to device - Bus communication in the overall system possible - Reduced functionality of application
-	- Not affected at all

Pin Name [2]	Failure	Remark	Severity Class
TXDL (not applicable for UJA1066)	Shorted to neighbor (nc)	-	-
	Shorted to neighbor (V1)	LIN stays recessive all time	D
	Shorted to GND	LIN stays recessive all time due to TXDL dominant timer protection	D
	Left open	LIN stays recessive all time	D
V1	Shorted to neighbor (RXDL)	LIN locally clamped recessive	D
	Shorted to GND	V1 off, system will enter fail-safe	C
	Open circuit	Microcontroller un-powered, system enters fail-safe state	C
RXDL (not applicable for UJA1066)	Shorted to neighbor (RSTN)	LIN message from slaves would cause repeated reset events, SBC enters fail-safe state	C
	Shorted to V1	No LIN communication	D
	Shorted to GND	No LIN communication	D
	Open circuit	No LIN communication	D
RSTN	Shorted to neighbor (INTN)	System interrupt causes permanent system reset, SBC reset monitoring function forces fail-safe state	C
	Shorted to V1	SBC reset monitoring function forces fail-safe state	C
	Shorted to GND	SBC reset monitoring function forces fail-safe state	C
	Open circuit	SBC reset monitoring function forces fail-safe state	C
INTN	Shorted to neighbor (EN)	No interrupts supported, unexpected interrupts,	D

[2] NC pin are regarded to be not connected in the application

Pin Name [2]	Failure	Remark	Severity Class
	Shorted to V1	SBC interrupt monitoring function forces fail-safe state on interrupts	D
	Shorted to GND	SBC interrupt monitoring function forces fail-safe state on interrupts	D
	Open circuit	SBC interrupt monitoring function forces fail-safe state on interrupts	D
EN	Shorted to neighbor (SDI)	No communication towards SBC -> fail-safe state	C
	Shorted to V1	No protection of critical application hardware, hardware always active	D
	Shorted to GND	Safety critical application hardware might be disabled all time	D
	Open circuit	Safety critical application hardware might be disabled all time	D
SDI	Shorted to neighbor (SDO)	No communication towards SBC -> fail-safe state	C
	Shorted to V1	No communication towards SBC -> fail-safe state	C
	Shorted to GND	No communication towards SBC -> fail-safe state	C
	Open circuit	No communication towards SBC -> fail-safe state	C
SDO	Shorted to neighbor (SCK)	No communication towards SBC -> fail-safe state	C
	Shorted to V1	Cannot read from SBC, behavior depends on software, fail-safe state possible	C / D
	Shorted to GND	Cannot read from SBC, behavior depends on software, fail-safe state possible	C / D
	Open circuit	Cannot read from SBC, behavior depends on software, fail-safe state possible	C / D
SCK	Shorted to neighbor (SCS)	No communication towards SBC -> fail-safe state	C
	Shorted to V1	No communication towards SBC -> fail-safe state	C
	Shorted to GND	No communication towards SBC -> fail-safe state	C
	Open circuit	No communication towards SBC -> fail-safe state	C
SCS	Shorted to neighbor (TXDC)	SPI failure interrupts, sporadic CAN error frames possible, depending on software fail-safe state possible	C / D
	Shorted to V1	No communication towards SBC -> fail-safe state	C
	Shorted to GND	No communication towards SBC -> fail-safe state	C
	Open circuit	No communication towards SBC -> fail-safe state	C
TXDC (not applicable for UJA1069)	Shorted to neighbor (RXDC)	SBC detects this short and disables the CAN transmitter -> no affect to other nodes (conventional transceivers clamp CAN continuously with this short)	C
	Shorted to V1	No transmission of CAN messages, node runs bus off, no effect on other nodes communication, reception still possible	C
	Shorted to GND	SBC detects this short and disables the CAN transmitter -> no effect on other nodes	C
	Open circuit	No transmission of CAN frames possible, CAN controller runs bus off, no effect on other nodes communication, reception still possible	C
RXDC	Shorted to neighbor (nc)	-	-

Pin Name [2]	Failure	Remark	Severity Class
(not applicable for UJA1069)	Shorted to V1	SBC detects this short and disables the CAN transmitter -> no affect to other nodes (conventional transceivers will disturb CAN communication with this short, because they expect an idle bus)	C
	Shorted to GND	No CAN communication possible, other nodes not affected	C
	Open circuit	SBC detects this short and disables the CAN transmitter -> no affect to other nodes (conventional transceivers will disturb CAN communication with interrupted RXD, because they expect an idle bus)	C
TEST	Shorted to neighbor (nc)	-	-
	Shorted to BAT14/BAT42	Possibility that Forced Normal Mode is entered on Battery connection with steep edge	D
	Shorted to V1	No test mode selected, pin needs to be pulled high before supplying the SBC at the BAT pins	
	Shorted to GND	No test mode, normal operation	-
	Open circuit	No test mode, normal operation	-
INH/LIMP	Shorted to neighbor (Wake)	If used as INH, external regulators might be kept active all time	D
	Shorted to BAT14 / BAT42	external regulator permanently active	D
	Shorted to GND	external regulator permanently off	D
	Open circuit	no control of external regulators possible	D
WAKE	Shorted to neighbor (RTL)	No damage to SBC, LIN termination might be affected while external signal connected to WAKE1 is pulled LOW -> temporary communication problem via CAN	C
	Shorted to BAT14 / BAT42	No damage to SBC, WAKE configuration: no local wake-up possible	D
	Shorted to GND	No local wake-up possible	D
	Open circuit	No local wake- possible	D
RTL (UJA1061)	Shorted to neighbor (V2)	Is the default for RTL -> no effect on normal communication, in case of additional bus failures on CANH, a slightly increased supply current is possible	-
	Shorted to BAT14 / BAT42	Handled by the fault-tolerant physical layer, single wire operation via CANH might occur, depending on the termination	-
	Shorted to GND	Handled by the fault-tolerant physical layer, single wire operation via CANH might occur, depending on the termination	-
	Open circuit	No CANL termination	-
V2 (not applicable for.	Shorted to neighbor (CANH)	Handled by the fault-tolerant physical layer, single wire operation via CANL	-
	Shorted to BAT14 / BAT42	SBC and other hardware is damaged	A

Pin Name [2]	Failure	Remark	Severity Class
UJA1069)	Shorted to GND	SBC detects this short and shuts-down V2 with reporting to the application -> no transmission possible, CAN controller might run bus off, reception still possible	C
	Open circuit	No buffering of V2, might increase EME	D
CANH (not applicable for UJA1069)	Shorted to neighbor (CANL)	Handled by the fault-tolerant physical layer, single wire operation via CANH	-
	Shorted to BAT14 / BAT42	Handled by the fault-tolerant physical layer, single wire operation via CANL	-
	Shorted to GND	Handled by the fault-tolerant physical layer, single wire operation via CANL, receiver stays differential	-
	Open circuit	Handled by the fault-tolerant physical layer, single wire operation via CANL, receiver stays differential	-
CANL (not applicable for UJA1069)	Shorted to neighbor (GND)	Handled by the fault-tolerant physical layer, single wire operation via CANH	-
	Shorted to BAT14 / BAT42	Handled by the fault-tolerant physical layer, single wire operation via CANH	-
	Shorted to GND	Handled by the fault-tolerant physical layer, single wire operation via CANH	-
	Open circuit	Handled by the fault-tolerant physical layer, single wire operation via CANH, receiver stays differential	-
GND	Shorted to neighbor (RTH, UJA1061 only)	Is the default for RTH -> no effect on normal communication, in case of additional bus failures on CANH, slightly increased supply current possible	-
	Shorted to neighbor (Split, UJA1065/66 only)	Slight increase in system supply current, communication not affected, bus failure will be signaled in CAN failure register if Split is connected to the split termination	D
	Shorted to BAT14 / BAT42	Fundamental problem of ECU, SBC not affected	C
	Shorted to GND	-	-
	Open circuit	Fundamental problem of ECU, no supply available, components on ECU might become damaged, SBC not affected	(A)
RTH (UJA1061 only)	Shorted to neighbor (LIN)	LIN pulled to 5V via 50 Ohms, LIN communication is down, CAN communication not affected but increased emission possible	C
	Shorted to BAT14 / BAT42	Handled by the fault-tolerant physical layer, single wire operation via CANL might occur, depending on termination scenario	-
	Shorted to GND	Is the default for RTH -> no effect on normal communication, in case of additional bus failures on CANH, increased supply current possible	-
	Open circuit	Integrated 6k termination of SBC guarantees default termination	-
Split (UJA1065 UJA1066	Shorted to neighbor (LIN)	LIN recessive voltage slightly decreased, some bias current will flow but communication not affected	D

Pin Name [2]	Failure	Remark	Severity Class
only)	Shorted to BAT14 / BAT42	Slight increase in system supply current, communication not affected, bus failure will be signaled in CAN failure register if Split is connected to the split termination	D
	Shorted to GND	Slight increase in system supply current, communication not affected, bus failure will be signaled in CAN failure register if Split is connected to the split termination	D
	Open circuit	DC stabilization not active, lower DC stabilization effect in system	-
LIN (not applicable for UJA1066)	Shorted to neighbor (RTLIN)	LIN communication might be affected due to low-ohmic termination to BAT, no damage to SBC, low-power modes still possible	C
	Shorted to BAT14 / BAT42	LIN communication down, no damage to SBC, low-power modes still possible	C
	Shorted to GND	LIN communication down, no damage to SBC, low-power modes still possible, termination current disabled via RTLIN switch	C
	Open circuit	LIN communication down, no damage to SBC, low-power modes still possible	C
RTLIN (not applicable for UJA1066)	Shorted to neighbor (BAT14)	Default level of RTLIN, loss of protection feature against unwanted currents on LIN, if LIN is shorted to GND	D / -
	Shorted to BAT14 / BAT42	BAT14 is default level of RTLIN, loss of protection feature against unwanted currents on LIN, if LIN is shorted to GND with short to BAT42, no communication possible	C / D
	Shorted to GND	No communication possible due to loss of master termination	C
	Open circuit	No communication possible due to loss of master termination	C
BAT14	Shorted to neighbor (nc)	-	-
	Shorted to BAT14 / BAT42	Short to BAT14 -> no problem Short to BAT42 -> in 14V system no problem since both pins are connected to BAT 14; In 42V system fundamental problem of ECU, SBC not damaged	D
	Shorted to GND	Fundamental problem of ECU, SBC not affected	C
	Open circuit	ECU not supplied -> fail-safe state	C
SYSINH	Shorted to neighbor (V3)	Both pins supplied out of BAT42, external DC/DC converter might stay active, depends on programming of V3 switch, no damage to SBC possible	D / -
	Shorted to BAT14 / BAT42	Weak switch to BAT42, slight bias current possible if shorted to BAT14, external DC/DC converter stays active all time	D
	Shorted to GND	External DC/DC converter cannot be started, system stays off all time -> fail-safe state entered If not used, slight bias current during normal operation, no current during low-power operation	C / -
	Open circuit	External DC/DC converter cannot be started, system stays off all time -> fail-safe state entered If not used, no effect	C / -

Pin Name [2]	Failure	Remark	Severity Class
V3	Shorted to neighbor (SENSE)	Unexpected “Loss of Battery” interrupt might be forced, application can disable V3 or the interrupt source in this case -> no affect anymore	D
	Shorted to BAT14 / BAT42	V3 is supplied out of BAT42, a short to BAT42 would not have an effect, cyclic supply of wake-up switches is replaced by continuous supply -> no problem Short to BAT14 will shut down V3 -> system gets status info about shut-down -> no problem, cyclic supply of wake-up switches is replaced by continuous supply -> no problem	D
	Shorted to GND	Short to BAT14 will shut down V3 -> system gets status info about shut-down -> no problem, Wake-up via switches supplied out of V3 are not possible if V3 is shorted during sleep, SBC will become waked-up in order to inform the application about the lost V3 supply and thus, about the lost wake-up source via V3 supplied switches	D
	Open circuit	Wake-up via switches supplied out of V3 are not possible, switches are unsupplied	D
SENSE	Shorted to neighbor (BAT42)	Input only behavior, is typically connected directly to BAT42 before the polarity protection diode, loss of battery monitoring feature	D
	Shorted to BAT14 / BAT42	Input only behavior, is typically connected directly to BAT42 or BAT14 before the polarity protection diode, loss of battery monitoring feature	D
	Shorted to GND	In case SENSE is connected to BAT14 or BAT42 the supply of the system is down -> fundamental problem of ECU, SBC not affected in case SENSE is not applied -> no affect	B / -
	Open circuit	No battery monitoring feature available	D
BAT42	Shorted to BAT14 / BAT42	Short to BAT14 is a fundamental ECU problem, SBC not affected	C
	Shorted to GND	Short to GND is a fundamental ECU problem, SBC not affected	C
	Open circuit	SBC not supplied -> fail-safe low-power state	C

4. Software meets Hardware

In this chapter some fundamental software issues are discussed dealing with system configuration and supervision. Depending on the external application hardware around the fail-safe SBC, a couple of register bits have to be set according to this specific hardware. This is needed to configure the System Basis Chip correctly and support the required hardware features.

For detailed information about the register and bit functions, please refer to A.1. The following paragraphs collect all relevant settings to offer an overview of the direct dependencies between the specific hardware around the fail-safe SBC and the software driver.

4.1. Supply System

4.1.1. Battery Monitoring

In case the SENSE[3] pin is connected to the battery input of the ECU, the fail-safe SBC is able to monitor the correct connection to battery and is able to detect a chattering battery contact. From software point of view, this feature needs to be enabled setting BATFIE ("1") within the Interrupt Enable Register of the fail-safe SBC. If this interrupt is enabled, a chattering battery contact is signaled to the software via the INT pin of the System Basis Chip and allows saving critical data within non-volatile memory before the microcontroller shuts down due to under-voltage.

It is a matter of course that sufficient supply buffering is needed on the ECU to withstand short power interruptions at the microcontroller supply, thereby allowing to take software action before the supply is down.

Apart from the interrupt the detected under-voltage at the SENSE Pin has no further impact on the fail-safe SBC's behavior. If this feature is not used it is recommended to disable the interrupt in order to reduce current consumption. Setting BATFIE "0" disables the pull-down current source integrated into the SENSE input pin of the SBC.

4.1.2. V1 Monitoring

V1 is intended to supply the main microcontroller of the ECU. A voltage drop at V1 below a selectable threshold immediately results in a system reset with the reset source "Low V1 supply, V1 has dropped below the selected reset threshold". The System Basis Chip provides three different Reset Thresholds that can be selected by the V1 Reset Threshold Control Bits (V1RTHC) within the Special Mode Register. The thresholds are fixed at 70%, 80% or 90% of the nominal V1 voltage (5V version only).

For fail-safe reasons, at first battery connection the fail-safe SBC always starts-up with the 90% under-voltage reset threshold. Thus, all microcontrollers are supported with a proper system reset, while V1 ramps up. Now the application can decide to select a lower reset threshold for future under-voltage resets. In order to stay fail-safe, the SBC automatically enables the 90% reset threshold again, each time Restart or Fail-Safe Mode is entered. This is, because the SBC assumes improper behavior of the microcontroller.

In order to still detect slight voltage drops, if the 70% or 80% reset threshold is selected, the fail-safe SBC provides an additional V1 under-voltage detector with a fixed 90% threshold. If V1 drops below this level the V1D bit within the Diagnosis Register is cleared. Additionally an interrupt can be forced if the Voltage Failure Interrupt is enabled (VFIE). If the SBC voltage V1 is used as reference voltage for an AD-converter the V1D bit can be used as a warning bit to inform the application that the AD-converter values might be wrong.

The V1D bit is only set again after successfully reading the Diagnosis Register while V1 is above the 90% threshold [4].

[3] The SENSE pin must not be used with the UJA1061. Please see 1.1.1

[4] In case of a bus failure it is possible that V1D bit is erroneously set if there are transients on the bus. So it is recommended not to evaluate the V1D information if bus failures are signaled in the Diagnosis Register.

4.1.3. CAN Transceiver, V2 Supply Monitoring

The CAN physical layer is based on a 5V supply due to the standardized bus levels. Thus, the V2 voltage regulator within the CAN transceiver needs to be active in order to support the physical layer with optimum behavior. Therefore V2 is always enabled whenever the CAN transceiver is in Active, On-line or On-line Listen Mode (see [1], [2] and [4] and 6.2.1, 6.2.2 and 6.2.3).

Since the CAN transceiver and the microcontroller have separate supplies (V1 and V2) the software should take care that the CAN transceiver is sufficiently supplied before starting a CAN transmission. There are two cases that have to be considered.

1. When V2 is activated it takes some time until V2 reaches the final value. Until then, received and transmitted CAN messages can be corrupted. The fail-safe SBC reports this state via the CAN Mode Diagnosis bits (CANMD) in the Diagnosis Register. As long as V2 is not charged CANMD shows that “CAN is in Off-line Mode or V2 is not active”. Especially when activating CAN, the software has to wait until CANMD reports that CAN is really in Active Mode before participating on CAN traffic. The time until V2 is finally charged depends on the capacitance at pin V2.
2. When V2 is on it might happen that it is deactivated due to an overload or a low BAT14 supply. In this case the fail-safe SBC clears the V2D bit within the Diagnosis Register and performs a Voltage Failure Interrupt (VFI), if enabled. **CAUTION: In order to prevent that corrupted messages are transmitted, the fail-safe SBC also disables the CAN transmitter by setting the CTC bit within the Physical Layer Register. To resume CAN traffic, the CTC bit has to be cleared by software.** For reactivating V2 after a failure see 6.3.2.

Conclusion: The CAN transceiver is ready for transmission if the CANMD bits say that CAN is in Active Mode, V2D is set and CTC is cleared.

4.1.4. V3 Monitoring

V3 is intended to supply peripheral components like relays, switches etc. An overload at V3 is reported via the V3 Diagnosis Bit (V3D) within the System Diagnosis Register. Additionally the Voltage Failure Interrupt (VFI) is forced, if enabled. Since V3 can be used to supply the local wake circuitry like wake-up switches, a loss of this voltage can result in losing local wake-up capabilities.

In case the SBC is in Standby Mode (microcontroller is supplied), an overload at V3 can be detected by the microcontroller via the V3D diagnosis function and the corresponding interrupt. In case of Sleep Mode, the microcontroller is un-powered and thus, cannot read the V3D bits. Therefore the SBC detects an overload of V3, even in Sleep Mode and wakes-up the system. The Reset Source is "V3 down due to short circuit situation during Sleep".

For reactivating V3 after a failure the V3 Control bits (V3C) just have to be rewritten.

4.2. WAKE Pin

The fail-safe SBC provides one WAKE pin for detecting local wake-ups. The pin is a battery related pin that is sensitive upon negative edges. Following bits of the UJA106x are dealing with this pin:

Table 4-1: Register bits for controlling WAKE pin

Bit	Register	Meaning
WEN	System Configuration Register	Wake Enable, decides whether WAKE pin is activated
WSC	System Configuration Register	Wake Sample Control, decides whether WAKE pin is sampled synchronously to V3
WLS	System Status Register	WAKE Level Status, shows whether WAKE pin is high or low, if enabled
EWS	System Status Register	Edge Wake Status, is set when a negative edge is detected at WAKE, cleared when Status Register is read
WI(E)	Interrupt Enable Register / Interrupt Register	Enables / indicates local wake-up interrupts
V3C	System Configuration Register	Selects operating mode of the V3 output (on/off/cyclic)

The actual status of this pin is reflected in the Wake Level Status bit WLS in the Status Register. A transition from high to low sets the Edge Wake Status bit EWS in the same register. This bit is cleared after the Status Register has been read out. An edge at the wake pin can additionally be signaled by an interrupt, if enabled via the Wake-up Interrupt Enable bit WIE. Besides signaling this event the SBC is waked-up out of any low-power state or the fail-safe SBC state.

If the Wake-up function is not used it should be disabled entirely. For this the Wake Enable bit (WEN) within the Configuration Register has to be cleared. The WAKE detector is then completely disabled. Thus also the WLS bit shows useless information.

4.2.1.1. System Behavior upon a wake-up via WAKE

In Standby Mode of the system (Microcontroller is still powered via V1) it can be chosen, whether the system should be waked-up by an interrupt or by a hardware reset. If the interrupt option is desired the Wake-up Interrupt (WIE) must be set within the Interrupt Enable Register. If the interrupt is not activated, a hardware reset is performed upon every negative edge on the WAKE pin. If neither a reset nor an interrupt is desired the wake-up function should be entirely disabled by clearing the WEN bit in the Configuration Register.

In Sleep and Fail-Safe Mode a negative edge at pin WAKE forces a system wake-up that activates V1 and performs a system reset. This function can be disabled by clearing the WEN bit in the Configuration Register.

Once a wake-up via WAKE is detected, the EWS bit is set until the Status Register is read. While this bit is set it is not possible to enter Sleep Mode or to disable the watchdog in Standby Mode. This makes sure that even last minute wake-up events cannot be lost, see also 5.6.5.

4.2.1.2. Wake-up Switch Supply out of V3

In combination with the power supply output V3 the local wake pin can be used for monitoring external switches. V3 can be configured to be continuously on or cyclically on with two different period times. This is configured using the V3 Configuration Bits (V3C) within the System Configuration Register.

Switching V3 continuously on (V3C = “01”) offers the fastest reaction time of the system upon a switching event. In this case V3 is continuously active and acts as a pull-up source for the local wake-up switches. Nevertheless, this mode has the highest power consumption, if there is a clamped wake-up switch to GND. This might discharge the battery of a car just due to a clamped switch.

A more fail-safe approach is configuring V3 with one of the cyclic modes (V3C = “10” or “11”). In this case V3 is enabled only for a short time period while there is a long period with unpowered switches. In order not to be waked by the cyclic switching of V3 the connected wake-pin must be configured in the Cycle Sample Mode by setting the WAKE Sample Control bit WSC ("1") within the System Configuration Register. Then the sampling of the wake pin is synchronized with the on-period of V3. If V3 is continuously on or if the wake-pins are not used in combination with V3 the wake pin has to be configured in the Continuously sample Mode by clearing the WSC bit ("0").

4.3. INH/LIMP Pin

The INH/LIMP pin of the fail-safe SBC is a battery related pin that can be used to control external hardware. Depending on the hardware configuration this pin can either be used as an inhibit pin or as a limp home output, see also 2.4.3. The pin can be in one of the three states, high, low and floating. The default setting after power-on is floating. Thus a connected pull-down or pull-up resistor decides whether the pin is high or low after power-on. After that the status of this pin can be controlled via two bits within the Configuration Register, see also [1] - [4]:

Table 4-2: Register bits for controlling INH/LIMP pin

Bit	Register	Meaning
ILEN	System Configuration Register	INH/LIMP enable, if set to "0" INH/LIMP is floating, if set to "1" INH/LIMP pin is high or low depending on ILC
ILC	System Configuration Register	INH/LIMP Control, controls status of INH/LIMP pin if ILEN is "1"

Table 4-3: INH/LIMP Bit Configuration

ILEN	ILC	Function (INH/LIMP)
0	0	Floating
0	1	Floating
1	0	Low (Limp Home active)
1	1	High (Inhibit active)

In case the INH/LIMP output is used for realizing an inhibit function it is normally low at power-on. The external hardware, e.g. an additional voltage regulator is then off. The software can decide to switch this hardware on by setting the INH/LIMP pin to active HIGH. For this the ILEN and ILC bits must both be set to "1". In certain situations, e.g. if the fail-safe SBC enters Sleep or Fail-Safe Mode, the INH/LIMP pin is set "floating" or "active low" so that the controlled hardware is automatically disabled.

Alternatively the INH/LIMP pin can be used as a Limp Home output. In this case it normally is HIGH at power-on. Whenever the SBC enters Fail-Safe Mode the INH/LIMP pin is set to active LOW. This signal allows activating e.g. warning lights or similar stuff via a PNP or P-channel transistor once the ECU fails completely.

4.4. Reset Length Control

Depending on the used microcontroller, a suitable reset length can be programmed.

The fail-safe SBC family supports two different reset lengths that can be adjusted by the **Reset Length Control bit (RLC)** within the System Configuration Register. The default setting after power-up of the SBC is the short reset length (RLC = "0"). For fail-safe reasons, the long reset pulse becomes automatically active in case of any start-up procedure problems. If the SBC is not started correctly in software after a reset event, the so-called Restart Mode is entered with a long reset pulse. This makes sure that in case the microcontroller has a problem with the short reset pulse the reset is repeated with a long period.

For fail-safe reasons it is not recommended to lengthen the external reset signals via external capacitors. External capacitors would cause an analogue R/C-based up-ramping of the reset signal thus leading to unpredictable reset timing between different devices connected to the SBC RSTN port. Nevertheless it is possible to lengthen the reset period e.g. by the microcontroller itself, holding the RSTN pin actively low for some extra time. The UJA106x monitors the RSTN input and checks, whether it is clamped for to long. If a certain time is exceeded the Fail-Safe state is entered (reset clamping detection feature).

4.5. Temperature Monitoring

The SBC sets the Temperature Warning Status Bit (TWS) within the System Status Register as soon as the chip temperature exceeds a certain level. Additionally the Over Temperature Interrupt (OTI) is forced, if enabled. To avoid damage to the system in case of over-temperature it is recommended to send the SBC into Fail Safe Mode. This can be done by repeatedly writing any illegal command into the Mode Register, e.g. 0x0000.

4.6. Behavior at Battery Undervoltage

When the supply at the battery pins drops below the minimum operating voltage, there are two different scenarios possible (see also Figure 4-1):

a) If the battery voltage is lower than the BAT undervoltage level of the SBC $V_{POR(BAT)}$, the SBC will be shut down and it will perform a cold-start when the supply voltage rises again, with the same behavior like first battery connection.

b) If the battery voltage stays above the BAT undervoltage level of the SBC, but V1 falls below the V1 undervoltage level (depending on the load current on V1 and on the individual voltage drops in an ECU) for longer than the V1 clamped LOW time $t_{V1(CLT)}$, the SBC does not perform a power-on reset, but it enters Fail-safe mode. When the supply voltage rises again the SBC leaves Fail-Safe mode after receiving a wake-up via CAN, LIN or the local WAKE pin and after the Fail-Safe retention timeout t_{ret} has elapsed (see also [1] - [4]).

Which of both scenarios will happen depends on the voltage drop of the V1 regulator and the actual V1 and power-on undervoltage threshold. This is illustrated in Figure 4-1 for the case that the pins BAT42 and BAT14 are shorted so that V_{BAT} is the actual supply at both pins. The figure shows a voltage drop of V_{BAT} in a VBAT versus time and V_{V1} versus V_{BAT} diagram.

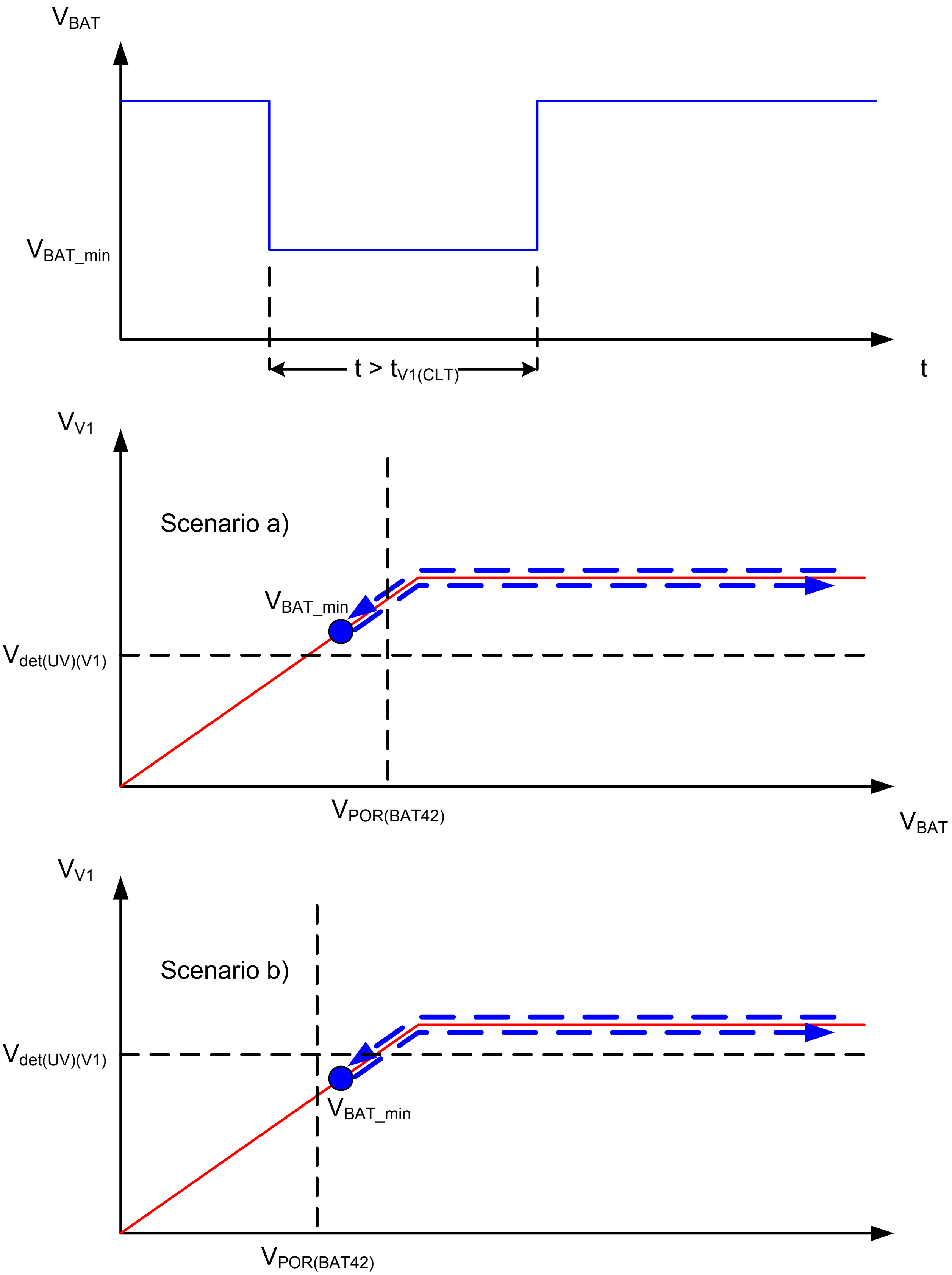


Figure 4-1: Behaviour of the SBC at temporary battery voltage drop

Influencing the conditions for scenario b) by changing the V1 undervoltage threshold

Since the power-on threshold and the drop of the regulator are device parameters that cannot be changed, only the selection of the V1 undervoltage threshold has influence on the undervoltage behavior of the SBC. Scenario b) can only happen if the battery supply limit for the case that V1 enters undervoltage condition is above the power-on threshold. Beside this the battery supply during the voltage drop has to hit the window between this limit and the threshold (see Figure 4-2). The way to calculate the width of this window is illustrated in the following example.

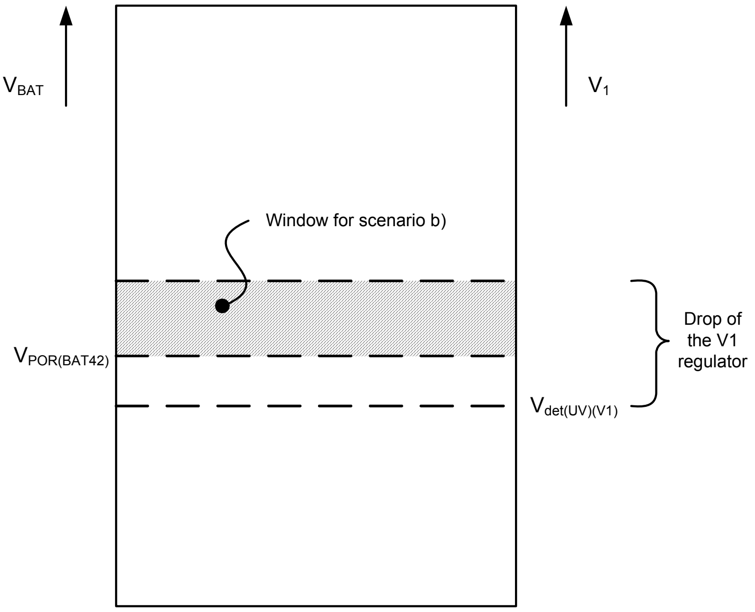


Figure 4-2: Dependency of the scenario b) window width from $V_{POR(BAT42)}$, $V_{det(UV)(V1)}$ and the voltage drop of the V1 regulator

4.6.1.1. Example: Calculation of the maximum width of the window for scenario b)

In a 5V application the maximum V1 current amounts 120mA. The 80% V1 undervoltage threshold is selected. The question is how wide the maximum possible width of the window for scenario b) is.

The power-on threshold for deactivation of the SBC is according to the datasheet between 4.45V and 5V. For calculation of the maximum width the lower power-on threshold has to be used. The V1 undervoltage threshold is between 4V and 4.25V for the 80% threshold. Here the upper limit has to be used for the worst case calculation. For determining the dedicated battery voltage, the drop of the regulator has to be added. According to the datasheet the internal resistance of the regulator is lower than 5Ω. This results in a voltage drop of 600mV, assuming an output current of 120mA. So if the battery supply drops below 4.25V + 600mV = 4.85V the SBC might enter V1 undervoltage condition. That means that if in this configuration the battery supply drops between 4.45V and 4.85V, there is a potential risk for scenario b).

With the same calculation it can be found that for the 90% undervoltage threshold the window is between 4.45V and 5.35V, and for the 70% threshold there is no window anymore.

5. Getting Started, Modes of the SBC

This chapter introduces the fail-safe SBC from the software's point of view. However this is not a complete description of the UJA106x. Instead it focuses on those parts of the UJA106x that can be influenced by software.

5.1. Software View of SBC Operating Modes

The fail-safe SBC is able to detect hardware and software system failures and to react on them in a defined way. This requires an advanced state diagram with many connections between the modes that can be run through under certain conditions. This state diagram is depicted in [1] - [4]. However, from the software's view not all states and their connections can be used directly. For example in, Sleep Mode the microcontroller is unpowered, so that no software can run here. For a software designer it is also not of interest what happens in case the microcontroller supply voltage V1 fails. Therefore it makes sense to work out a simplified software focused state diagram of the fail-safe SBC.

5.2. Simplified software state diagram of the UJA106x

The following software state diagram is derived from the complete state diagram of the fail-safe SBC by cancelling all transitions that cannot be influenced by the software.

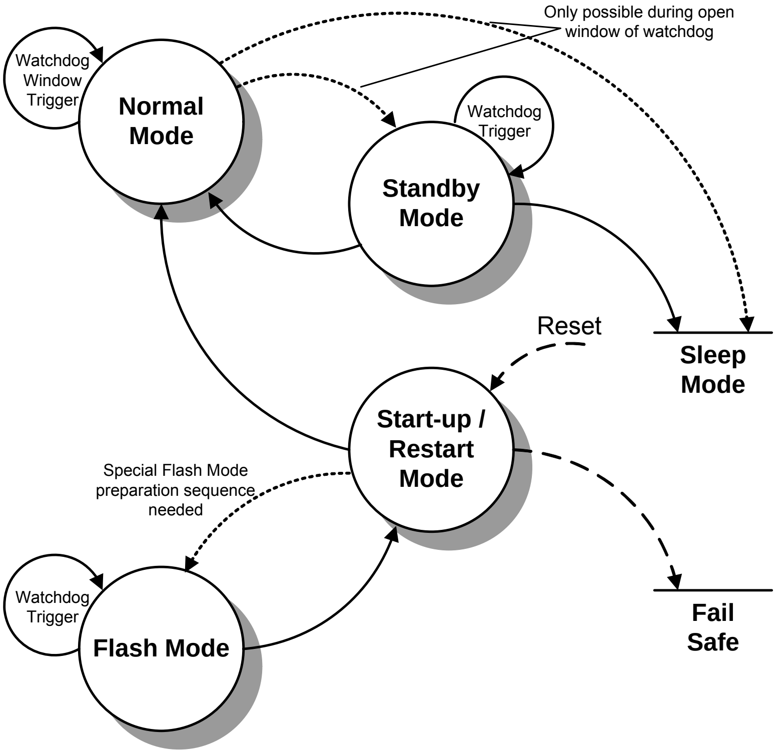


Figure 5-1: Simplified Software State Diagram of the SBC

First of all it is obvious that the Sleep and the Fail Safe Mode are not shown as individual states. They are rather illustrated as a kind of dead end. That is because the microcontroller is unpowered (V1 is off here) in both modes and therefore the software cannot work. The Sleep and the Fail Safe Mode can only be left via a wake-up event causing a system reset (V1 will start from zero with the according Reset).

After every reset event the SBC enters Start-up or Restart Mode. In general the software cannot distinguish between Start-up and Restart Mode. Consequently both modes were put together.

Between the states only those connections are drawn that can be controlled via registers of the fail-safe SBC. Some of these lines are dashed, e.g. the transition from Start-up / Restart Mode to the Fail Safe Mode. These transitions cannot directly be chosen via an instruction. They can only be used indirect, e.g. by a “SPI failure”. This could make sense if for example the software has found a serious failure in the program memory during a diagnosis routine and decides therefore to enter the Fail Safe Mode to ensure that the rest of the system cannot be disturbed. In such a situation the microcontroller can provoke a system reset by writing a wrong code to the Mode Register (see also A.6).

Some of the transitions between the states are bound to certain conditions. These connections are drawn dotted in Figure 5-1. These are the transitions in Flash Mode and all connections coming from Normal Mode. In Normal Mode the watchdog of the SBC is in the so-called window mode. That means that the watchdog can only be triggered within a certain time window. Since watchdog triggering and mode changes are both done by writing to the Mode Register, regular mode changes are only possible within open watchdog windows (see also 5.4.1).

The transition into Flash Mode is only allowed after a certain procedure. The course of this procedure is described in 5.7.1. This prevents that the Flash Mode can be started by a defect software routine. After the Flash procedure has been completed successfully the SBC resets the system and report “SBC ready to enter Flash Mode" as reset source (see also 5.7.1). Only then the Flash Mode can be initialized.

In the following sections all modes shown in Figure 5-1 are described in detail and all transitions are explained.

5.3. Start-up / Restart Mode

The Start-up / Restart Mode is an intermediate mode that is entered after each system reset. This is the mode after powering up of the system and the starting point for any operation with the UJA106x. Start-up and Restart Mode are intended to give a detailed overview of the current situation of the system and to choose an appropriate initialization scenario for the application. The Start-up / Restart Mode is terminated by initialization of the fail-safe SBC, that means entering Normal or Flash Mode.

While staying in Start-up or Restart, the watchdog is operating in a special timeout mode with one single period of about 256ms. The initialization time starts after releasing the RSTN pin to HIGH. Before the initialization time has elapsed the fail-safe SBC has to enter Normal or Flash Mode. Otherwise the fail-safe SBC performs a system reset and reports an initialization failure as reset source.

The fail-safe SBC causes system resets due to several reasons. In general it can be distinguished between regular and failure-caused resets. Regular resets can occur in a correct running system if the related features are used. Failure-caused resets are only forced in case the fail-safe SBC discovers a system failure. Table 5-1 shows how the different reset sources are generally classified. However, the exact meaning depends on the concrete application. A watchdog overflow for example normally indicates that the software of the microcontroller has failed. But it can also mean a wanted cyclic wake-up during Standby Mode.

Table 5-1: Classification of the Reset Sources

Reset Source	Classification
Power-on reset, external reset	Regular, Application specific
Cyclic wake-up out of Sleep	Regular
Low V1 supply	Failure-caused
V1 current above threshold within Standby Mode	Regular
V3 down due to overload occurring during Sleep Mode	Failure-caused
SBC successfully left Flash Mode	Regular
SBC ready to enter Flash Mode	Regular
Wake-up via CAN	Regular
Wake-up via LIN	Regular
Local wake-up event	Regular
Wake-up out of Fail-Safe Mode	Regular
Watchdog overflow	Application-specific
Watchdog not initialized in time	Failure-caused
Watchdog triggered too early	Failure-caused
Illegal SPI access	Failure-caused
Interrupt not served in time	Failure-caused

5.3.1. Register access during Start-up or Restart Mode

Generally the software cannot distinguish between Start-up and Restart Mode. Therefore in Figure 5-1both states are put together in one state. The microcontroller has full read-access to all SBC registers during both modes. However, it is only possible to write into the General Purpose Registers, the Special Mode Register and the Mode Register. Writing an allowed code into the Mode Register initializes the SBC and ends Start-up / Restart Mode.

5.3.2. CAN / LIN Status during Start-up or Restart Mode

When entering Start-up / Restart Mode, the CAN- and the LIN- transceiver leave their Active Mode regardless of the CAN and the LIN Mode Control bits (CMC and LMC) within the Physical Layer Control register (see 6.2.1 and 7.2.1). Hence it is not possible to transmit and receive data via CAN or LIN. If there is activity on the bus lines this can be interpreted as a wake-up event. Then the dedicated Wake Status bit within the Status Register is set (CWS or LWS) and the RXDC or RXDL pin is set LOW.

5.3.3. V3 Status during Start-up or Restart Mode

The power supply output V3 remains untouched during these modes.

5.3.4. EN Status during Start-up or Restart Mode

The EN-Pin is intended to provide a global hardware enable output for safety critical hardware. The EN pin is always low in Start-up / Restart Mode so that these critical hardware components are always disabled after a reset.

Note: In case EN is configured to output the CAN failure status (ERREM Bit in Special Mode Register), the EN permanently reflects the current content of the CAN Failure Flags in the System Diagnosis Register.

5.3.5. Transition from Start-up to Restart Mode

The Restart Mode is activated when a system reset occurs during Start-up Mode. The system reset can either be caused by an external component or by the fail-safe SBC itself. The UJA106x performs a system reset during the Start-up Mode if its initialization has failed. This can happen if the fail-safe SBC is not initialized in time, e.g. if a faulty SPI-access is carried out or if a forbidden value is written into the Mode Register. Such failures during initialization can only occur in case of a serious system failure or, if the system reset entering the Start-up Mode was not successful. The Restart Mode was initiated to give the microcontroller in the latter case a second chance for initializing the fail-safe SBC. Some parameters of the fail-safe SBC are automatically changed when entering the Restart Mode. These are the Reset Length RLC, the V1 Reset Threshold V1RTC and the status of the INH-Pin if it is used as inhibit pin. This ensures that the second system start-up is performed with fail-safe parameter settings.

5.3.6. Transition from Restart to Fail-safe Mode

Whenever the fail-safe SBC initialization fails during Restart Mode it is assumed that there is a serious system failure. In this case the System Basis Chip enters the so-called Fail-safe Mode. In the Fail-safe Mode V1 is disabled and the reset pin PSTN is down. Furthermore the CAN and the LIN transceiver leave their Active Mode. This ensures that the affected ECU does not disturb the rest of the network and it reduces the power consumption to a minimum value.

The SBC can only leave the Fail Safe Mode via a system wake-up or a real power-on condition when the battery supply at pin BAT42 passes the power-on threshold. The possible wake-up sources are the local WAKE pin, the CAN or the LIN bus. After the wake event a timeout with duration of about 1.5s passes in order to guarantee a properly discharged V1 buffer capacitor before the ECU becomes re-booted. After that time with active discharging of the V1 buffers, V1 is enabled again and Start-up Mode is entered after a system reset. The microcontroller can now try to initialize the fail-safe SBC again.

5.3.7. Entering Normal Mode

The following figure shows a simple flowchart of a typical system initialization routine. At first the microcontroller reads out an overview of the current status of the system by accessing the System Status Register and the System Diagnosis Register. Based on the received information, the microcontroller can now decide on how to initialize the system. E.g. after a power-on reset some general settings in the system have to be performed.

Now the software initializes the fail-safe SBC by writing the corresponding initialization code into the Mode Register. This command contains the “Init Normal Mode” code and the required watchdog period. Furthermore the status of the EN output is defined here.

Due to fail-safe reasons, the “Init Normal Mode” code is only allowed once after a reset event. Using this code twice is an indication for a software hang-up. The fail-safe SBC interprets this as an illegal SPI command and forces a reset. Furthermore this mechanism provides a “hand-shake” between the SBC and the microcontroller for each reset event. This allows the fail-safe SBC to detect an interrupted reset wire.

It should be mentioned that the fail-safe SBC monitors the start-up period of the software. After releasing the reset pin, the software has about 256ms time to initialize the Normal Mode. If this does not happen, the fail-safe SBC assumes a failed start of the microcontroller and performs a reset that leads into Restart or Fail-safe Mode.

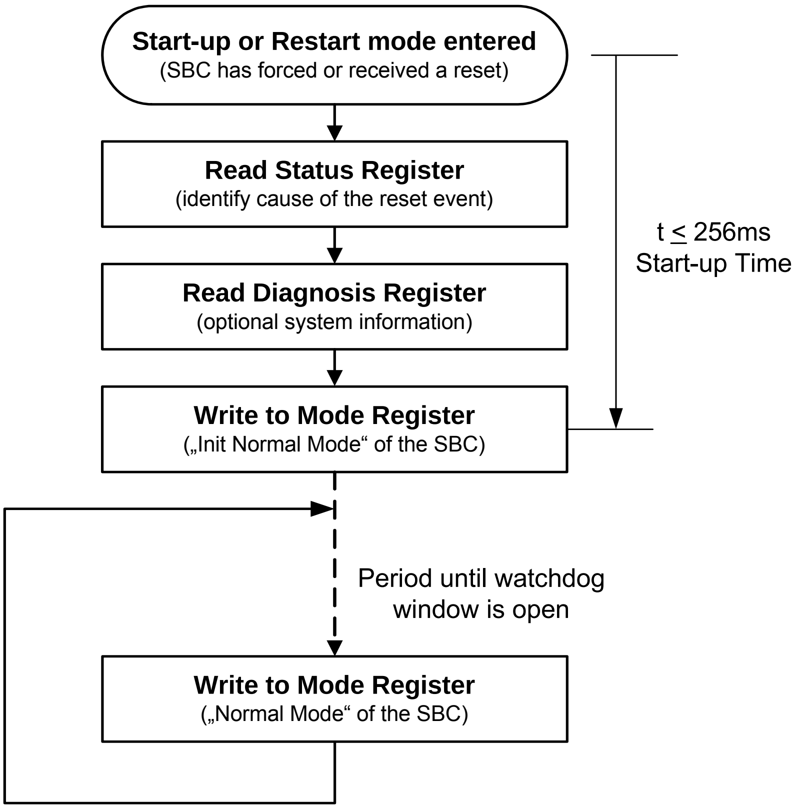

Start-up or Restart mode entered
(SBC has forced or received a reset)
Read Status Register
(identify cause of the reset event)
Read Diagnosis Register
(optional system information)
Write to Mode Register
(„Init Normal Mode“ of the SBC)
t ≤ 256ms
Start-up Time
Period until watchdog
window is open
Write to Mode Register
(„Normal Mode“ of the SBC)

Figure 5-2: Software flow, entering Normal Mode of the fail-safe SBC

5.4. Normal Mode

The Normal Mode is the basic operating mode of the fail-safe SBC. In this mode the microcontroller can activate the CAN and the LIN transceiver and has thereby access to the bus system.

The microcontroller can read out all UJA106x registers and write into the Interrupt Enable, the System Configuration and the Physical Layer Control Register. This allows configuring the SBC, as it is necessary for the specific application. Writing into the General Purpose Registers and the Special Mode Register in Normal Mode is not possible. This prevents these registers from being erroneously overwritten by corrupted software.

Please note that for the UJA1066 (HS-CAN transceiver), which has no LIN interface, bit 0 and bit 1 of the “Physical Layer Control” register have to be set by software as follows: Bit 0 shall be set to 1 and bit 1 shall be set to 0.

5.4.1. Watchdog triggering

In Normal Mode the microcontroller has unlimited access to all system functions including CAN and LIN bus networking. This requires a strict observation by the watchdog. Therefore in Normal Mode the watchdog has the shortest cycles and it works in its so-called window mode.

The window mode is characterized by the fact that the trigger window is closed for the first half of the watchdog period after a successful trigger event. In the second half the window is open. Triggering the watchdog is only permitted when the window is open. When the watchdog is triggered too early or when the watchdog overflows, the fail-safe SBC immediately forces a system reset.

The observation of the software by the watchdog is only guaranteed, if the trigger signals come directly out of the running program and not out of a timer controlled interrupt routine. To select the correct trigger moment the tolerances of the watchdog timer must be taken into account. The software has to send the trigger signal after the latest possible window opening and before the earliest end of a period.

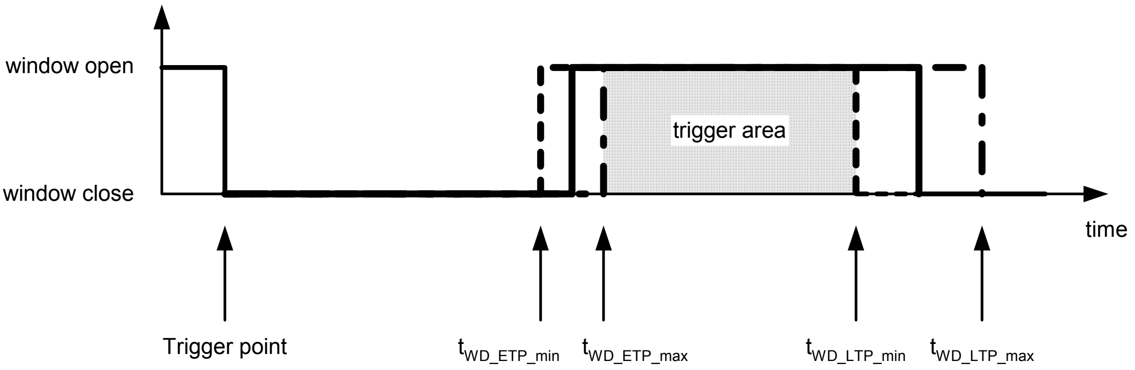


Figure 5-3: Watchdog in window mode, allowed trigger area

As shown in Figure 5-3, the trigger moment has to be located between the maximum value of t_{WD_ETP} and the minimum value of t_{WD_LTP}. The related values can be found in [1] - [4]. In case the tolerances of the microcontroller's clock generator cannot be neglected the following equations should be used to define the correct trigger moment:

$$t_{trigger\_min} > \frac{t_{WD\_ETP\_max}}{1-F}, \qquad t_{trigger\_max} < \frac{t_{WD\_LTP\_min}}{1+F}.$$

F here is the magnitude of relative deviation of the microcontroller's clock frequency. It is calculated by:

$$F = \left|\frac{f - f_0}{f_0}\right|$$

where f is the actual frequency and f_0 is the nominal frequency. $t_{trigger\_min}$ and $t_{trigger\_max}$ limit the area where the software has to choose the trigger point supposing the nominal frequency of the clock generator.

A watchdog trigger is performed with every valid write access to the Mode Register. After a successful SPI transfer the watchdog is reset. The transmitted value for the Nominal Watchdog Period NWP (see A.1) defines the duration of the next watchdog cycle. This is shown in Figure 5-4.

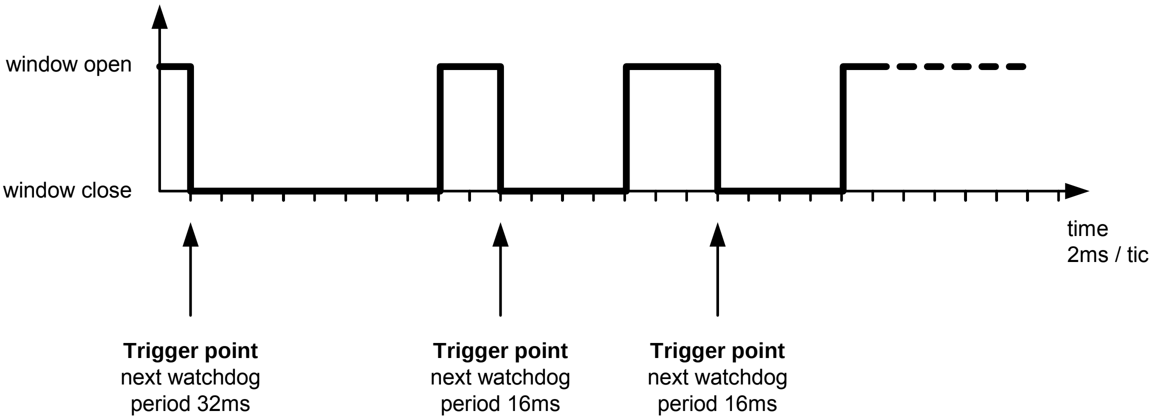


Figure 5-4: Watchdog triggering with different periods

It is important to understand that any intended Operation Mode change of the SBC is coupled with a watchdog trigger event. This makes sure that a new watchdog period starts whenever a new SBC mode is entered.

Because of the coupling with the watchdog every access to the Mode Register is just allowed during an open watchdog window. This must be considered if e.g. the status of the EN pin should be changed.

Hint:

For easy software development, the UJA106x supports a so-called “Software Development Mode” that disables the watchdog function during first coding. Please refer to the corresponding chapter within this document.

5.4.1.1. Example: Watchdog Triggering, entering Sleep Mode

The SBC is in Normal Mode and the software decides to go into Sleep Mode. In Sleep Mode the watchdog should be disabled. The Software Development Mode is not used.

After the decision to go into Sleep Mode the software prepares the next Mode Register command. The Mode Register Address is **00X0** XXXX XXXX XXXX. The watchdog disable function in Sleep Mode is selected with 00X0 **1101** **10**XX XXXX. The code for the Sleep is defined as 00X0 1101 10**10** **0**XXX. The Software Development Mode is disabled during the whole session 00X0 1101 1010 0**0**XX. The setting of the EN bit is not important because it is automatically cleared after entering Sleep Mode. The resulting Mode Register command is: **0000 1101 1010 0000 = 0x0DA0**. This command has to be written within the open window of the watchdog.

5.5. Standby Mode

The Standby Mode is a low power mode of the fail-safe SBC that is characterized by deactivation of the CAN and the LIN transceiver for reducing the power consumption. Hence, neither of the transceivers can send nor receive data. Instead activity on both busses is registered and interpreted as a wake signal. The WAKE pin is also observed. Depending on the specific configuration a wake signal results in an interrupt or a system reset.

5.5.1. V1 Status during Standby Mode

In Standby Mode the voltage V1 is permanently activated for supplying the microcontroller and the memory. The read and write access to all available UJA106x registers is enabled (similar to the Normal Mode it is not possible to write into the Special Mode Register and to the General Purpose Registers). The state of the EN pin depends on the setting in the Mode Register. In principle the microcontroller is able to work as in Normal Mode but without having access to the CAN and LIN bus. Since V1 is activated, the microcontroller may also enter a low power mode without loosing the content of the volatile memory.

As like in Normal Mode the fail-safe SBC permanently observes the supply voltage V1 of the microcontroller. Any voltage drop below the selected V1 under-voltage threshold immediately results in a system reset.

5.5.2. V2 and CAN status during Standby Mode

The state of the power supply for the CAN Transceiver V2 depends on the state of the CAN transceiver and on the V2 Control bit (V2C) within the Physical Layer Control Register. The CAN transceiver can never be in Active Mode while the fail-safe SBC is in Standby Mode. But if there is CAN communication on the bus the transceiver can be in On-line or On-line Listen Mode, depending on the CPNC bit in the Physical Layer Control Register. In these states V2 is on. As soon as the CAN transceiver enters Off-line Mode V2 is deactivated. However, the user can decide to keep V2 on even in Off-line Mode in order supply other components if CAN is not used. For this the V2C must be set to “1”.

5.5.3. Watchdog Behavior in Standby Mode

In Standby Mode the watchdog is still active. It is working in timeout mode. In contrast to the window mode the watchdog can be triggered at any time before the watchdog period elapses. The watchdog periods are longer compared to Normal Mode. It is also possible to deactivate the watchdog entirely (see 5.5.3.4). The interaction between microcontroller and watchdog can be used in different ways. This is discussed in the following paragraphs.

5.5.3.1. Running microcontroller during Standby Mode

Standby Mode is mainly used as a low power mode that allows fast start-up of the microcontroller or as an intermediate mode before entering Sleep Mode. In the latter case it allows for example to store important data in non-volatile memory and to quickly react on unexpected events during the power down process. For the early detection of erroneously running software it is recommended to use a short watchdog period. The interrupt option for the watchdog overflow during Standby Mode should then be disabled (WTE-Bit in the Interrupt Enable Register, see 5.5.3.3).

The watchdog is triggered by a valid write access to the Mode Register as in Normal Mode. The Nominal Watchdog Period determines the duration of the next watchdog cycle. For fail-safe reasons it is not possible to disable the watchdog while the microcontroller is still running (see 5.5.3.4).

5.5.3.2. Cyclic wake-up out of Standby Mode via hardware reset

In Standby Mode the watchdog can be used for cyclically waking the microcontroller. For this the microcontroller triggers the watchdog with the desired period and goes into stop mode directly afterwards. When the watchdog overflows the fail-safe SBC causes a system reset and enters Start-up Mode. The microcontroller becomes awaked. The RSS bits in the Status Register report a “watchdog overflow” as reason for this reset. Since the microcontroller’s memory was continuously powered, the system initialization might be shortened significantly.

If the watchdog overflow should cause a system reset it must be attended that the Watchdog Timeout Interrupt bit (WTIE) in the Interrupt Enable Register must be set to “0”.

5.5.3.3. Cyclic wake-up in Standby Mode via interrupt

Similar to the operating mode discussed in the previous section the watchdog can also be used to cyclically wake-up the microcontroller via interrupt with the pin INTN. For this the Watchdog Timeout Interrupt (WTIE) in the Interrupt Enable Register must be set to “1”. If then the watchdog period elapses the fail-safe SBC stays in Standby Mode and only performs an interrupt that wakes-up the microcontroller.

It is important to note that the microcontroller must react on this interrupt within the time defined by $t_{RSTN(INT)}$ [1] - [4] by reading the Interrupt Register. Otherwise the fail-safe SBC assumes that the microcontroller’s software has hung-up or that the interrupt wire is damaged and therefore performs a system reset with the reset source “interrupt not served in $t_{RSTN(INT)}$”.

Furthermore it is important to know that for fail-safe reasons the Watchdog Timeout Interrupt is automatically disabled each time the interrupt is performed. The software has to take care that the WTIE bit is set to “1” each time the microcontroller goes into stop mode. Otherwise the next watchdog overflow results in a system reset. This procedure allows the fail-safe SBC to detect software hang-ups even if the interrupt service routine of the microcontroller is still working. To guarantee this safety feature it is recommended to implement the reading sequence for the Interrupt Register in the interrupt service routine while the reactivation of the WTIE bit should be incorporated within the main program.

5.5.3.4. Disabling the Watchdog

During Standby Mode it is possible to entirely disable the watchdog. But for fail-safe reasons, this is only allowed in case the microcontroller reduces its current consumption by entering low power mode and if there is no pending wake-up. If the Mode Register is triggered in Standby Mode with the “Watchdog off” code as Nominal Watchdog Period (NWP) the watchdog starts a new cycle with the previous cycle time. At the end of this time the fail-safe SBC checks if the current consumption at pin V1 has dropped below the undercurrent threshold $I_{thL(V1)}$ [1] - [4] and if neither of the wake status flags (CWS, LWS, EWS) is set in the Status Register. If all conditions are fulfilled the watchdog is deactivated.

If one of these conditions is not fulfilled the watchdog immediately restarts with a new period and signals this with a Watchdog Restart Interrupt (WDRI) if enabled. If this happens the watchdog is not stopped unless all mentioned stop-conditions are fulfilled and the watchdog is triggered again with the “watchdog off” code. If the watchdog overflows, the fail-safe SBC performs a system reset.

If the watchdog is not stopped due to high current consumption at V1 and if the V1 Current Monitor Control bit (V1CMC) within the Configuration Register is set, the fail-safe SBC immediately performs a reset at the end of the watchdog period. The reset source is then “V1 current above threshold”.

The fail-safe SBC continuous to observe the V1 current after the watchdog has been successfully deactivated. As soon as the current consumption rises above the undercurrent threshold, the fail-safe SBC either restarts the watchdog or performs a system reset. The reaction again depends on the V1CMC bit. If this bit is “0” the watchdog is restarted with the previous period. If the V1CMC bit is “1” the fail-safe SBC immediately performs a reset with the reset source “V1 current above threshold”.

5.5.3.5. Example: Cyclic wake-up via watchdog

In an application the fail-safe SBC should wake-up the microcontroller cyclically via interrupt with a period of about one second. After a short control loop using the LIN Bus the ECU should enter Standby Mode again. Program execution is stopped then until the next period elapses.

Following figure shows a possible flowchart for this software routine. The flowchart before stop mode of the microcontroller is shown on the left side of the figure and the flowchart after awaking from stop mode on the right side.

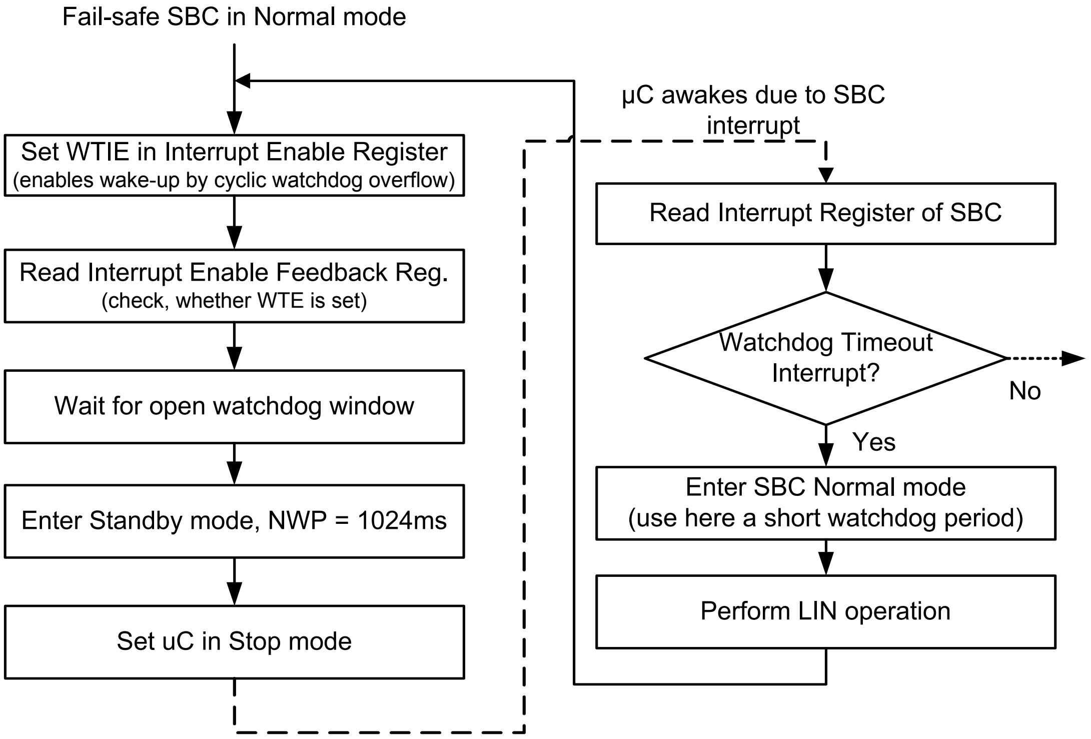

Fail-safe SBC in Normal mode
Set WTIE in Interrupt Enable Register
(enables wake-up by cyclic watchdog overflow)
Read Interrupt Enable Feedback Reg.
(check, whether WTE is set)
Wait for open watchdog window
Enter Standby mode, NWP = 1024ms
Set uC in Stop mode
μC awakes due to SBC interrupt
Read Interrupt Register of SBC
Watchdog Timeout Interrupt?
No
Yes
Enter SBC Normal mode
(use here a short watchdog period)
Perform LIN operation

Figure 5-5: Example Flow : Cyclic wake-up out of Standby Mode

In order to prepare a cyclic watchdog wake-up via interrupt in Standby Mode the microcontroller has to enable the Watchdog Timeout Interrupt. After checking the correct settings in the Interrupt Enable Register and waiting for the next open watchdog window the microcontroller performs a transition to Standby Mode in the fail-safe SBC. A watchdog period of 1024ms is selected. Then program execution in the microcontroller is stopped.

The microcontroller awakes after an SBC interrupt. The microcontroller must read the Interrupt Register within a certain time in order to acknowledge this interrupt. If the interrupt was caused by a watchdog overflow the microcontroller performs its cyclic task. For this the Normal Mode of the fail-safe SBC is activated in order to exchange data via LIN. After this task has been completed the next low power cycle can be prepared. Since the fail-safe SBC can only change into Standby Mode when the watchdog window is open, it is recommended to use a short watchdog period in Normal Mode. This reduces the waiting period thus reducing the average power consumption.

5.6. Sleep Mode

The Sleep Mode is the second level low power mode of the System Basis Chip. As soon as the fail-safe SBC enters Sleep Mode the power supply for the microcontroller V1 is shut down. At the same moment the reset line is pulled to ground level safely. Thus the program execution is terminated immediately. So from the software's point of view there is no further action required within Sleep Mode. However, before the SBC enters Sleep Mode its behavior during and after Sleep Mode has to be configured.

5.6.1. CAN and LIN Status during Sleep Mode

The CAN and the LIN transceiver generally leave their Active Mode when the fail-safe SBC enters Sleep Mode independent from the setting of the CAN Mode Control and the LIN Mode Control bit (CMC, LMC) in the Physical Layer Control Register. If the CAN transceiver enters Off-line Mode, the supply V2 is also disabled except if the V2 Control bit (V2C) in the Physical Layer Control Register is set.

5.6.2. INH/LIMP during Sleep Mode

If the INH/LIMP pin is set on HIGH level (inhibit function, ILEN and ILC set to “1”) it is disabled as soon as the UJA106x enters Sleep Mode. That means that the ILEN bit is automatically set to “0” and the pin becomes floating. For reactivating this pin the ILEN bit must again be set to “1” after waking the system.

If the INH/LIMP pin is set to LOW level (limp home function, ILEN = “1” and ILC = “0”) the status of the pin remains unchanged during Sleep Mode. For further information see also 2.4.3 and [1] - [4].

5.6.3. EN behavior during Sleep Mode

In Sleep Mode the EN pin is always LOW.

5.6.4. V3 and WAKE pin behavior during Sleep Mode

The status of the V3 output and the WAKE pin configuration remains unchanged in Sleep Mode, see also [1] - [4]

5.6.5. Leaving Sleep Mode

The fail-safe SBC leaves Sleep Mode when a wake-up event is detected. A wake-up event can either be a wake-up via the CAN or the LIN bus, a negative edge at the WAKE-pin or a cyclic wake-up via watchdog. After the wake-up the fail-safe SBC performs a system reset turns on V1 and enters Start-up Mode. The wake-up source is reported via the RSS bits in the Status Register and for CAN, LIN and the WAKE pin additionally with the CWS, LWS and EWS bits. In case more than one wake-up occurred nearly at the same time the RSS bits show the first wake-up event. A wake-up via LIN and the WAKE pin is only possible if the according wake-up source is enabled by setting the WEN bit or the LWEN bit respectively.

It should be noted that Sleep Mode is immediately left if the CWS, the LWS or the EWS bit is set to “1”. This is also true if these bits were set to “1” before the Sleep Mode command was sent to the fail-safe SBC. This prevents that Sleep Mode is entered with a pending wake-up or with a “last-minute” wake-up. The fail-safe SBC requires that every wake-up event is acknowledged by reading the Status Register. If a Sleep Mode command is sent while one of the wake status bits is set, the SBC performs a reset and returns the wake-up event as reset source. From software’s point of view this looks similar to a wake-up out of Sleep Mode.

In case V3 is kept on in Sleep Mode the fail-safe SBC also observes this voltage source because it can be used for supplying important hardware components. If V3 is deactivated due to an overload condition the fail-safe SBC also awakes out of Sleep Mode in order to give the microcontroller the chance to react on this failure condition. The fail-safe SBC reports this event with the reset source bits in the Status Register.

5.6.6. Example: Entering Sleep Mode via Standby Mode

An ECU should be shut down in two steps. First the node is put into the Standby Mode for about two seconds in order to react quickly if there is a late wake-up event. If this is not the case the node should finally enter Sleep Mode. The waiting period before going into Sleep Mode should be realized with the cyclic wake-up feature of the fail-safe SBC.

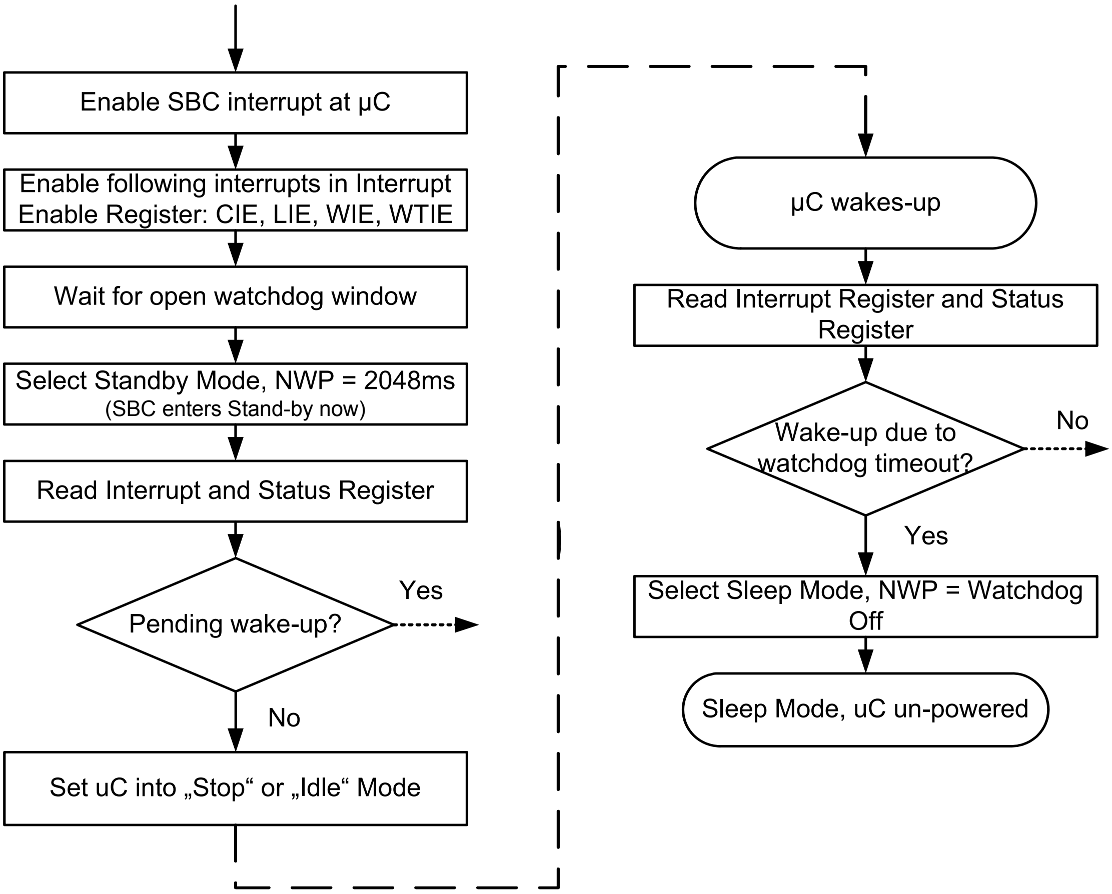

Enable SBC interrupt at µC
Enable following interrupts in Interrupt Enable Register: CIE, LIE, WIE, WTIE
Wait for open watchdog window
Select Standby Mode, NWP = 2048ms
(SBC enters Stand-by now)
Read Interrupt and Status Register
Pending wake-up?
Yes
No
Set uC into „Stop“ or „Idle“ Mode
µC wakes-up
Read Interrupt Register and Status Register
Wake-up due to watchdog timeout?
No
Yes
Select Sleep Mode, NWP = Watchdog Off
Sleep Mode, uC un-powered

Figure 5-6: Proposal for a flowchart for entering Sleep *Mode* via Standby *Mode*

A possible flow-chart for this routine is depicted in Figure 5-6. After the decision to enter low-power mode, first the interrupt input of the microcontroller and the wake-up interrupts for CAN, LIN and the WAKE pin (CIE, LIE and WIE) are enabled to ensure that a wake-up event during Standby Mode does not result in a system reset. Furthermore the Watchdog Timeout Interrupt (WTIE) is activated to allow a soft wake-up via interrupt when the watchdog overflows in Standby Mode. After that the microcontroller waits for the next open watchdog window and then writes the Operating Mode code for the Standby Mode and the desired watchdog period into the Mode Register. Since the microcontroller should awake after about two seconds the watchdog period 2048ms is chosen. After this command has been sent, the fail-safe SBC enters Standby Mode. Now the microcontroller checks whether there is a pending wake-up by reading the Interrupt Register and checking the wake status bits (CWS, LWS and EWS) in the Status Register. If this is not the case the microcontroller finally enters Stop mode.

The microcontroller awakes due to an interrupt. By reading the Interrupt and the Status Register it can check whether there is a wake-up via CAN, LIN or the local WAKE pin. If this is the case the power-down routine is left. But, if the interrupt was causes only by the Watchdog Timeout Interrupt (WTI) the microcontroller can be sure that the two-second waiting time is passed without a wake-up event. Now it can set the ECU into Sleep Mode by sending the Sleep Mode code together with the “watchdog off” code to the Mode Register.

5.7. Flash Mode

State of the art ECU’s in automotive applications allow updating of the FLASH memory via CAN during service of the vehicle in a garage. For this special purpose the fail-safe SBC offers a dedicated mode, the Flash Mode.

Flash Mode allows accessing the CAN and the LIN bus as in Normal Mode but with relaxed watchdog triggering. In Flash Mode the watchdog periods are similar to those in Standby Mode with the exception that the watchdog cannot entirely be deactivated. Furthermore during Flash Mode, the watchdog works permanently in timeout mode. That means that the trigger command can be sent at any time before the watchdog overflows. The relaxed timing requirements of the Flash Mode facilitate software design because it is often difficult to guarantee an exact timing during execution of a certain flash update routine.

5.7.1. Entering Flash Mode

For entering Flash Mode a fail-safe coded entry sequence has to be written into the Mode Register during Normal Mode. After successful transmission of this sequence the fail-safe SBC forces a system reset with the reset source information “SBC ready for entering Flash Mode” within the Status Register. Now the window for entering Flash Mode is open. Figure 5-7 shows a simple flow for this sequence. It shall be noted that the Configuration and the Physical Layer Control Register are not accessible in Flash Mode and therefore have to be configured before it is entered.

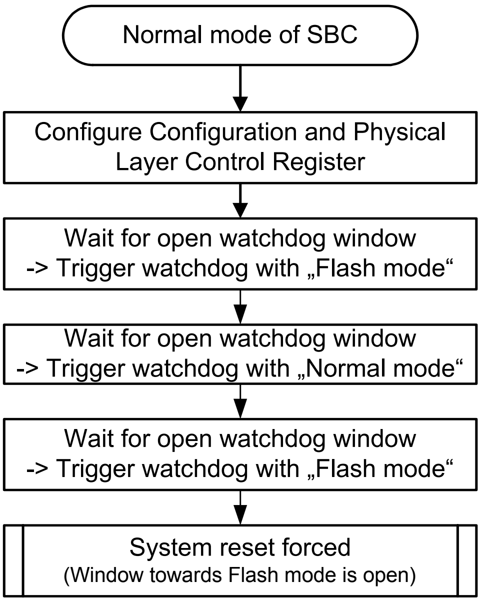

Normal mode of SBC
Configure Configuration and Physical Layer Control Register
Wait for open watchdog window -> Trigger watchdog with „Flash mode“
Wait for open watchdog window -> Trigger watchdog with „Normal mode“
Wait for open watchdog window -> Trigger watchdog with „Flash mode“
System reset forced
(Window towards Flash mode is open)

Figure 5-7: Preparing the SBC for Flash Mode

For initializing Flash Mode a special Operating Mode code is needed that has to be written into the Mode Register. It is only allowed to send this code when the fail-safe SBC is in Start-up Mode after a Flash Mode preparation reset. This fail-safe procedure ensures that the microcontroller has really seen the reset. At any other time this Flash Mode initialization code is forbidden and immediately results in an SPI-failure reset.

Once the fail-safe SBC is in Flash Mode, the microcontroller has to trigger the watchdog according to the selected period. The microcontroller has to use the Flash Mode code for these trigger commands.

If Flash Mode initialization fails, e.g. due to another reset or due to entering Normal Mode instead of Flash Mode, the window towards Flash Mode is closed. To open it again the complete Flash Mode preparation sequence has to be sent to the fail-safe SBC.

5.7.2. Leaving Flash Mode

Flash Mode is left upon every reset event. This can either be an external reset or an internal reset, e.g. a V1 under-voltage event or a watchdog overflow. The fail-safe SBC then goes into Start-up Mode. In order to give the microcontroller certainty during the following Start-up period that Flash Mode was not left due to an error the fail-safe SBC provides a special operating mode code for leaving Flash Mode. If this code is written into the Mode Register during Flash Mode a system reset is immediately performed and the reset source “SBC successfully left Flash Mode” is reported in the Status Register.

After Flash Mode has been left it is only possible to re-enter this mode by running through the entire entry sequence as described in 5.7.1 again.

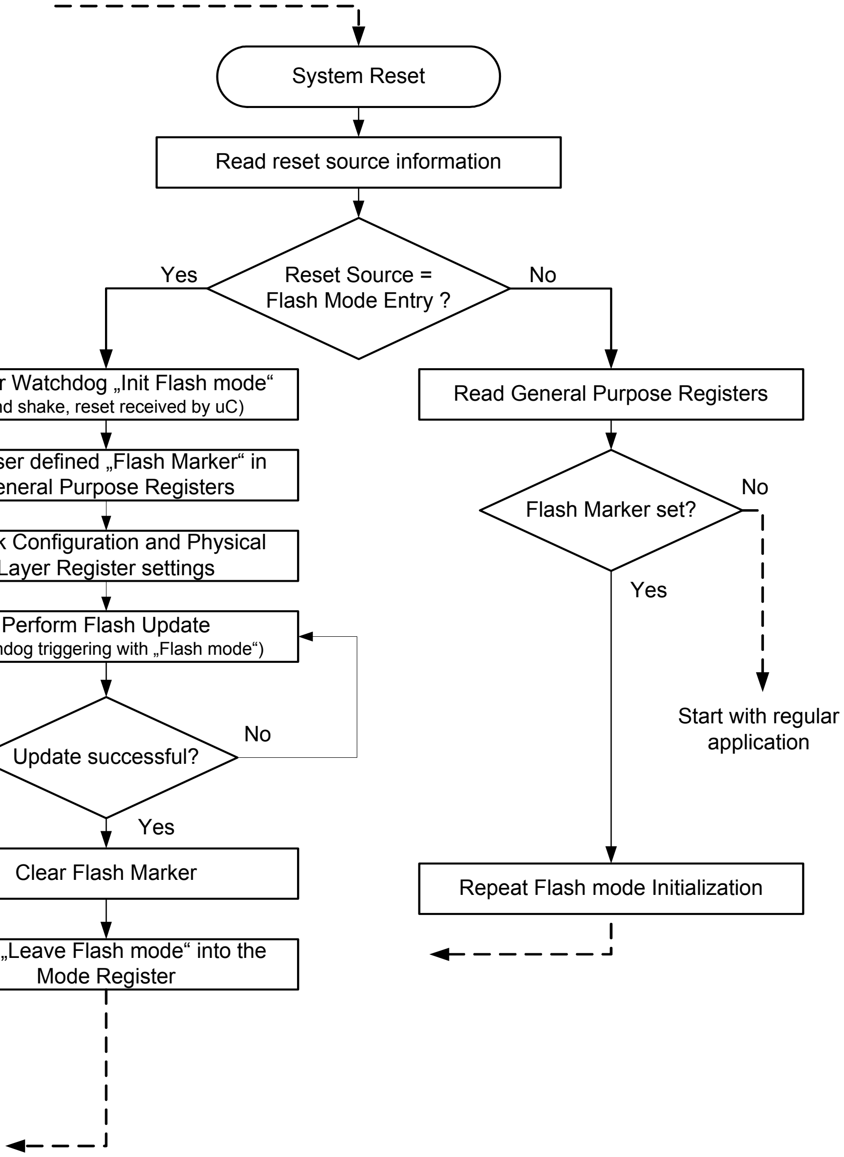

System Reset
Read reset source information
Reset Source = Flash Mode Entry ?
Yes
No
Trigger Watchdog „Init Flash mode“ (hand shake, reset received by uC)
Set user defined „Flash Marker“ in General Purpose Registers
Check Configuration and Physical Layer Register settings
Perform Flash Update (Watchdog triggering with „Flash mode“)
Update successful?
No
Yes
Clear Flash Marker
Write „Leave Flash mode“ into the Mode Register
Read General Purpose Registers
Flash Marker set?
No
Yes
Start with regular application
Repeat Flash mode Initialization

Figure 5-8: Simplified start-up routine when using Flash Mode

5.7.3. Behavior during Flash Mode

Flash Mode is characterized by the ability to access the CAN and the LIN bus as in Normal Mode but with a relaxed watchdog trigger requirement. The watchdog is working in timeout mode as it does in Standby Mode. This means that it can be triggered any time before it overflows. The watchdog periods are similar to those in Standby Mode. It is possible to select periods from 20ms up to 28s. But, it is not possible to disable the watchdog entirely according to car manufacturer requirements.

In Flash Mode the microcontroller has access to both General Purpose Registers. This allows storing important data for the next start-up, e.g. a marker that indicates that the flashing process was successful, as shown in Figure 5-8. The Configuration and Physical Layer Register are both not accessible in Flash Mode because they share their SBC addresses with those of the General Purpose Registers. That means that the configuration of the fail-safe SBC has to be done before entering Flash Mode. It is only possible to read back the information of these registers. Thus the microcontroller can read the information after initialization of Flash Mode in order to take care that all settings are correct before the flash download process starts.

5.8. Special modes for evaluation purposes

The fail-safe SBC provides two special modes that are intended for system evaluation and testing during hardware and software development. These modes are the Software Development Mode and the Forced Normal Mode. These modes are described in the following sections.

5.8.1. Software Development Mode

The Software Development Mode (SDM) is a special mode that can be used for simplifying software design with the fail-safe SBC. The properties of this mode are described in detail in the following section.

5.8.1.1. Behavior during Software Development Mode

"No Fail-Safe Mode"

Fail-Safe Mode cannot be entered. Instead always Restart Mode is re-entered. Only if V1 is permanently overloaded, Fail-Safe Mode is entered for self-protection.

The fail-safe SBC does not check whether the reset pin is clamped. This facilitates the start-up with an external microcontroller emulator because emulators often hold down the reset line during initialization process.

"No initialization timeout in Start-up or Restart Mode"

The 256ms initialization timeout in Start-up and Restart Mode is completely deactivated to simplify start-up of the system with an external microcontroller emulator.

"No watchdog triggering necessary in Normal Mode"

In Normal Mode the watchdog is still active but it does not cause a system reset if it overflows. The watchdog can be triggered at any time or not at all. However, if the watchdog is triggered too late or too early this is still displayed with the reset source bits in the Status Register.

If desired it is possible to signal every watchdog overflow by enabling the Watchdog Timeout Interrupt that is normally used in Standby Mode. This can be used for test purposes to check if the watchdog is triggered correctly without the risk of a system reset.

In Standby and Sleep Mode the functionality of the watchdog is the same as without SDM. Here a watchdog overflow causes a system reset. This allows using the cyclic wake-up feature of the fail-safe SBC.

"No interrupt monitoring"

The interrupt monitoring timeout is completely deactivated. If there is an interrupt the INTN pin of the fail-safe SBC goes down until the Interrupt Register is read.

5.8.1.2. Entering Software Development Mode

For fail-safe reasons the Software Development Mode can only be entered directly after activating the battery supply at pin BAT42. There are two possibilities to enable this special mode, via the TEST pin or via SPI. To activate SDM via the TEST pin a certain voltage must be applied at this pin before powering-up BAT42 as described in 2.10. This option has to be used for activating SDM without running software. The second way to activate SDM is by setting the ISDM bit within the Special Mode Register after power-on as long as the UJA106x is in Start-up Mode.

To stay in SDM the SDM bit has to be set to “1” each time the software writes into the Mode Register. The SDM is immediately deactivated as soon as this bit is written with “0”. Thus it is possible to start-up the system with SDM because an emulator is used and to leave this mode later to see if the software is running correctly even without SDM. Once SDM is left it is not possible to re-enter this mode unless a power-on is performed.

For fail-safe reasons it is recommended to always write a “0” into the SDM bit within the Mode Register in the final software. Thereby Software Development Mode is immediately left even if it was erroneously activated.

5.8.1.3. Example: Using the Software Development Mode

Which commands must be sent to the SBC to activate the Software Development Mode via SPI?

To activate SDM the following SPI command has to be sent directly after powering-up the device: 0100 0010 0000 0000 = 0x4200. This command writes the Special Mode Register and sets the ISDM bit. Afterwards the fail-safe SBC is in Start-up Mode. Since the start-up timer is disabled there are no timing constrains for initializing the watchdog and entering Normal Mode. To keep Software Development Mode activated must be taken care of that the SDM bit always has to be set when writing into the Mode Register. The following command initializes the Normal Mode with a watchdog period of 80ms and activates the EN-Pin: 0000 1101 1010 1110 = 0x0DAE.

5.8.2. Forced Normal Mode

The Forced Normal Mode is a static test mode that can be used for evaluation purposes during development and for first programming of the microcontroller in production. In Forced Normal Mode all supplies (V1, V2 and V3) of the fail-safe SBC are on and the CAN and LIN transceiver are in Active Mode. The SPI interface is deactivated. Thus it is not possible to configure the UJA106x. The watchdog is not running.

Forced Normal Mode is entered by applying a certain voltage at pin TEST before powering the fail-safe SBC (see 2.10). If this is done correctly, the fail-safe SBC directly goes into this special mode after performing a power-on reset. There is no need for further actions to be taken by software.

6. The CAN transceiver

6.1. Introduction

The UJA106x fail-safe SBC family provides derivatives with fault-tolerant CAN as well as with high-speed CAN transceiver. Due to the family approach both kinds of transceiver have the same state diagram and the same settings. Therefore all descriptions in this chapter are valid for both CAN transceivers unless it is explicitly mentioned.

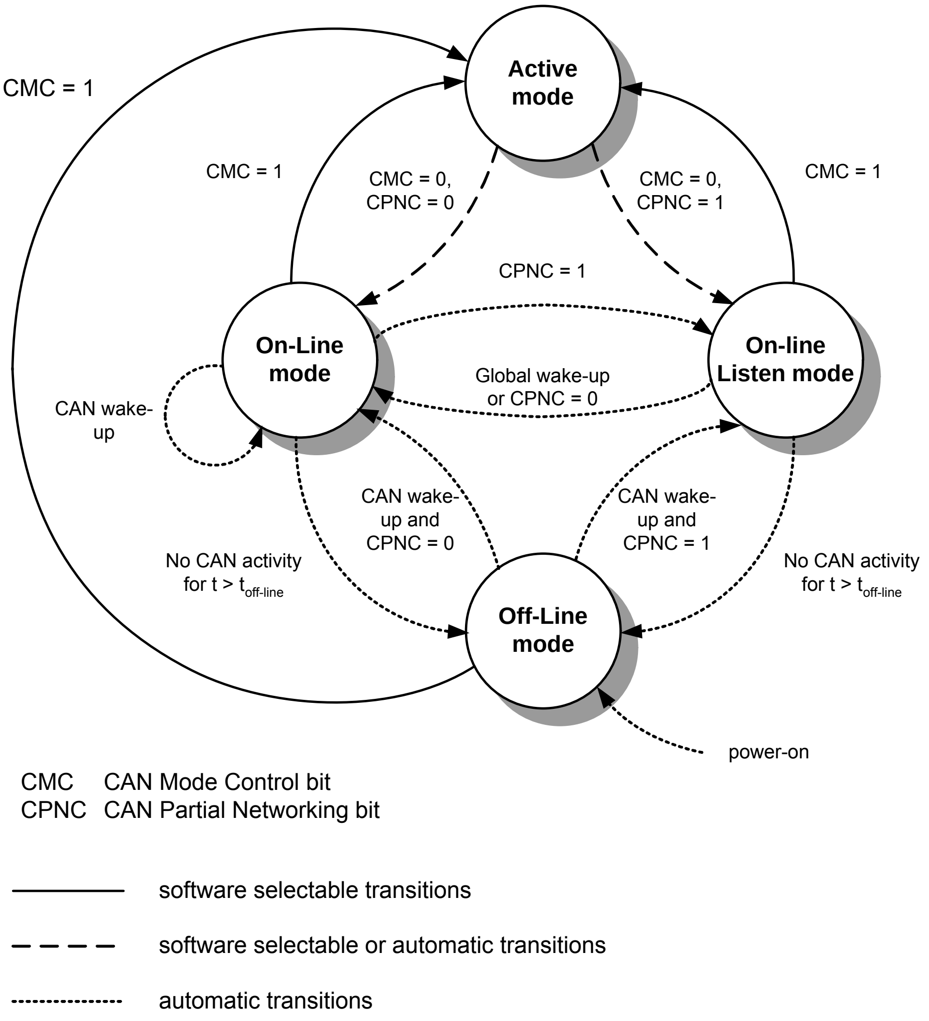


Figure 6-1: State diagram of the CAN Transceiver

6.2. CAN Modes

The state diagram of the CAN transceiver within the fail-safe SBC family consists of four states as shown in Figure 6-1. The Active Mode is the normal operating mode of the CAN transceiver. This is the only mode that allows transmitting and receiving data via CAN. The Off-line Mode is the low power mode of the CAN transceiver. In this mode the CAN transceiver has the lowest power consumption because only the low power wake-up receiver and the low power bus termination are supplied. The transmitter, the Normal Mode receiver and the Normal Mode bus termination are deactivated. V2 is off.

Between Active and Off-line Mode there are two intermediate states, the On-line and the On-line Listen Mode. They allow the fail-safe SBC to autonomously support the physical medium even if the microcontroller is unable to do anything. This has the advantage that the defect node does not affect the functionality of the remaining network. Therefore in both states V2 is on and the Normal Mode CAN termination is supplied, but the CAN transmitter and receiver are off. Instead the low power receiver is active to receive wake-up messages.

On-line and On-line Listen Mode differ in the way they interpret a CAN wake-up message. Which mode is entered depends on the CAN Partial Networking Control bit CPNC in the Physical Layer Control Register. If On-line Mode is active any CAN message, that fulfils a certain timing-pattern, results in a wake-up. In On-line Listen Mode only a certain bit-pattern that can be generated by special CAN messages performs a CAN wake-up [1], [2] and [4]. This mode can be used for partial networking when a few nodes are active and communicate with each other while the other nodes are sleeping, see 6.6.

The state-diagram in Figure 6-1 shows three kinds of lines between the CAN states, solid, dashed and dotted. The solid lines mark state transitions that can only be performed by the software. The dashed lines identify transitions that can be controlled by the software and depending of the circumstances automatically. The dotted ones are transitions that cannot be influenced by the software. These transitions are only performed automatically.

6.2.1. Active Mode of CAN

Active Mode has to be selected for sending and receiving data via CAN. The power supply V2 is generally enabled to supply the transceiver. For the CAN transceiver of the fail-safe SBC to enter Active Mode two conditions have to be fulfilled. The CAN Mode Control bit (CMC) in the Physical Layer Register has to be set to “1” and the SBC has to be either in Normal or Flash Mode. Active Mode is immediately left as soon as either condition is not met anymore, e.g. when Normal Mode is left due to a system reset.

6.2.1.1. Listen-only Mode of CAN

The fail-safe SBC allows disabling the CAN transmitter in Active Mode so that it is working as a listen-only device. For this the CAN Transmitter Control bit (CTC) must be set to “1” in the Physical Layer Control Register. If this bit is set, no data fed into TXDC is forwarded to the CAN bus anymore. However the CAN receiver is still working.

For test purposes it is possible to bypass the TXDC signal to the RXDC pin while the transmitter is off. This allows the CAN controller to see its own signal when the transceiver is working in Listen–only Mode. The bypass is activated when the CAN Receiver Control bit (CRC) in the Physical Layer Control Register is set to “1”.

6.2.2. On-line Mode of CAN

On-line Mode is entered when Active Mode is left or when a wake-up signal is detected on the CAN bus. It is intended to support the bus termination and the bus failure management autonomously, even if the microcontroller is unable to do anything. It is a fail-safe function that ensures that the overall CAN bus performance is not affected by a defect node. Another important advantage is that the physical condition on the CAN wires immediately switches to the 5V based termination after a wake-up. There is no extra delay until the CAN part becomes active, caused by the software start of the application. This allows a cleaner start-up of the CAN communication without disturbances due to “late” ECU’s in the system or even "dead" ECU's.

The power supply V2 for the CAN transceiver is always enabled in this mode. The CAN transmitter and receiver are always off. Hence it is not possible to send or receive any messages. Instead any CAN event fulfilling the wake-up conditions causes a CAN wake-up (see. 6.4.2).

On-line Mode is an intermediate state that is automatically left after there has been bus silence for a certain time. This time is called the Off-line Time. After this time has elapsed, the CAN transceiver changes into Off-line Mode. There are two different situations that have to be distinguished:

1. Leaving Active Mode

When the application should enter low power mode, the Active Mode of the CAN transceiver is left. After that there should be silence on the bus. The CAN transceiver stays in On-line Mode until the Off-line Time has elapsed. During this time the Active Mode termination and the bus failure management is supported. Wake-up events on the bus are immediately detected.

2. CAN wake-up

A CAN wake-up event that is detected while not in Active Mode always causes the CAN transceiver to enter On-line Mode. This wake-up is reported via the Status Register and via the RXD_C pin.

The fail-safe SBC provides two settings for the Off-line Time. The selection is done with the CAN Off-line Timer bit (COT) within the Physical Layer Control Register. However, when On-line Mode is entered due to a CAN wake-up, a fixed long Off-line Time is applied independent from the setting of the COT bit in order to give the application enough time to start-up without changing the CAN bias condition back to off-line behavior too early.

6.2.3. On-line Listen Mode of CAN

The On-line Listen Mode is mainly intended to support partial networking within a vehicle. It is entered by setting the CAN Partial Networking bit CPNC in the Physical Layer Register. During On-line Listen Mode, conventional CAN messages are not forwarded to the system anymore and thus the local ECU may enter a low-power mode without being waked-up by the ongoing CAN communication of other nodes.

The CAN transceiver stays in On-line Listen Mode as long as it detects CAN communication on the bus. The supply V2 is on and the Active Mode termination is enabled in order to take care that the physical conditions on the bus are the same as if all nodes were active.

When CAN communication stops, the Off-line Timer starts running (see 6.2.4). After the selected Off-line Time has elapsed without any further bus traffic, the CAN transceiver goes into Off-line Mode. The timer is reset with any event on the CAN bus. The CAN transceiver also re-enters On-line Listen Mode from Off-line Mode if CAN communication is started again as long as the CPNC bit is set. During this transition no CAN wake-up is detected. Thus the state of the fail-safe SBC is not affected. If it is e.g. in Sleep Mode it stays in Sleep Mode without taking notice of the bus traffic.

To wake-up the fail-safe SBC via CAN while in On-line Listen Mode a special pattern has to be sent on the CAN bus ([1], [2] and [4]). If the transceiver detects this pattern it immediately enters On-line Mode and performs a CAN wake-up. That means that the fail-safe SBC leaves Sleep Mode or performs an interrupt or a wake-up reset while in Standby Mode. The RXD_C pin is then LOW and the CWS bit within the Status Register is set.

The wake-up pattern can be generated by two special CAN messages. These are the Initial message and the Global wake-up message ([1], [2] and [4]). The Global wake-up message causes the CAN wake-up. This message must be received within the time $t_{timeout}$ ([1], [2] and [4]) after the Initial message or after the CAN transceiver changed from Off-line to On-line Listen Mode. That means in practice that one has to distinguish two different cases:

1. The fail-safe SBC should be waked during running CAN communication

Since the CAN transceiver is in On-line Listen Mode both messages have to be sent, the Initial and the Global wake message.

2. The entire system is in low power mode and there is no communication

In this case the CAN transceiver is in Off-line Mode. Then only the Global wake-up message is sufficient to wake-up the fail-safe SBC.

For fail-safe reasons the CPNC bit is automatically cleared with certain conditions. These are:

1. A CAN wake-up is detected
2. Active Mode is entered
3. V2 is shut down due an overload
4. Fail-safe Mode is entered

Therefore it should be kept in mind that for re-entering On-line Listen Mode the CPNC bit must always be set again after a CAN wake-up has been received or before leaving Active Mode.

6.2.4. Off-line Mode of CAN

Off-line Mode is the state of the CAN transceiver with the lowest power consumption. It is automatically entered after a power-on reset or when Active Mode is left and the Off-line timer has elapsed.

In Off-line Mode the CAN transceiver is not supplied. Hence the transmitter and the receiver are disabled. Only the wake-up receiver is waiting for CAN activity. The Active Mode termination is off and the CAN wires are terminated according to the low power behavior of the dedicated CAN system (fault-tolerant or high-speed CAN). In derivatives with fault-tolerant CAN the low-power bus failure management is active.

Since CAN is not supplied in Off-line Mode V2 is normally off. However, if V2 is also used for supplying other components (which is not recommended, while CAN is actively used!) it can be decided to keep this voltage source on. For this the V2 Control bit (V2C) within the Physical Layer Control Register must be set to “1”.

Off-line Mode is left when Active Mode is entered by user command or when a conventional CAN pattern is detected as defined in [1], [2] and [4] or 6.4.2. Depending on the setting of the CPNC bit in the Physical Layer Register the CAN transceiver in the latter case either enters On-line or On-line Listen Mode. Note that any conventional CAN wake-up pattern in general is necessary for leaving Off-line Mode.

6.3. Power supply for CAN

One major difference between the fail-safe SBC and a discrete solution is that the microcontroller and the CAN transceiver have separated power supplies. In discrete solutions often one voltage regulator is used for both. Here the software designer can ignore on what happens if the CAN supply voltage brakes down because then the microcontroller is also stopped. But, when using the UJA106x it should be noted that CAN is not automatically supplied whenever the microcontroller is running. There are two situations that have to be considered, the activation of CAN and a CAN supply failure.

6.3.1. Activating CAN

When Off-line Mode is left due to a CAN wake-up or Active Mode should be entered, V2 is generally activated. When V2 is switched from off to on, it takes some time until it has reached the target value because the capacitor at pin V2 has to be charged. This time directly correlates with the size of the capacitor. As long as V2 is below the under voltage detection threshold $U_{det(UV)(V2)}$ ([1], [2] and [4]) the CAN Mode Diagnosis bits CANMD in the Diagnosis Register show that CAN is still in Off-line Mode. This allows to read back whether CAN is properly supplied. In order to avoid that mutilated signals appear on the bus, it is recommended not to start any CAN transmission before the CANMD bits stand on “Active Mode”. This indicates that CAN is ready for participating in CAN traffic (see also 6.3.2).

6.3.2. V2 failure

If V2 is activated but stays below the V2 undervoltage detection threshold $U_{det(UV)(V2)}$ for the time $t_{2(CLT)}$ ([1], [2] and [4]) it is disabled again, the V2 Diagnosis bit (V2D) in the Diagnosis Register is cleared and a Voltage failure interrupt (VFI) is signaled, if enabled. At the same moment the CAN Transmitter Control bit (CTC) is set to “1” and the V2 Control flag (V2C) and the CAN Partial Networking bit (CPNC) are set to “0”. Due to the setting of the CTC bit the CAN transmitter is disabled. **It should be noted that the CTC bit must be cleared by software in order to resume CAN communication.** This bit is never cleared automatically even if V2 is working again (see also 4.1.3).

A V2 failure usually happens if the voltage breaks down due to an overload, e.g. a shortcut caused by a damaged capacitor. It is also possible that a V2 failure is detected because of a low battery voltage. This can happen if e.g. the battery supply slowly ramps up. Since the latter case is only a temporary failure it is very important to provide a failure recovery routine in the software. Otherwise the CAN communication might permanently be stopped after such situations.

For restarting V2 after a failure there are several possibilities:

- clearing the CTC bit in the Physical Layer Register
- setting the V2C bit in the Physical Layer Register
- leaving and entering Active Mode again

Furthermore V2 is also reactivated whenever a CAN wake-up is detected in Off-line Mode.

After reactivation of V2 the V2D bit in the Diagnosis Register is immediately set to “1”. V2 stays on for at least $t_{2(CLT).}$ If the voltage is still below the under voltage threshold after this time, V2 is again deactivated and a failure is reported via the V2D bit. Because of the minimum time in that V2 stays on after reactivation it should be noted that permanent recovery of a V2 failure might heat up the device in case of a real hardware failure. Therefore it is recommended to stop V2 failure recovery after a certain number of unsuccessful attempts.

As mentioned above the CAN transmitter is disabled by automatically setting the CTC bit in case of a V2 failure. This should ensure that no corrupted CAN signals appear on the bus. By clearing the CTC bit V2 and the CAN transmitter are reactivated. Although this is a very simple way for recovering this failure it might be better to execute it in two steps, especially in safety critical applications:

First reactivate V2 and once V2 is active again, the CAN transmitter should be reactivated.

For this procedure the V2 failure should be recovered by leaving and re-entering Active Mode. When Active Mode is left the content of the CANMD bits in the Diagnosis Register switch from Active Mode to Off-line Mode[5]. If Active Mode is re-entered V2 is restarted. After some time either the CANMD bits show “Active Mode” or the V2D bit is cleared again. If the CANMD bits stay on “Active Mode” V2 is obviously stable again. Then the CAN transmitter can be reactivated by clearing the CTC bit. Otherwise the V2 failure is still there and therefore the CAN transmitter should stay off.

In case of a soft shortcut at pin V2 it is possible that the voltage only breaks down during active transmission because then the regulator has to deliver the highest current. In this case only corrupted messages are transmitted. This is a very disturbing situation for the rest of the network. Because of this, a successful V2 failure recovery has to include that at least one error-free message has been transmitted. That means that if an error counter is installed for counting unsuccessful V2 restart attempts this counter shouldn’t be cleared before the first message has been transmitted successfully.

6.4. CAN wake-up

6.4.1. Reaction upon a CAN wake-up

The fail-safe SBC is sensitive to CAN wake-ups whenever the CAN transceiver is not in Active Mode. That means that either the fail-safe SBC is in another mode than Normal or Flash Mode or that the CMC bit in the Physical Layer Register is set to “0” (see also 6.2.1). It is not possible to entirely disable the CAN wake-up in order to prevent a system state without any of the wake-up sources active anymore (coma-mode).

Whenever a CAN wake-up has been received the RXD_C pin goes down and the CAN Wake Status bit (CWS) is set in the Status Register. Further reactions of the fail-safe SBC upon a CAN wake-up depend on the mode of the SBC.

CAN wake-up in Start-up or Restart Mode

No further reactions.

CAN wake-up in Normal or Flash Mode

CAN wake-ups are only detected if the CMC bit in the Physical Layer Register is set to “0”. Additionally a CAN wake-up interrupt can be forced.

CAN wake-up in Standby Mode

Either a system reset or an interrupt is performed. If the CAN Interrupt (CANIE) is enabled an interrupt is caused. If not, a system reset is performed with the reset source “CAN wake-up event”.

CAN wake-up in Sleep and Fail-safe Mode

A CAN wake-up in Sleep and Fail-safe Mode causes the fail-safe SBC to turn on V1 and to enter Start-up Mode after performing a system reset. When coming from Sleep Mode the reset source is “CAN wake-up

[5] Note: If V2 fails while CAN is in Active mode, only the V2D bit is cleared. The CANMD bits stay on Active mode.

event”. When the SBC was in Fail-safe Mode before the reset source “wake-up out of Fail-safe Mode” is reported. In that case the wake-up source can only be determined via the CWS bit.

Note that after receiving the CAN wake-up in Fail-safe Mode the 1.5s retention timer starts first. The wake-up is performed after the retention time t_{ret} has elapsed ([1], [2] and [4]) in order to guarantee a properly discharged V1 regulator buffer capacitor before the system re-boots.

6.4.2. Definition of the CAN wake-up message

As described in [1], [2] and [4] it has to be distinguished between a conventional and a global wake-up pattern. A conventional wake-up is defined by a dominant-recessive-dominant transition with a certain timing as known from stand-alone transceivers like the TJA1054/55/41. A global wake-up is a more complex pattern that can be generated with special CAN messages (dedicated data content using any CAN identifier). Which of those wake-up patterns results in a CAN wake-up depends on the setting of the CAN Partial Networking bit (CPNC) in the Physical Layer Register. If this bit is “0” any conventional CAN wake-up pattern results in a wake-up like with stand-alone transceivers. If this bit is “1” a conventional wake-up pattern just causes that CAN enters On-line Listen Mode without a CAN wake-up. That means e.g. that the fail-safe SBC stays in Sleep Mode.

It should be noted that for leaving Off-line Mode, independent of the setting of the CPNC bit, a conventional wake-up pattern is needed. If the CAN transceiver is in On-line Listen Mode it is sensitive to the global wake-up pattern. If this is the case the CAN transceiver immediately goes from On-line Listen to On-line Mode and causes a CAN wake-up.

6.5. Failure diagnosis with the CAN transceiver

The CAN Transceiver of the UJA106x family provides extensive failure detection mechanisms for both systems, fault-tolerant CAN and high-speed CAN. That means that the fail-safe SBC delivers detailed information about detected failures on the bus wires and on the microcontroller interface. In case of a fault-tolerant CAN transceiver the bus termination is also handled autonomously according to the ISO 11898-3 standard.

6.5.1. Bus failure diagnosis with fault-tolerant CAN

Bus failures affect one or both wires of the CAN bus. They typically are visible for all participants in the system. The CAN transceiver in the UJA1061 provides detailed bus failure information in the CANFD bits within the Diagnosis Register. As soon as a failure is detected or recovered it is displayed in the CANFD bits. Additionally any change of the CANFD bits can be signaled with an interrupt. For this the CAN Failure Interrupt (CANFIE) must be enabled.

For all fail-safe SBCs with fault-tolerant CAN (UJA1061) the following bus failures are distinguished:

- Interruption of CANH
- Interruption of CANL
- CANH clamped to its recessive level
- CANL clamped to its recessive level
- CANH clamped to its dominant level
- CANL clamped to its dominant level
- CANH is shorted to battery
- CANL is shorted to battery
- CANH is shorted to CANL
- CAN bus is clamped recessive (dual failure)
- CAN bus is clamped dominant (dual failure)

It shall be noted that some of the failures are only detectable or distinguishable during active transmission. Since the fail-safe SBC shows a failure as soon as it is detected it is possible that at first a wrong failure is reported that looks similar to the real one. As soon as both failures can be distinguished the display switches to the correct one. The following bus failures are only detected during transmission:

- CANH or CANL clamped to their recessive level
- CANH is shorted to CANL

6.5.1.1. Example: Recessive clamped wire detection within a 3-Node Network

A simple network consists of three nodes A, B, C as depicted in Figure 6-2. At the beginning there are no failures in the network. After some time one wire is shorted to the recessive level. Node A then sends some data. After that node B is responding. Node C is listening all the time. What are the different fail-safe SBCs within the nodes A, B and C reporting during the bus communication?

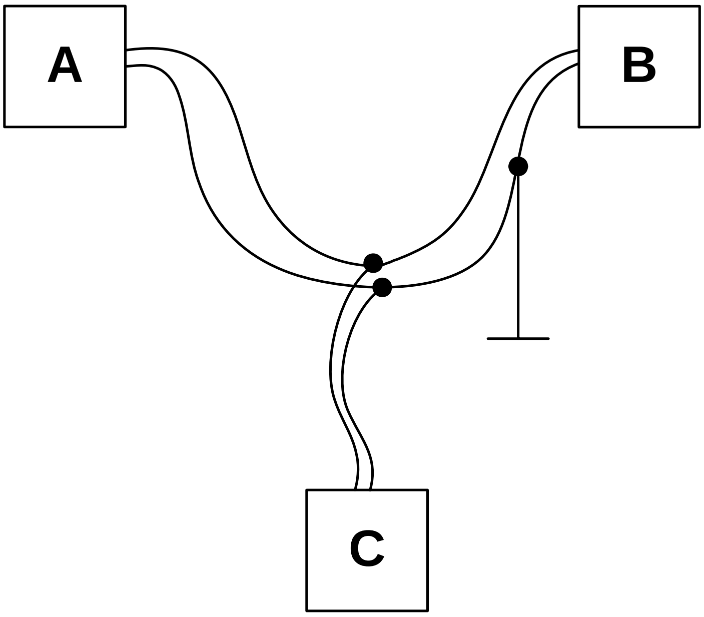


Figure 6-2: Simple network with interruption of one line

Table 6-1 summarizes the reported failures of each node. All nodes start with no failure. When node A is sending the first time node B and C observe that one wire stays recessive while receiving data. Thus both nodes report an interruption failure on this wire. Since A is sending, this node can directly see that one wire is clamped. This node immediately displays that this wire is clamped recessive.

When node B starts to transmit node A and node C observe a missing signal on one wire. Node A already knows that one wire is clamped recessive. Here the display keeps stable. For node C nothing changes. Node B in contrast now sees that the wire is not interrupted but clamped recessive. Here the displayed bus failure changes.

Table 6-1: Failure report of the nodes in the example

Action	Node A	Node B	Node C
Before failure	No failure	No failure	No failure
Node A is sending	Bus wire recessive	Bus wire interrupted	Bus wire interrupted
Node B is sending	Bus wire recessive	Bus wire recessive	Bus wire interrupted

6.5.2. Bus failure diagnosis with high-speed CAN

Similar to fault-tolerant CAN transceivers the high-speed CAN transceiver of the UJA1065 and UJA1066 also provides detailed failure diagnosis on the bus wires. The high-speed CAN transceiver can distinguish the following bus failures:

- CANH is shorted to VCC or battery
- CANL is shorted to VCC or battery
- CANH is shorted to GND
- CANL is shorted to GND
- CANH is shorted to CANL
- CAN bus is clamped dominant (dual failure)

The bus failures are only distinguishable during active transmission. Hence it is possible that during reception of a signal at first a wrong bus failure or no bus failure is displayed. As soon as the fail-safe SBC transmits itself, the CAN failure diagnosis information shows the correct bus failure [6].

It should be noted that for high-speed CAN transceivers, bus failure diagnosis is only possible in Active Mode. While not in Active Mode the CAN failure display is frozen.

6.5.3. Diagnosis of TXD-RXD failures

The UJA106x is able to detect failures at the CAN interface to the microcontroller. For CAN (fault-tolerant as well as high-speed) the following failures can be detected:

- TXDC is continuously clamped dominant
- RXDC is continuously clamped dominant
- RXDC is continuously clamped recessive

The failures are also displayed with the CANFD bits in the Diagnosis Register. They have a higher priority than the bus failures. So if there is a bus failure and a TXD-RXD failure at the same time, the TXD-RXD failure is reported.

In case of a TXDC dominant or RXDC recessive failure, the CAN transmitter is disabled to prevent the rest of the network from being disturbed. The transmitter is automatically reactivated as soon as the failure is gone and the CANFD bits switch back to “no failure”. For automatically recovering these failures it is necessary that they are obviously gone when receiving something from another node. This has the advantage that bus communication is not disturbed even if there is a shortcut between TXDC and RXDC[7]. This failure normally results in total loss of communication on the entire CAN bus.

It has to be kept in mind that automatic recovery of these failures is not possible as long as there is no communication on the bus, e.g. when all nodes are sleeping. Hence the transmitter stays disabled. In order to allow sending a wake-up message it is necessary to force a failure recovery by software. This is done by setting and clearing the CAN Transmitter Control bit (CTC) in the Physical Layer Register. Therefore two SPI accesses are necessary. When doing this the CANFD bits switch back to “no failure” and the CAN transmitter is enabled again. If now the failure is really gone, it is possible to send the wake-up signal. **Note that it is highly recommended to include such a recovery routine in the software especially if the affected node is usually supposed to wake-up the network**. Otherwise an erroneous detection of these failures might result in a system hang-up.

[6] The CAN failure diagnosis mainly checks for asymmetries on the bus and maps them to the CANFD bits. Especially in small networks it is possible that an asymmetry that is caused by one node or by GND shift is misinterpreted as physical bus failure.

[7] This requires that the RXDC pin of the fail-safe SBC is stronger than the TXD pin of the microcontroller. This can be guaranteed if the TXD pin is configured as open-drain output with pull-up resistor.

If a failure recovery is forced while the failure is still present, the fail-safe SBC detects it again and disables the transmitter. In order not to disturb the network communication forced failure recovery should be stopped after some unsuccessful tries. This is commonly realized with a software error counter.

6.5.4. GND shift diagnosis

All fail-safe SBCs with CAN interface, fault-tolerant as well as high-speed CAN, provide a GND shift diagnosis feature. The SBC is able to detect if the GND level of the ECU differs from that of other CAN nodes. The current status of the GND shift diagnosis is displayed with the Ground Shift Diagnosis bit (GSD) in the Diagnosis Register. Every change of this bit can additionally be signaled with an interrupt. (This interrupt underlies the interrupt limitation as described in 9)

The GND Shift diagnosis is refreshed every time the CAN bus is recessive for about 80µs while the CAN transceiver is in active mode. A noise gate reduces EMC related disturbances. Whenever the CAN transceiver is not in active mode GND shift diagnosis is not active and the according bit is frozen.

The Recessive bus-level is used as reference to detect a Ground shift. There is a slight difference between Ground shift detection for FT-CAN and HS-CAN with respect to Measuring Levels. For FT-CAN the recessive level of the CANH signal is directly measured. For HS-CAN the recessive level of CANH is compared to +2,5V.

The fail-safe SBC provides two different GND shift thresholds, a normal and an increased one which can be selected via the Threshold Control bit (GSTHC) in the Configuration Register. The default setting after power-on is the normal threshold. It is recommended to use this threshold for first detection of GND shift. If then a Ground Shift is detected, the wider threshold can be used to find out whether there is a very deep GND shift.

It should be noted that GND shift diagnosis is only working if there is no bus failure condition present. It is possible that with certain bus failures the GSD bit is set, too. Therefore the GND shift diagnosis should be ignored whenever a bus failure is detected.

6.5.5. Emulation of Error Pin functionality (NERR)

Stand-alone transceiver with error detection provide a dedicated error pin (NERR) that goes to the LOW level as soon as a bus failure is detected, e.g. the fault-tolerant CAN transceiver TJA1054/55 or the high-speed CAN transceiver TJA1041. Here normally no detailed information about the failure is available. However, for many applications this information is sufficient. To allow an easy replacement of the stand-alone transceiver by a fail-safe SBC the UJA106x provides a so-called Error Emulation Mode. In this mode the EN-pin simulates the functionality of an error pin. That means that this pin is only HIGH when the CANFD bits show no failure. As soon as a CAN failure is detected the pin is set to LOW.

The Error Emulation Mode is activated by setting the Error-pin Emulation Mode bit (ERREM) in the Special Mode Register to “1”. This is only possible in Start-up or Restart Mode.

6.6. Partial Networking in a CAN System

Partial networking deals with situations, in which parts of the network are communicating while other nodes stay within their low-power mode. Normally a node would wake-up as soon as it detects traffic on the bus. For partial networking it is necessary that the down powered nodes do not wake-up upon normal bus traffic. However, it is of course desired that waking-up the nodes is still possible somehow. The sleeping nodes should only awake after a special wake message is received. Since the microcontroller and the CAN controller are inactive, only the fail-safe SBC itself is able to detect this special wake message. For such applications the On-line Listen Mode is provided that is activated by setting the CAN Partial Networking bit (CPNC) within the Physical Layer Register; see also 6.2.3.

If the CPNC bit is set, the fail-safe SBC does only awake by CAN if a certain pattern is detected that can be generated with the Initial and the Global wake-up messages ([1], [2] and [4]). The wake-up via LIN and via the WAKE pin is unaffected and still possible. Partial networking is realized by setting the CPNC bit in all nodes that should not participate at CAN communication anymore. Then these nodes can enter low-power mode. They do not wake-up due to conventional CAN traffic, even if the other nodes also enter low-power mode and wake-up again after receiving a standard CAN wake-up message.

For partial networking it is important that those nodes that are always active are prepared for sending the special wake-up message patterns if necessary. Otherwise the nodes with the set CPNC bit stay in low-power mode until there is a power-on or a Wake-pin or LIN wake-up.

The Initial and the Global wake-up messages are identical for all UJA106x derivatives. Hence all devices wake-up on receiving these messages. It is not possible to wake-up one individual node only. Instead all nodes will wake up based on the Global wake-up message pattern. After that global wake-up all nodes, which are not needed during the following communication period would have to be set into partial networking mode again via the CPNC bit, if desired.

6.7. Differences between high-speed CAN and fault-tolerant CAN

6.7.1. General differences

High-speed CAN and fault-tolerant CAN differ from each other in bit rate and the ability to handle single bus failures. High-speed CAN transceivers allow bit rates up to 1Mbit/s. In general it cannot be guaranteed that they can continue communication if one bus wire is damaged, e.g. shorted to GND. The physical interface consists of three pins, the CAN wires CANH and CANL and the SPLIT pin; see 2.6.2.

Fault-tolerant CAN transceivers are able to maintain communication if one bus is shorted to GND, V_{CC} or battery supply or if both bus wires are shorted to each other. Communication is then done with one bus wire. Detection of the bus failure and switching to single-wire communication is autonomously done according to the ISO11898-3 standard. The bit rate is limited to 125kBit/s. The CAN levels are different to those of high-speed CAN and the CAN termination is different. The UJA106x derivatives with fault-tolerant CAN have the termination pins RTH and RTL in addition to the CAN wires CANH and CANL.

One major advantage of the UJA106x family is that apart from the CAN physical interface all other pins as well as the register mapping is identical for all derivatives. So the software is nearly identical for both CAN variants. This allows easy an exchange of the CAN physical layer, from fault-tolerant CAN to high-speed CAN and vice-versa.

6.7.2. SPLIT Pin of High-Speed CAN

The SPLIT pin in the high-speed CAN derivatives is intended for stabilizing the recessive level of the CAN bus. This allows reduction of the electromagnetic emissions. Hence it is recommended to use this pin. Ultimately the usage of the SPLIT depends on the carmaker’s demands.

The SPLIT pin is controlled by the CAN Split Control bit (CSC) in the Physical Layer Register. If this bit is set to “1” the recessive bus level is provided at the SPLIT pin and can be used according to the split termination concept. Otherwise the pin is floating. The SPLIT pin is supplied out of V2. Hence the pin is off (floating) if V2 is not available, e.g. if CAN is in Off-line Mode or if V2 is disabled due to an overload. Note that SPLIT is also off if V2 is activated via the V2C bit in the Physical Layer Control Register while CAN is in Off-line Mode.

7. The LIN transceiver

7.1. Introduction

Some derivatives of the UJA106x family contain a fail-safe LIN transceiver that can be used in any LIN or SAE J2602 network. The transmitter is designed according to the LIN2.0 requirements and thus, offers optimum system performance in any of the existing LIN systems (e.g. LIN1.2 or LIN1.3). Fail-safe means that the LIN transceiver prevents the LIN bus from disturbances if possible and it prevents the battery from being discharged in case of a failure. For this the LIN transceiver has a dedicated termination pin RTLIN. Furthermore it is possible to select the low slope function of the driver in order to reduce electromagnetic emissions at lower bit rates. The LIN transceiver also supports "Cooling" applications that require higher driver strengths than specified in the LIN 2.0 standard.

7.2. LIN modes

Figure 7-1 shows the state diagram of the LIN transceiver. It consists of two states, Active Mode and Off-line Mode. Similar to the CAN Active Mode, the LIN Active Mode is only entered if the fail-safe SBC is either in Normal or Flash Mode and additionally the LIN Mode Control bit (LMC) in the Physical Layer Register is set to "1". Otherwise LIN is in Off-line. The state diagram distinguishes between transitions that can be influenced by software, automatic transitions and transitions that can either be influenced by software or automatically.

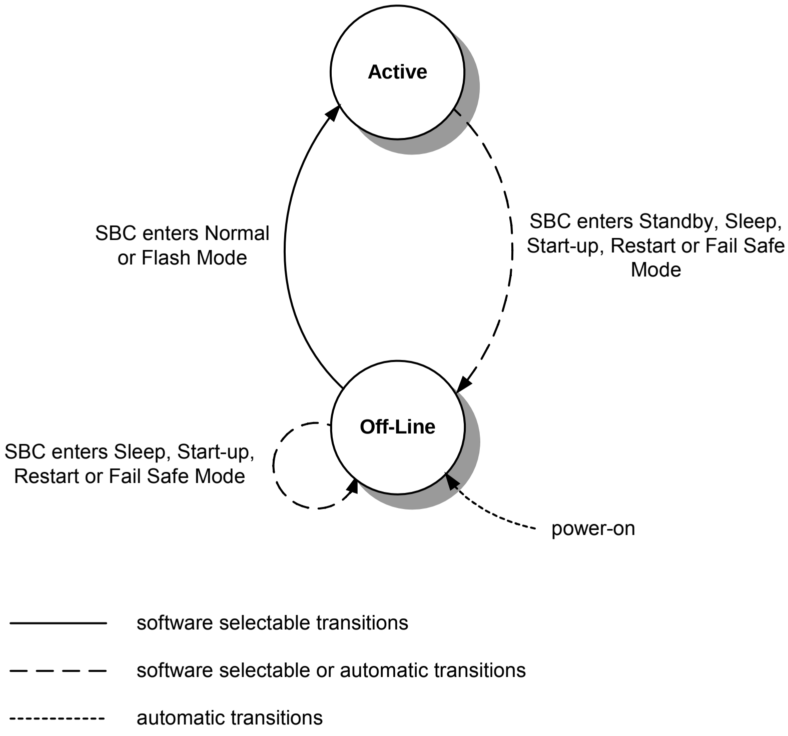

Active
SBC enters Normal or Flash Mode
SBC enters Standby, Sleep, Start-up, Restart or Fail Safe Mode
Off-Line
SBC enters Sleep, Start-up, Restart or Fail Safe Mode
power-on
software selectable transitions
software selectable or automatic transitions
automatic transitions

Figure 7-1: State diagram of the LIN Transceiver

7.2.1. Active Mode of LIN

Active Mode is necessary for transmitting and receiving data via LIN. The LIN transceiver of the fail-safe SBC enters Active Mode when both conditions are true, the LIN Mode Control bit (LMC) in the Physical Layer Register is set to "1" and the device is either in Normal or Flash Mode. If either condition is not true, e.g. when Normal Mode is left due to a system reset, the transceiver immediately switches into Off-line Mode.

7.2.1.1. Slope control of LIN

LIN is intended for bit rates up to 20kBit/s. However, in some applications only a bit rate of 10kBit/s is applied, especially in applications that accord to the American Standard SAE J2602. Here the maximum allowed bit rate is 10.4kBit/s. For such application the LIN transceiver provides a Low Slope Mode in order to reduce electromagnetic emission according to the LIN2.0 specification. The Low Slope Mode is enabled by setting the LIN Slope Control bit (LSC) in the Physical Layer Register.

7.2.1.2. "Cooling" applications with enhanced driver strength

Some climate applications require an increased LIN driver current capability for the configuration of special non-LIN conform slave nodes. The LIN transceiver of the UJA106x family even supports this feature by increasing driver strength. This is done by setting the LIN Driver Control bit (LDC) in the Physical Layer Control Register to "1". In order to reduce electromagnetic emission it is recommended to set the LDC bit to "0" once the initialization of the slaves has been completed.

7.2.1.3. Listen-only Mode of LIN

The fail-safe SBC allows disabling the LIN transmitter in Active Mode so that it works as a listen-only device. For this the LIN Transmitter Control bit (LTC) has to be set to "1" in the Physical Layer Control Register. If this bit is set, no data fed into TXDL is forwarded to the LIN bus anymore. Meanwhile the LIN receiver is still working.

7.2.2. Off-line Mode of LIN

In Off-line Mode the power consumption is lower than in Active Mode. In this state it is not possible to transmit or receive data. Only the low-power receiver is active waiting for LIN wake-up signals. The Active Mode termination is also off. The LIN wire is terminated with reduced current strength.

7.3. LIN wake-up

7.3.1. Reaction upon a LIN wake-up

The fail-safe SBC is sensitive to LIN wake-ups whenever the LIN transceiver is in Off-line Mode. That means that either the fail-safe SBC is outside Normal or Flash Mode or the LMC bit in the Physical Layer Register is set to "0" (see also 7.2.1).

Whenever a LIN wake-up was received the RXDL pin goes down and the LIN Wake Status bit (LWS) is set in the Status Register. Further reactions of the fail-safe SBC upon a LIN wake-up depend on the mode of the SBC. The behavior is similar to that of a CAN wake-up.

LIN wake-up in Start-up or Restart Mode

In these modes there are no further reactions.

LIN wake-up in Normal or Flash Mode

LIN wake-ups are only detected if the LMC bit is set to "0" in the Physical Layer Register. It is possible to additionally cause a LIN wake-up interrupt.

LIN wake-up in Standby Mode

Either a system reset or an interrupt is performed. If the LIN Interrupt (LINIE) is enabled an interrupt is caused. If not, a system reset is performed with the reset source "LIN wake-up event".

LIN wake-up in Sleep and Fail-safe Mode

A LIN wake-up in Sleep and Fail-safe Mode causes the fail-safe SBC to turn on V1 and to enter Start-up Mode after performing a system reset. When coming from Sleep Mode the reset source is "LIN wake-up event". When the SBC was in Fail-safe Mode before the reset source "wake-up out of Fail-safe Mode" is reported. In this case the wake-up source is only determined by the LWS bit. Note that after receiving the LIN wake-up in Fail-safe Mode first the 1.5s retention timer starts. The wake-up is performed after the retention time t_{ret} has elapsed ([1], [2] and [3]) in order to guarantee a properly discharged V1 buffer capacitor before the system re-boots.

It should be noted that after a LIN wake-up the RXDL pin stays low until the LWS bit in the Status Register is read or LIN has been set into Active Mode. As long as RXDL stays low no further LIN wake-up can be received.

7.3.2. Definition of the LIN wake-up message

For a LIN wake-up a recessive-dominant-recessive transition with defined timing according to the LIN2.0 standard has to appear on the LIN bus. Details can be found in [1], [2] and [3].

7.3.3. Deactivation of the LIN wake-up

The fail-safe SBC allows to disable the LIN wake-up entirely by setting the LIN Wake-up Enable bit (LWEN) in the Physical Layer Register to "0". Then absolutely no LIN wake-up is detected. LIN communication in Active Mode is not affected. The default setting of the LWEN bit after power-on is "1" so that LIN wake-up is active.

Note that the pure LIN derivative, the UJA1069, does not allow disabling the LIN wake-up because it is the main wake-up source. This prevents this wake-up source from being accidentally deactivated, which might result in a deadlock (coma-mode without any active wake-up source).

7.4. LIN termination

The LIN transceiver of the UJA106x family provides a fail-safe LIN termination with the pin RTLIN. The state diagram of the RTLIN pin can be found in [1], [2] and [3]. The RTLIN termination prevents the battery from being discharged when the LIN bus wire is shorted to GND. In this case RTLIN disables the strong LIN termination and activates a weak 75µA termination when LIN is in Active Mode or deactivates the LIN termination entirely if LIN is in Off-line Mode.

According to the RTLIN state diagram in [1], [2] and [3] the strong LIN termination is always reactivated when LIN enters Active Mode. This allows checking whether the shortcut is still there. If this is the case, the LIN termination is deactivated again after the LIN dominant detection time $t_{LIN(dom)(det)}$. Beside this, the LIN termination is working completely autonomously.

Note: In case there is a weak shortcut of the LIN wire to GND, e.g. caused by a leakage current, it is possible that LIN transmission is working without problems. But due to the weak 75µA current source in Off-line Mode the LIN wire is pulled to GND. Hence the LIN dominant failure is detected and the LIN termination is disabled. As soon as Active Mode is re-entered the strong LIN termination that pulls the bus wire back on the recessive level is enabled again. Then the LIN dominant failure is recovered. Conclusion: If a LIN dominant failure is only detected in Off-line and not in Active Mode this is an indicator that there is an unexpected leakage current between the LIN bus wire and GND.

Of course the usage of the RTLIN pin is not a strict obligation. If desired, the LIN transceiver can also be terminated with a diode and a resistor directly to the battery supply (see also 2.7).

7.5. Failure Diagnosis with the LIN Transceiver

Similar to the integrated CAN transceiver the fail-safe SBC provides detailed information about failures on the LIN bus. The detected failures are displayed with the LIN Failure Diagnosis bits (LINFD) in the Diagnosis Register. A LIN Failure Interrupt can additionally signal every change of the LINFD bits. The LIN transceiver can distinguish the following failures:

- LIN is clamped dominant, e.g. due to a short of the LIN wire to GND
- LIN is clamped recessive, e.g. due to a short of the LIN wire to battery
- TXDL is clamped dominant, e.g. due to a wrong programming of the corresponding port pin or a hardware failure

The TXDL dominant failure has the highest priority. So if there is a bus failure and a TXDL dominant failure at the same time, the TXDL dominant failure is reported.

In case TXDL is clamped dominant or the LIN wire is clamped recessive the LIN transmitter is internally disabled to prevent that the rest of the network is disturbed if the affected node is slave. The transmitter is automatically reactivated as soon as the failure is gone and the LINFD bits switch back to “no failure”. For automatically recovering these failures it is necessary that they are obviously gone when receiving something from another node. This has the advantage that bus communication is not disturbed even if there is a shortcut between TXDL and RXDL[8]. This failure normally results in total loss of communication of the entire LIN bus also at a slave node.

It has to be kept in mind that automatic recovery of these failures is not possible if there is no communication on the bus, e.g. when all nodes are sleeping or if the affected node is the master. So the transmitter stays disabled. In order to allow sending a wake-up message or a synch-break to the bus it is necessary to force a failure recovery by software. This is done by setting and clearing the LIN Transmitter Control bit (LTC) in the Physical Layer Register. So two SPI accesses are needed. When doing this the LINFD bits switch back to “no failure” and the LIN transmitter is re-enabled. If the failure is really gone it is again possible to send data.

[8] This requires that the RXDL pin of the fail-safe SBC is stronger than the TXD pin of the microcontroller. This can be guaranteed if the TXD pin is configured as open-drain output with pull-up resistor.

Note that it is strongly recommended to include such a recovery routine in the software especially if the affected node is the master or if it usually wakes-up the network. Otherwise an erroneous detection of these failures might result in a system hang-up.

If a failure recovery is forced in a slave while the failure is still there the fail-safe SBC detects it again and disables the transmitter. In order not to disturb network communication, forced failure recovery should be stopped after some unsuccessful attempts. This is commonly realized with a software error counter.

7.6. K-Line Interfacing

Besides supporting a LIN sub-bus, the integrated LIN transceiver of the UJA106x family can also be used for driving a K-Line interface. The following paragraphs are showing some constraints, if the fail-safe SBC should be used within a K-Line application.

7.6.1. Temperature parameters

Description	**ISO 9141**	**LIN**
Temperature Range	0...50 °C	-40...125 °C

From environmental temperature point of view there is no incompatibility.

7.6.2. Timing parameter

Description	**ISO 9141**	**LIN**
Transmission Rate	10.4 kbit/s	≤ 20kbit/s
Slope Time	< 10% of Bit Time	Slew Rate = about 13us => 13% of Bit Time @ 10 kbit/s & 12V

Due to the slope control function of the LIN transceiver, the 10% requirement cannot be fulfilled. Nevertheless this will not have a negative impact on the communication since the bit rate tolerance on the K-Line interface is much smaller than +/- 2% between the communicating nodes.

7.6.3. Capacitance:

Description	**ISO 9141**	**LIN**
Diagnostic Tester / Master	< 2 nF	< 2.5 nF
Wiring	< 2 nF	< 40 m * 150 nF => 6nF
ECU / Slave	< 500 pF	< 250 pF
Total	< 9.6 nF	< 10 nF

From capacitance point of view the LIN Transceiver can be adapted without problems. The master is simply applied with less than 2nF. The LIN restriction on slave side allowing not more than 250pF is not relevant for functionality. It might be increased without trouble as long as the total bus capacitance does not increase above the maximum allowed level of 10nF.
It shall be mentioned that the LIN transceiver operates with optimum performance, if the total time constant of the bus system does not increase about 5us. The time constant can be calculated by multiplying the total capacitance of the bus with the overall pull-up resistance of the network including all connected nodes.

7.6.4. Resistance:

Description	**ISO 9141**	**LIN Vers. 1.2**
Diagnostic Tester / Master	510 Ohm	1 kOhm
ECU / Slave	> 100 kOhm	20...47 kOhm

It is no problem for the LIN transceiver to be applied with an external 510 Ohms resistor in order to support the K-Line termination resistance.

The slave termination resistance is the only parameter of the LIN physical layer, which does not fit to the definition of the K-Line interface. However in contrast to stand-alone LIN transceivers the integrated LIN transceiver of the UJA106x family does not contain an internal termination and the 100k slave termination resistance can therefore be realized. The RTLIN pin can simply be applied with 100k and a diode to the LIN bus line. Hence the UJA106x is the only LIN device, which fits the K-Line specification regarding the slave termination resistance.

8. Software Interface

8.1. Overview

The software interface is the communication interface between the application software on the microcontroller and the fail-safe SBC. Software control is provided by the two channels

- SPI-Interface and
- Interrupt-line

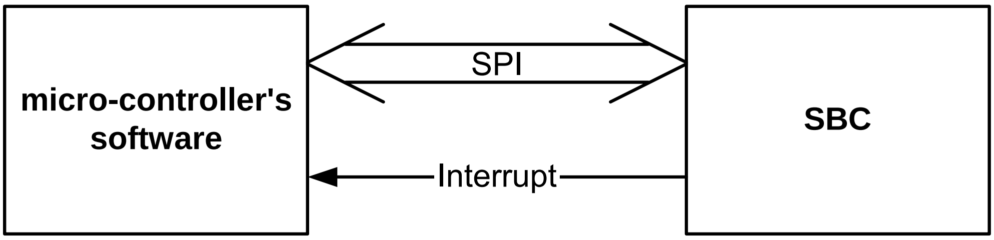


Figure 8-1: Software-Interface between software and SBC

The SPI-Interface is the main communication channel between the microcontroller and the fail-safe SBC. The microcontroller uses SPI for configuring the SBC, triggering the watchdog and receiving information about the current status of the system. The Interrupt connection is used to inform the software about events in the network or in the periphery of the SBC.

This section shows how these software interfaces are working in general.

8.2. Configuration of the SPI-Interface

The fail-safe SBC uses a 16-bit SPI-Interface. SPI transmissions with exact 16 Bit (clock cycles on the SCK line while SCS is LOW) are accepted. If more or less bits are sent the SBC performs an interrupt or a reset, depending on the programming of the device.

Figure 8-2 shows how the SPI-Interface has to be configured. The fail-safe SBC samples data on the falling edge of the pin SCK and shifts them with the rising edge. The transmission begins with the most significant bit.

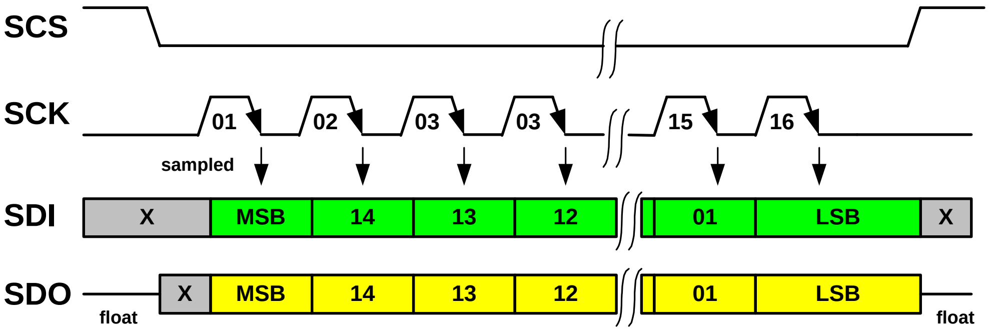


Figure 8-2: Configuration of the SPI-Interface

8.3. Registers of the fail-safe SBC

8.3.1. Register Architecture

An SPI access is a bidirectional data transfer. As one bit is written into SDI another bit is shifted out of SDO as depicted in Figure 8-2. The upper four bits of the 16-bit message determine which register is addressed and whether something is written or not. Hence these four bits are always mirrored at SDO. Figure 8-3 shows the address structure of the fail-safe SBC.

The most significant two bits are the address bits. Behind each address there is one register for writing and two registers for reading. Which of these registers is actually read depends on bit 13, the Read Register Select bit (RRS). So with the upper three bits it is determined which register is selected for writing and which one for reading. Whether something is actually written into the write register depends on bit 12, the Read Only bit (RO). If this bit is set to “1” the SPI transfer is a read-only access. In this case bit 0 to 11 that are written into SDI are ignored. If the RO bit is “0” bit 0 to 11 are written to the write register.

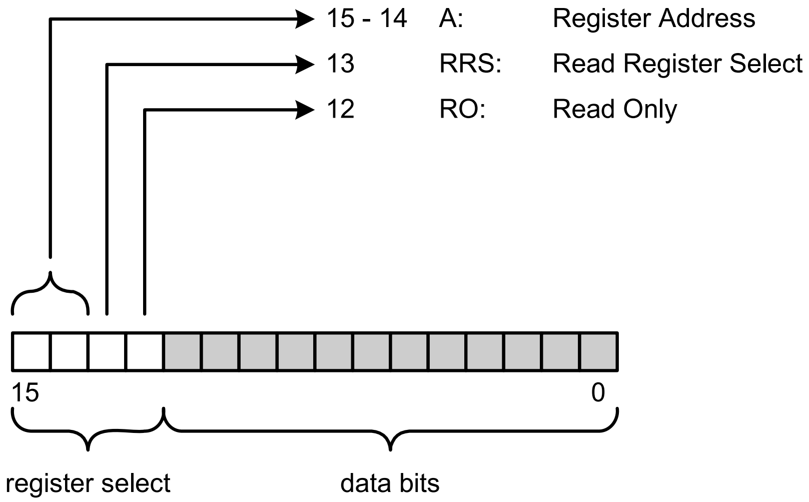


Figure 8-3: Structure of the SBC registers

8.3.2. Register mapping

Table 8-1 shows the four **Register Groups** of the UJA106x in detail. Each group contains one write and two read registers. It should be noted that some registers share the same address. For those registers it depends on the state of the fail-safe SBC which register actually is accessed.

Table 8-1: Register mapping of the UJA106x family

Register Group	Addressing Bits				Write to Register	Read from Register
	15	14	13	12		
	ADR0	ADR1	Read Register Select	Read Only		
A	0	0	0	0	Mode Register	System Status Register
			1			System Diagnosis Register
			0	1	No write access	System Status Register
			1			System Diagnosis Register
B	0	1	0	0	Interrupt Enable Register / Special Mode Register [9]	Interrupt Enable Feedback Reg.
			1			Interrupt Register / Special Mode Feedback Register
			0	1	No write access	Interrupt Enable Feedback Reg.
			1			Interrupt Register / Special Mode Feedback Register
C	1	0	0	0	System Configuration Register / General Purpose Register 0 [10]	System Config. Feedback Reg.
			1			General Purpose 0 Feedback Reg.
			0	1	No write access	System Config. Feedback Reg.
			1			General Purpose 0 Feedback Reg.
D	1	1	0	0	Physical Layer Control Register / General Purpose Register 1 [7]	Physical Layer Control Feedback
			1			General Purpose 1 Feedback Reg.
			0	1	No write access	Physical Layer Control Feedback
			1			General Purpose 1 Feedback Reg.

8.4. SPI failure handling

The fail-safe SBC always expects a 16-bit transmission at one SPI access. If at the moment the CSC pin goes HIGH more or fewer than 16 bits (clock cycles) have been received an SPI clock count failure is detected. The reaction upon this failure depends on its state. In Start-up and Restart Mode a system reset is performed with the reset source “illegal SPI access” in order to take care that initialization is done correctly. In Normal, Standby or Flash Mode only an SPI failure interrupt is caused, if enabled.

[9] The Special Mode Register is only accessible in Start-up and Restart mode.

[10] The two General Purpose Registers are only writeable in Start-up, Restart and Flash mode. This allows using these registers for some failure statistic features described later within this report. These registers can always be read.

9. Interrupt System

Besides SPI the interrupt system of the fail-safe SBC provides another software interface to the microcontroller. It reports events concerning the device itself or the bus system. In case the related interrupt sources are activated within the Interrupt Enable Register and the fail-safe SBC operates in Normal, Flash or Standby Mode, each interrupt event sets the dedicated bit within the Interrupt Register. This causes the INTN pin to go to LOW level. The pin stays low until the Interrupt Register is read. In the meantime all interrupt events are captured and displayed in the Interrupt Register. After reading the Interrupt Register is cleared and the INTN pin is released.

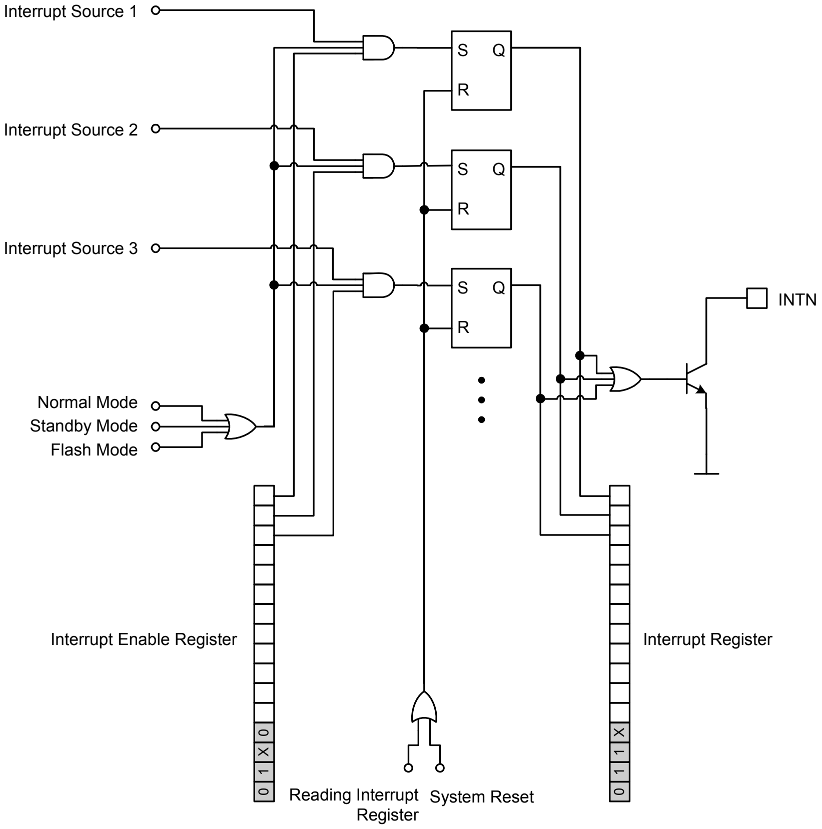


Figure 9-1: Block diagram of the interrupt system

When the INTN pin goes down an interrupt monitoring timer starts until the Interrupt Register is read. If this timer overflows a system reset is caused with the reset source “Interrupt not served within $t_{RSTN(INT)}$”. This is a fail-safe feature that allows detection of an interrupted connection between the interrupt pins of the microcontroller and the SBC.

In order to prevent that software execution is permanently disturbed by interrupts, the fail-safe SBC provides an interrupt limitation feature. Each "informative interrupt" can only cause one interrupt per watchdog period while some "system critical interrupts" are always signaled immediately without limitation. If an informative interrupt of the same source occurs again within the same watchdog period, the corresponding bit in the Interrupt Register is set as usual, but it does not cause the INTN pin to go LOW. After the next valid watchdog trigger, these stored interrupts are signaled to the INTN output pin.

The interrupt limitation is only implemented for Normal Mode, since here the application software might be heavily affected by interrupt overloading. In Standby or Flash Mode each enabled interrupt is signaled to the INTN output pin without limitation. The interrupt limitation does not affect the SPI clock count failure interrupt and the BAT failure interrupt, because these are regarded to be "system critical interrupts". These interrupts indicate a serious system problem and are therefore not limited. The interrupt limitation feature takes care that the software cycle time is foreseeable, which is very important when a watchdog must be triggered in a certain window. Beside this the important SPI and BAT failure interrupts cannot be missed even if there is a high number of interrupts of other interrupt sources.

If the interrupt limitation feature is not wanted in an application, it can be bypassed by writing to the interrupt enable register after processing of each interrupt. A write access to the interrupt enable register will reset the interrupt limitation.

The Interrupt Register is cleared and the INTN pin is released when the fail-safe SBC performs a system reset. In Start-up and Restart Mode the SBC does not perform any interrupts. This ensures that no old interrupts stay pending after a reset.

A Appendix

A.1 Register Mapping & Derivative Differences

The Fail-Safe System Basis Chip family includes five derivatives. Based on the family approach the main design of all derivatives is in principal equal. This means that all derivatives have the same 32-pin package, pin order and functionality as well as the same register set and hence require the same software control for proper operation.

The only differences between those five derivatives are caused by the different combination of implemented CAN and LIN physical layer blocks.

Table A-1 shows the SBC's register mapping. It shall be noted that not all bits are used in all derivatives. While most bits are the same for all SBCs, some bits exist only in specific derivatives. The Bits marked with a "C" exist only in derivatives with a CAN-transceiver (e.g. UJA1061, UJA1065 and UJA1066) while all Bits marked with "L" exist only in derivatives that include a LIN-tranceiver (e.g. UJA1061, UJA1065 and UJA1069).

In each derivative, those Bits that are not used, always return "0" when read and stay unaffected by Write Accesses.

Based on the combination of physical layer blocks, the current consumption of the derivatives differs. Evidently a UJA1069 derivative with only the LIN physical layer included consumes less current than a UJA1065 derivative that includes both, a LIN and a CAN physical layer block. Detailed current consumption characteristics can be found in the according data sheets.

The electromagnetic compatibility (EMC) and the electrostatic discharge (ESD) characteristics of the specific physical layer implementations are comparable for the CAN Fault-Tolerant derivative (UJA1061), the CAN High-Speed derivatives (UJA1065, UJA1066) and the LIN derivatives (UJA1061, UJA1065, UJA1069). All specific physical layer implementations are re-used in the complete family to show equal behavior.

From functional point of view two minor differences have to be stated. First the coding of the CAN Failure Diagnosis bits in the System Diagnosis Register of the CAN derivatives have a slight different function. The Fault-Tolerant CAN SBCs offer a more detailed failure diagnosis compared to the High-Speed CAN SBCs. Please refer to the relevant data sheets for detailed register information.

Secondly the LWEN bit (LIN Wake-up Enable) in the Physical Layer Control Register of the UJA1069 SBC is slightly different. Since it always has to be possible to wake-up the SBC in a low power mode from at least one wake source, the LWEN bit can not be set to '0' and hence the LIN wake-up cannot be disabled by software control in the UJA1069.

From pinning point of view all derivatives are available in 32-pin packages. Since the UJA1069 includes only a LIN physical layer block, this SBC is also available in a 24-pin package for those applications, in which more focus is taken on a small board space than on SBC replace ability.

For distinguishing the SBC derivatives a device specific ID can be read from the General Purpose Register 0 after start-up. Please refer to the relevant data sheets for detailed register information.

[1]

Table A-1 - Register Mapping

Bit	Mode Register	System Status Register		System Diagnosis Register		Interrupt Enable Register		Interrupt Register	
11	**NWP.5**	**RSS.3**		**GSD**	C	**WTIE**		**WTI**	
10	**NWP.4**	**RSS.2**		**CANFD.3**	C	**OTIE**		**OTI**	
9	**NWP.3**	**RSS.1**		**CANFD.2**	C	**GSIE**		**GSI**	
8	**NWP.2**	**RSS.0**		**CANFD.1**	C	**SPIFIE**		**SPIFI**	
7	**NWP.1**	**CWS**	C	**CANFD.0**	C	**BATFIE**		**BATFI**	
6	**NWP.0**	**LWS**	L	**LINFD.1**	L	**VFIE**		**VFI**	
5	**OM.2**	**EWS**		**LINFD.0**	L	**CANFIE**	C	**CANFI**	C
4	**OM.1**	**WLS**		**V3D**		**LINFIE**	L	**LINFI**	L
3	**OM.0**	**TWS**		**V2D**	C	**WIE**		**WI**	
2	**SDM**	**SDMS**		**V1D**		**WDRIE**		**WDRI**	
1	**EN**	**ENS**		**CANMD.1**	C	**CANIE**	C	**CANI**	C
0	**-**	**PWONS**		**CANMD.0**	C	**LINIE**	L	**LINI**	L

Bit	System Config. Register		Physical Layer Control Register		Special Mode Register		General Purpose Register 0	General Purpose Register 1
11	**-**		**V2C**	C	**-**		**DIC**	**GP1.11**
10	**-**		**CPNC**	C	**-**		**GP0.10**	**GP1.10**
9	**GSTHC**	C	**COTC**	C	**ISDM**		**GP0.9**	**GP1.9**
8	**RLC**		**CTC**	C	**ERREM**	C	**GP0.8**	**GP1.8**
7	**V3C.1**		**CRC**	C	**-**		**GP0.7**	**GP1.7**
6	**V3C.0**		**CMC**	C	**WDPRE.1**		**GP0.6**	**GP1.6**
5	**-**		**CSC**[11]	C	**WDPRE.0**		**GP0.5**	**GP1.5**
4	**V1CMC**		**LMC**	L	**V1RTHC.1**		**GP0.4**	**GP1.4**
3	**WEN**		**LSC**	L	**V1RTHC.0**		**GP0.3**	**GP1.3**
2	**WSC**		**LDC**	L	**-**		**GP0.2**	**GP1.2**
1	**ILEN**		**LWEN**	L	**-**		**GP0.1**	**GP1.1**
0	**ILC**		**LTC**	L	**-**		**GP0.0**	**GP1.0**

C – Only in CAN-Derivatives **L** – Only in LIN-Derivatives

[11] Only in HS Can Derivatives

A.2 Design checklist

This section is intended to give the user of the fail-safe SBC an overview of design issues that should be considered in an application. Several hardware and software issues are briefly described in a table. Details can be found in the referring chapters.

The design checklist should point out several hard- and software issues that should be attended in design. That does not mean that every point can really be applied to a certain application or that every point absolutely has to be considered. But we recommend to carefully go through the list and to check if the described situation might result in a problem in the specific application.

The design checklist has to be understood as an open list that can permanently be extended if new ECU design issues come up. Because of this completeness of the list cannot be guaranteed.

A.2.1 Hardware design issues

No.	Item	Description	Checked
H01	Power supply	The filter capacitors on BAT42, BAT14, V1 and V2 should be as close as possible at the SBC.	
H02	Power supply	The BAT inputs of the fail-safe SBC require external polarity protection diodes. If very low battery supply situations need to be handled, a split polarity protection with separate buffering capacitors for the pins BAT42 and BAT14 is recommended; see 2.1 and 2.2.	
H02a	Power supply	If a split polarity protection is used as described in H02 – an additional diode across Bat14 and Bat42 is needed see 2.2.	
H03	SPI	Take care that there is a proper SPI communication between the microcontroller and the SBC. Any transmission problems, e.g. spikes on any interface signal might cause the SBC to enter Fail Safe Mode.	
H04	INT-Pin	The Interrupt-Pin of the SBC is an open drain output and requires external pull-up behavior; see 2.5.	
H05	V3	It is recommended to apply a series resistor within V3 line if it is directly routed outside of the ECU; see 2.8.	
H06	WAKE	It is recommended to apply a series resistor within the WAKE line, if it is directly routed outside of the ECU, see 2.8.	
H07	WAKE	WAKE provides an internal pull-up current source to BAT42 and carries battery voltage. In case this signal is routed to any low-voltage hardware, a corresponding voltage divider is recommended.	
H08	RSTN	If an external pull-up resistor is applied at pin RSTN it should be kept in mind that this resistor shouldn’t be too low in order to ensure low reset output voltage at low battery voltages (< 1.5V); see also 2.5.	
H09	INH/LIMP	The INH/LIMP output voltage is related to the BAT42 pin of the SBC and carries battery voltage, if switched on HIGH level. In case this signal is routed to low-voltage hardware, a corresponding voltage divider or clamping device is recommended; see also 2.4.3.	
H10	TEST	Within the final application the TEST input should be connected to GND. Optionally a resistor between TEST and GND is possible, if the TEST pin is used for e.g. ECU programming. Then it is also recommended to connect a capacitor in parallel to the TEST pin; see 2.10.	

No.	Item	Description	Checked
H11	SYSINH	The SYSINH output voltage is related to the BAT42 pin of the SBC and carries battery voltage, if switched on. In case this signal is routed to low-voltage hardware, a corresponding voltage divider or clamping device is recommended.	
H12	Heat sink	The heat sink of the SBC package has to be well contacted with a sufficient cooling area. The heat sink should be connected to GND potential.	
H13	SCS	The SCS pin of the fail-safe SBC should always be on a defined level. Any (unintentional) HIGH-to-LOW transition is interpreted as the start of an SPI communication cycle. Background: When the reset pin is down the pins of many microcontrollers are floating. If the SCS pin is not well defined at this moment a chattering on SCS might occur when the reset pin is released. If necessary the internal pull-up should therefore be strengthened with an external resistor.	
H14	RSTN	The RSTN pin internally has a 3k3 (TYP) pull-up in series with a PMOS transistor. The PMOS transistor is switched off whenever RSTN is pulled LOW by the SBC. If RSTN is pulled LOW externally, the PMOS transistor stays on at first and therefore the pull-up needs to be overcome by the external Reset source.	
H15	TEST	If one of the special modes is entered via pin TEST, the voltage at pin TEST has to be very stable. As soon as it falls outside the specified window, the special mode is left. Therefore it can be necessary to apply an additional Capacitor at pin TEST to eliminate voltage spikes.	

A.2.2 Software design issues

No.	Item	Description	Checked
S01	Power-on	The power-on flag is not cleared before the SBC enters Normal Mode for the first time after power-on. If the first start-up fails the power-on flag is still set even if another reset source than the power-on / external reset is reported by the RSS bits in the Status Register.	
S02	SPI	Take care of maximum SPI speed as described in the according datasheet.	
S03	SPI	An SPI access consists of exactly 16 bits that are transmitted and received. Take care that there are no spikes on the SCS pin during data transfer. An SPI transfer starts with a falling edge on SCS and ends after 16 CLK cycles with the rising edge on SCS. This also has to be guaranteed in case of a microcontroller interrupt during a running SPI transmission; see also 8.2.	
S04	General Purpose Register	Note that the DIC bit in the General Purpose Register 0 cannot be overwritten.	
S05	SBC-ID	Note that once the SBC-ID in the General Purpose Register 0 is overwritten it cannot be retrieved until the next power-on. Only if the DIC bit is cleared the register contains a trustable SBC-ID.	
S06	SBC-ID	It is recommended to compare the SBC-ID data read out of the SBC with	

No.	Item	Description	Checked
		a software expected value in order to check, whether hardware and software match together. In case of deviation, an application specific fallback behavior (e.g. diagnosis entry) is useful.	
S07	Reset source	It is recommended to use the reset source information within the Status Register in order to detect permanently repeating reset events and take application specific measures upon detection of the different resets.	
S08	Wake-up	The following bits in the Status Register are cleared after reading: CAN Wake Status CWS - LIN Wake Status LWS - Edge Wake Status EWS. In order to avoid that information is lost these bits should be evaluated each time the Status Register is read. Note that either the Status or the Diagnosis Register is also read when writing into the Mode Register, e.g. while triggering the watchdog; see also 4.2.	
S09	Local wake	The Wake Level Status bit WLS in the Status Register allows to read back whether the level of the WAKE-Pin is correct before entering Standby or Sleep Mode; see also 4.2.	
S10	Wake-up	Take into account that more than two or all wake-up sources might occur nearly at the same time. If the wake-up causes a reset the RSS bits captures only the first of the wake-up reset events. Details about other wake-up events can be found within CWS, LWS and EWS bits.	
S11	Wake-up	Note that in Normal Mode wake-up interrupts might occur if enabled, but no wake-up resets.	
S12	Wake Port	Note that the WLS bit in the Status Register is working only if the local wake port is enabled by setting the WEN bit in the Configuration Register; see also 4.2.	
S13	Wake-up in cycle mode	In case WAKE is powered out of V3 within cyclic mode, the WSM (Wake Sample Mode) has to be set too in order to prevent permanent unwanted wake-ups by the cyclic V3 supply.	
S14	LIN wake-up	Note that after a wake-up via LIN the LIN wake-up function is blocked until the LWS bit is cleared by reading the Status Register. Take care that every LIN wake-up is noticed and cleared; see 7.3.	
S15	LIN wake-up	Note that the LWEN bit is fixed to ‘1’ in the UJA1069 in order to prevent a "coma-mode" without remote wake-up capability.	
S16	Enter Sleep Mode	Note that entering Sleep Mode is only possible if there is no pending wake-up. That means that the CWS, LWS and EWS bits must be cleared before the mode change is performed; see 5.6.5.	
S17	Enter Flash Mode	Take care that a started sequence for preparing Flash Mode must be completed without interruption. Otherwise the SBC might detect an illegal Mode Register access and performs a reset. All steps of the sequence have to be applied in separate and consecutive watchdog trigger windows; see also 5.7.1.	
S18	Configuration / Physical Layer Register	Note that the Configuration and the Physical Layer Register are not accessible if the SBC is in Flash Mode. Both registers must be configured before entering Flash Mode; see 5.7.1.	

No.	Item	Description	Checked
S19	Self-modifying bits	Note that the following SBC register bits can be self-modified by the SBC in certain situations: Watchdog Timeout Interrupt Enable WTIE -> cleared after a watchdog timeout interrupt event Reset Length Control RLC -> set when entering Restart or Fail-safe Mode INH/LIMP Enable ILEN and INH/LIMP Control ILC -> modified entering Fail-safe state or and at several reset conditions V2 Control V2C -> after detection of a V2 overload or under-voltage CAN Partial Networking CPNC -> cleared after wake-up out of partial networking and when entering Active Mode CAN Transmitter Control CTC -> upon V2 failures V1 Reset Threshold Control V1RTHC -> entering Restart or Fail-safe Mode If necessary these bits have to be reconfigured under software control.	
S20	V1, V2, V3 Diagnosis	Note that the voltage diagnosis bits VxD in the Diagnosis Register are set after reading the Diagnosis Register if the failure is not visible anymore. In order to avoid that information is lost, these bits should be evaluated each time the Diagnosis Register is read. Note that either the Status or the Diagnosis Register is also read when writing into the Mode Register, e.g. while triggering the watchdog.	
S21	V1 under-voltage	A cleared V1D bit in the Diagnosis Register indicates that the V1 voltage was below about 90% of the nominal value. That should be considered if this voltage is used as reference, e.g. for an Analog to Digital Converter.	
S22	V2 failure	Note that if there is a V2 failure while CAN is in Active Mode the CANMD bits in the Diagnosis Register remains on the setting “Active Mode”. The V2 failure is reported by clearing the V2D bit in the same register. Note that the CANMD setting “Active Mode” alone does not say that CAN is ready for transmission. CAN communication is only possible if both conditions are true, the CNAMD bits report “Active Mode” and the V2D bit is set to “1”, see 6.3.2.	
S23	V2 failure	The software should provide a routine that manages V2 failures. Note that the CAN Transmitter is disabled after a V2 failure by automatically setting the CTC bit in the Physical Layer Register. For reactivation of the CAN transmitter the CTC bit must be cleared by software. Avoid overheating the SBC by permanently reactivating V2. Be aware that a V2 failure can be caused either by an overload of V2 or by an under-voltage at BAT14. See also 6.3.2.	
S24	V2 failure in Flash Mode	If the SBC detects a V2 failure while the SBC is in Flash Mode, the failure can only be recovered by leaving Flash Mode, recovering the failure in Normal Mode and entering Flash Mode again. When V2 is off it is not possible to receive data from the bus, see also 5.7.1.	
S25	V3 failure	For reactivating V3 after an overload was detected, the V3C bits in the	

No.	Item	Description	Checked
		Configuration Register must be rewritten. Avoid overheating the SBC by permanently reactivating V3 in case of a hard short; see also 4.1.4.	
S26	Temperature Warning	It is recommended to evaluate the Temperature Warning Status bit in the Status Register and to reduce power consumption.	
S27	CAN / LIN in Active Mode	Note that the CAN and LIN transceivers are only in Active Mode if the dedicated mode control flag is set (CMC or LMC) and if the SBC is in Normal or Flash Mode, see also 6.2.1 and 7.2.1.	
S28	CAN / LIN Transmitter Control	If the CTC or LTC bit is set, the CAN or the LIN transceiver is in listen-only mode. If a CAN controller tries to send data while the transceiver is in listen-only mode the controller might run into bus-off. Note that the CAN transceiver automatically enters listen-only mode in case of a V2 failure, see also 6.2.1.1, 7.2.1.3 and 6.3.2.	
S29	CAN	Note that the CAN transceiver is only ready for communication if the CANMD bits report 'Active Mode' and the V2D bit is '1' (both located in the Diagnosis Register). When activating CAN it might take a few milliseconds until V2 is powered-up and CAN is ready for communication; see also 6.3.	
S30	CAN failure	If one of the CAN failures 'TXDC clamped dominant' or 'RXDC clamped recessive' is reported, the CAN transmitter is automatically deactivated. The CAN transmitter is only reactivated automatically if correct data is received. If the network is sleeping and should be waked, a manual reactivation of the transmitter is necessary using the CTC bit in the Physical Layer Register. It is recommended to provide at least one try to reactivate the transmitter in software. See also 6.5.3.	
S31	LIN failure	If the LIN failures 'TXDL clamped dominant' or 'LIN is shorted to VBAT' are reported the LIN transmitter is automatically deactivated. The LIN transmitter is only reactivated automatically if correct data is received. If the network is sleeping and should be waked-up by the affected node, a manual reactivation of the transmitter is necessary using the LTC bit in the Physical Layer Register. It is recommended to provide at least one try to reactivate the transmitter in software. See also 7.5.	
S32	Disable Watchdog in Standby Mode	Note that disabling the watchdog in Standby Mode is only possible if there is no pending wake-up. That means that the CWS, LWS and EWS bits must be cleared; see 5.5.3.4.	
S33	Disable Watchdog in Standby Mode	Note that disabling the watchdog in Standby Mode requires the current consumption on V1 to be below a certain threshold. The SBC checks the V1 current at the end of the running watchdog period. Take care that the watchdog period is long enough for reducing the V1 current. See also 5.5.3.4.	
S34	Disable Watchdog in Standby Mode	Note that once a stopped watchdog was restarted in Standby Mode it can only be stopped again when the Mode Register is triggered again with the "watchdog off" code and all conditions for disabling the watchdog are fulfilled. See also 5.5.3.4.	
S35	Interrupt	Note that an SBC interrupt requires reading the Interrupt Register within a certain time. Details can be found in 9.	

No.	Item	Description	Checked
S36	Interrupt limitation	Due to the interrupt limitation in Normal Mode (one interrupt per watchdog cycle) for several interrupt sources, it can happen that interrupts are forced with a delay. See also 9.	
S37	Watchdog Timeout Interrupt	Note that the watchdog timeout interrupt enable is cleared automatically after this interrupt was triggered. In case another interrupt is required at the end of the next WD period, WTIE has to be set again under software control; see also 5.5.3.3.	
S38	EN used as error pin	If the EN pin is used in Error Emulation Mode (ERREM == 1) the EN bit in the Mode Register has no function. The ENS bit in the Status Register shows then the status of the EN pin. That means this bit is “0” if there is a CAN failure present. See also 6.5.5.	
S39	Software Development Mode	In a final application the Software Development Mode indicated by the SDMS bit in the Status Register must be cleared, if erroneously enabled. Please make sure that the final code does not make use of any Software Development Mode feature. See also 5.8.1.1.	

A.3 Software Header File (API)

For the UJA106x devices there is a header file available providing some useful declarations for the fail-safe SBC family. This file is attached to this PDF file.

In order to get access to this file, you need to have installed the Adobe™ Reader 6 or above. Opening the files is possible as follows:

Using Adobe™ Reader 6:

Open this PDF File with and select → Document → File Attachment

Following window allows selecting and saving the attached file:

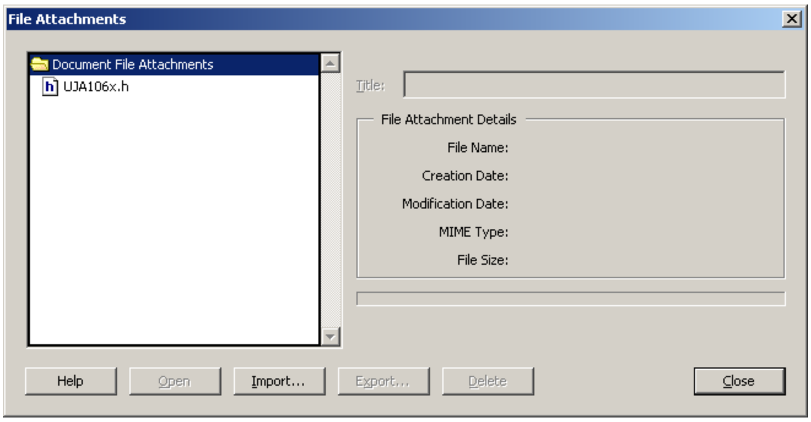
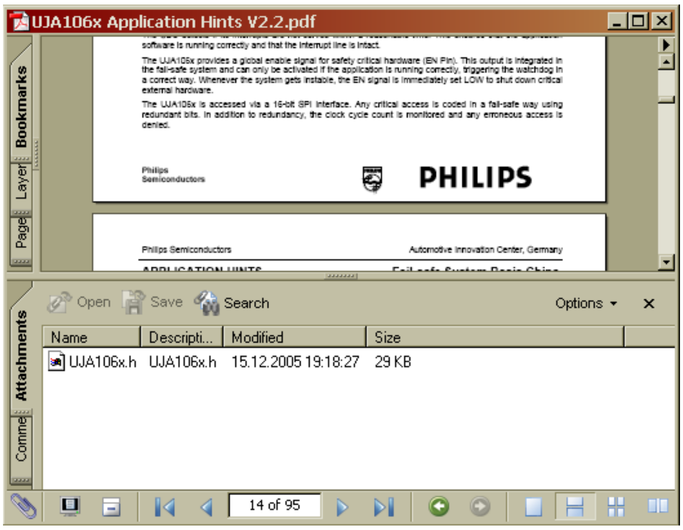


Figure A-1 : Options window of Adobe™ Reader 6.0, Detaching Files

Using Adobe™ Reader 7:

Open this PDF File. Click the “Attachments” tab on the lower left side of the screen.



Name	Descripti...	Modified	Size
UJA106x.h	UJA106x.h	15.12.2005 19:18:27	29 KB

Figure A-2 : Options window of Adobe™ Reader 7.0, Detaching Files

Hint: The Adobe™ Reader can be downloaded at: http://www.adobe.com.

A.4 Thermal issues, package and ECU layout

Since the UJA106x devices include some linear voltage regulators (V1 and V2) and some power consuming transceiver circuits, an ECU based on these fail-safe SBCs has to provide some heat sinking capabilities in order to get rid of the power dissipation. The package of the SBC is designed to have a very low thermal resistance, if the heat sink on the backside of the package is directly soldered onto the printed circuit board.

In order to get a low thermal resistance of the overall ECU it is highly recommended to thermally connect the ECU GND plane to the heat sink of the UJA106x package. In case the GND plane is on the opposite side of the SBC a group of vias can ideally be used to dissipate heat towards the GND plane.

The thermal resistance depends on the size of the GND plane and the layer structure.

In order to estimate the expected power dissipation of the fail-safe SBC in a real system environment the parameters in Table A-2 have to be taken into account. Please note that the values in the table are just examples used for demonstration. The exact numbers can be found in the datasheets of the used fail-safe SBC.

Table A-2: Example data for power consumption calculation

Item	Typical Value	Comment
BAT14 supply voltage	13.5V	Higher supply voltages are typically valid only for relative short time frames and might increase the power dissipation temporarily. Since the SBC will not shut down due to high temperatures and the temperature gradient will be quite slow there are no problems expected.
BAT42 supply voltage	13.5V 42.0V	In a 12V vehicle this input will be connected to the 12V battery. Here about 250uA will flow into the device during Normal Mode. In case of a 42V environment, the 42V have to be assumed. Since here the internal regulator for the LIN recessive voltage needs to be active, the current is about 600uA.
RTL termination	560 Ω externally + 50 Ω internally	This resistor is relevant during all dominant bit times independently from the actively sending node
CANH/L driver load	40mA @ 1,2V 60mA short circuit	This current is relevant during dominant bit times only. The voltage drop of the CAN driver transistors is about 1,2V @ 100 Ω CAN load resistance. In case a CAN wire is shorted to GND, the current increases to 60mA without voltage drop.
CAN transceiver internal bias currents	6mA 18mA	Bias current during recessive bus state, I_{BAT14} (Normal Mode, CAN recessive) Bias current during dominant bus state, I_{BAT14} (Normal Mode, CAN dominant)
LIN load	25mA @ 1,5V	This current is relevant during dominant bit times only It is directly driven out of BAT14. The voltage drop of the LIN driver transistor is about 1,5V @ 500 Ω LIN load resistance.
V1 output voltage	5V / 3,3V	Depending on the derivative the used output voltage is relevant
V2 output voltage	5V	Is always 5V due to CAN requirements
Ambient Temperature	85°C 125°C	Depending on the environmental situation, the ambient temperature of the ECU has to be selected
ICC @ V1	120mA 150mA	Depending on the BAT 14 input voltage, the V1 regulator is able to deliver a corresponding output current.
ICC @ V2	100mA	Due to the CAN transceiver requirements V2 can deliver up to 100mA. This 100mA will never flow permanently because the CAN driver is active only short time in average. Thus, from thermal point of view, this mean current is much lower.

Based on these numbers the power dissipation of the SBC can be calculated.

A.4.1 Example Calculation (UJA1061)

Power consumption related to BAT42 within a 12V system:

$$P_{BAT42} = V_{BAT42} \times I_{BAT42} = 13{,}5V \times 250\mu A = 3{,}4mW$$

Maximum power consumption caused by application microcontroller:

$$\mathrm{P_{V1\_max}} = (\mathrm{V_{BAT14}} - \mathrm{V_1}) \times \mathrm{I_{V1}} = (13{,}5\mathrm{V} - 5\mathrm{V}) \times 120\mathrm{mA} = 1020\mathrm{mW}$$

Power consumption caused by the integrated LIN Transceiver:

$$P_{LIN\_dom} = (V_{BAT14} - V_{LIN\_dom}) \times I_{LIN\_dom} = (13{,}5V - 1{,}5V) \times 25mA = 300mW$$

Since the LIN bus is not permanently dominant and there is not 100% busload on the LIN interface, the average dominant power dissipation is much lower. It could be assumed that a duty cycle of 50% on LIN is a realistic condition. Furthermore the LIN busload will typically be about 50% since there are idle phases between the LIN frames. This leads to following average power dissipation caused by the LIN interface:

$$P_{LIN\_av} = P_{LIN\_dom} \times 0{,}5 \times 0{,}5 = 300mW \times 0{,}25 = 75mW$$

Power consumption caused by the integrated CAN driver (no CAN bus fault):

$$P_{CANH\_dom} = (V_{BAT14} - (V_2 - V_{DropCAH})) \times I_{CAN\_dom} = (13{,}5V - (5V - 1{,}2V)) \times 40mA = 388mW$$
$$P_{CANL\_dom} = V_{Drop\_CANL} \times I_{CAN\_dom} = 1{,}2V \times 40mA = 48mW$$

Since the CAN bus is not permanently dominant and there is not 100% busload on the CAN interface, the average dominant power dissipation is much lower. It could be assumed that a duty cycle of 50% on CAN is a realistic condition. Furthermore the CAN busload will typically be below 50% in a vehicle and this particular node will transmit only a part of the 50% busload, e.g. 10% of it. So, one can assume that the average dominant transmit time is about 0,5 x 0,5 x 0,1. This leads to following average power dissipation caused by the CANH driver:

$$P_{CANH\_av} = P_{CANH\_dom} \times 0{,}5 \times 0{,}5 \times 0{,}1 = 388mW \times 0{,}025 = 9{,}7mW$$
$$P_{CANL\_av} = P_{CANL\_dom} \times 0{,}5 \times 0{,}5 \times 0{,}1 = 48mW \times 0{,}025 = 1{,}2mW$$

Besides the CANH driver there is some current flowing within the CANL termination path, which is also connected to V2. For CANL the same average current assumptions are true like for CANH with the exception that any dominant bit causes this current. The maximum current in the CANL termination calculates as follows:

$$I_{RTL\_dom} = \frac{V_2 - V_{DropCANL}}{R_{RTL\_ext} + R_{RTL\_int}} = \frac{5V - 1{,}2V}{560\Omega + 50\Omega} = 6{,}2mA$$

The power dissipation during dominant bit times calculates to:

$$P_{RTL\_dom} = (V_{BAT14} - V_{DropCANL}) \times I_{RTL\_dom} = (13{,}5V - 1{,}2V) \times 6{,}2mA = 76{,}6mW$$

According to the busload assumptions about dominant bus times, the average power dissipation is:

$$P_{RTL\_av} = P_{RTLH\_dom} \times 0{,}5 \times 0{,}5 = 76{,}6mW \times 0{,}25 = 19{,}15mW$$

The CAN transceiver itself is also consuming some bias current, which is loading the V2 regulator and thus, causes some power dissipation:

$$P_{Trans\_rec} = V_{BAT14} \times I_{Trans\_rec} = 13{,}5V \times 6mA = 81mW$$

$$P_{Trans\_dom} = V_{BAT14} \times I_{Trans\_dom} = 13{,}5V \times 18mA = 243mW$$

Due to the bus loading assumptions, following average power dissipation is caused by the CAN transceiver:

$$P_{Trans\_rec\_av} = P_{Trans\_rec} \times (1 - 0{,}5 \times 0{,}5 \times 0{,}1) = 81mW \times 0{,}975 = 79mW$$

$$P_{Trans\_dom\_av} = P_{Trans\_dom} \times 0{,}5 \times 0{,}5 \times 0{,}1 = 243mW \times 0{,}025 = 6{,}1mW$$

$$P_{Trans\_av} = P_{Trans\_dom\_av} + P_{Trans\_rec\_av} = 79mW + 6{,}1mW = 85{,}1mW$$

Collecting all power dissipation portions, the overall power dissipation calculates to:

$$P_{SBC\_\max V1load} = P_{BAT42} + P_{V1\_\max} + P_{LIN\_av} + P_{CANH\_av} + P_{CANL\_av} + P_{RTL\_av} + P_{Trans\_av}$$

$$\mathrm{P_{SBC\_max\,V1load}} = 3{,}4\mathrm{mW} + 1020\mathrm{mW} + 75\mathrm{mW} + 9{,}7\mathrm{mW} + 1{,}2\mathrm{mW} + 19{,}2\mathrm{mW} + 85{,}1\mathrm{mW} = 1{,}2\,\mathrm{W}$$

Assuming a typical ECU layout making use of a GND plane connected to the heat sink of the SBC the thermal resistance will be about 40K/W. Thus, the temperature increase with respect to the environmental temperature will calculate as follows:

$$\Delta\vartheta = \mathrm{P_{SBC\_max\,V1load}} \times \mathrm{TK_{SBC\_to\_ambient}} = 1{,}2\mathrm{W} \times 40\frac{\mathrm{K}}{\mathrm{W}} = 48\mathrm{K}$$

Since the UJA1061 is specified for virtual junction temperatures up to 150° C, a typical design would allow to make use of the maximum V1 current capabilities of the UJA1061 up to 120mA in case the environmental temperature stays below 100°.C.

It should be mentioned, that the power dissipation will increase slightly in case there is a bus short circuit on CANH towards GND. In this case the average CANH power dissipation increases as follows:

$$P_{CANH\_dom\_fail} = V_{BAT14} \times I_{CANH\_dom\_fail} = 13{,}5V \times 60mA = 810mW$$

$$P_{CANH\_av} = P_{CANH\_dom\_fail} \times 0{,}5 \times 0{,}5 \times 0{,}1 = 810mW \times 0{,}025 = 20{,}3mW$$

In case of this CANH short, the power dissipation will increase by about 10.6mW and thus, cause about 0.5° C extra temperature. Compared to the other factors, this extra dissipation can be neglected.

A.5 Power saving options with the fail-safe SBC

The fail-safe SBC offers several possibilities for saving power in a system. From ECU's designed with stand-alone transceivers and voltage regulators, two main low-power modes are already in use:

- Standby Mode Microcontroller stays supplied, all peripherals are in low-power condition
- Sleep Mode Microcontroller completely un-powered, transceiver monitors system with respect to wake-up conditions

These well-known functions are of course also available in the fail-safe SBC. Besides this the fail-safe SBC offers further possibilities to significantly decrease the system current. Since the UJA106x family integrates a time base for the watchdog block, there is the possibility to make use of this time base for a cyclic wake-up of the ECU with programmable periods.

This cyclic wake-up function can be used very effectively in applications, which need to check e.g. a sensor value from time to time without the need of keeping the microcontroller alive all the time. A cyclic wake-up of the SBC can be programmed in a wide timing range up to 28 seconds. Depending on the ECU requirements such a cyclic wake-up can force a system reset or an interrupt to the microcontroller.

A.5.1 Cyclic Wake-up in Standby Mode

During Stand-by Mode the application microcontroller is typically still supplied. Depending on the type of microcontroller a so-called "Stop mode", "Idle mode" or "Power-down mode" can be entered in order to save current consumption in the system.

In conventional ECUs with discrete components a system start out of this Standby condition is performed with a hardware reset and thus, requires a relatively long time. The software has to be re-initialized and a lot of time is wasted until the short application task can be executed. For a simple sensor application most time would be needed to start the system. Furthermore there is the need to keep some timer alive, which starts the ECU from time to time. Thus, there is no big amount of current, which could be saved using discrete ECU's.

Using the fail-safe SBC the advantage is that starting out of Standby can be performed with an interrupt from the SBC without causing a hardware reset. Furthermore the time base starting the system is already implemented in the SBC. So, there is no extra timer hardware required. Starting on interrupt base allows continuing executing the application code directly without the need for re-initialization of all the software. So the relatively long boot time of the application can be saved.

A.5.2 Cyclic Wake-up in Sleep Mode

Since the lowest system current consumption is achieved in Sleep Mode, this mode could achieve lowest system currents combined with cyclic wake-ups. Here the microcontroller is entirely un-powered and thus, only the SBC sleep current is flowing.

For the average current consumption it has to be taken into account that an ECU waking up from sleep always has to perform the system initialization routine starting up the application. Thus, there is a significant extra time with a running microcontroller, similar to the cyclic wake-up out of Standby with Reset behavior.

As a consequence a cyclic wake-up out of Sleep becomes more interesting for very long off-times of the application or for systems with very short start-up times. This is, why the SBC offers longer cyclic period times during Sleep Mode than during Standby Mode.

Furthermore the start-up time of an application can become quite short, since the SBC offers very detailed system information at start-up combined with a short reset period.

From system point of view it has to be checked, which of the options allows the lowest average current consumption. Depending on the system requirements the Standby Mode combined with an interrupt based wake-up will have an advantage for short cycle times while the Sleep Mode will have advantages at long cycle times.

A.5.3 Power saving examples

An application with 40mA current consumption is assumed. The start-up time after Reset is about 30ms, the real application just requires 5ms. The application should check a sensor value from time to time. The SBC requires about 250uA during Standby and 50uA during Sleep. Four different possibilities are compared:

- Standby Mode with cyclic Reset (4seconds period)
- Standby Mode with cyclic Interrupt (4 seconds period)
- Sleep Mode with cyclic power-on (4 seconds period)
- Sleep Mode with cyclic power-on (8 seconds period)

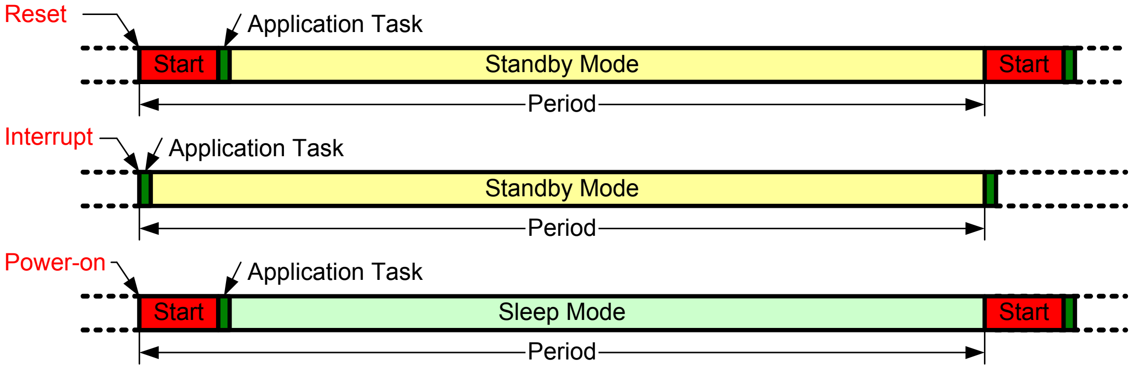


Figure A-3 : Comparison of cyclic wake-up scenarios

With the above assumptions, following average current consumptions would be achievable using the SBC:

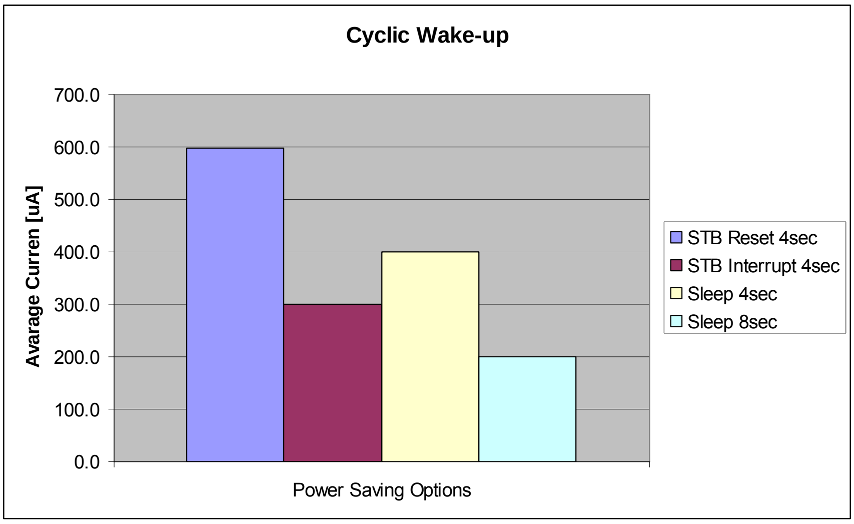


Figure A-4 : Avarage current consumption of example

For this particular example, the Standby Mode with cyclic interrupt every 4 seconds is more attractive than the cyclic Sleep Mode with 4 seconds period time due to the required software start-up time with high microcontroller current consumption. The power-saving options of the Sleep Mode are getting more attractive with longer sleep period times (see 8-sec. example).

A.6 FAQ, Frequently Asked Questions

Within this chapter some frequently asked questions are collected. Furthermore some “Tips & Tricks” can be found here.

1. What has to be done to perform a system reset initiated by software?

When a system reset is performed it is essential that this event is seen simultaneously by the microcontroller and the fail-safe SBC. In principle there are two ways to initiate this reset. The first one is directly performed by the microcontroller. The second one is done by the fail-safe SBC after provocation by software.

Many microcontrollers support the possibility of initiating resets by software. Here it must be attended that the reset must also be visible at the reset pin so that also the SBC is reset. Hence the reset pin of the microcontroller must be bidirectional. After such a reset the fail-safe SBC reports an external reset as reset source in the Status Register.

If the reset pin of the microcontroller is not bidirectional or if it does not support software resets, it is possible to cause a system reset of the fail-safe SBC. This can be done by writing any illegal command into the Mode Register, e.g. 0x0000. If this is done the SBC immediately performs a reset and reports an “illegal SPI access” as reset source.

2. What must be considered for entering Sleep Mode?

It is possible that the fail-safe SBC does not enter Sleep Mode after sending the dedicated command. Instead a system reset is performed. There can be several reasons for this behavior:

1. The Sleep Mode command used an illegal watchdog code
2. The Sleep Mode command was not sent in an open watchdog window (only for Normal Mode)
3. There was a pending wake-up

In case 1 the fail-safe SBC has received an illegal Mode Register command. After the reset the Status Register returns the reset source “illegal SPI access”.

In case 2 the command was sent before half of the watchdog period has elapsed. Here the Status Register returns the reset source “watchdog triggered too early” after the reset.

In case 3 the fail-safe SBC has previously received a wake-up either via CAN, LIN or the local WAKE pin. After the reset the Status Register shows the dedicated wake-up as reset source. In order to prevent that any wake-up is lost the fail-safe SBC requires that every wake-up is acknowledged by software. Details can be found in 5.6.5.

3. How to ensure that no wake-up is lost when entering Sleep Mode?

After receiving a global “go to sleep” command from the bus it may take some time before this command can be executed because changing in to Sleep Mode is only allowed when the watchdog window is open. In order to prevent that until then a wake-up is lost it is recommended to leave immediately Active Mode of the CAN and the LIN transceiver by clearing the CMC and / or the LMC respectively in the Physical Layer Register. If this is done any CAN or LIN wake-up message is seen and the dedicated wake status flag in the Status Register is set (CWS, LWS or EWS in case of an edge at the local WAKE pin). In order to prevent that the received wake-up is lost the fail-safe SBC does not execute the Sleep Mode command if one of the mentioned bits is set. Instead a system reset is performed and the dedicated wake-up event is reported as reset source in the Status Register.

4. How to detect a valid SBC identification code?

After power-on the General Purpose Register 0 is loaded with the identification code of the SBC type. Since it is thinkable that the General Purpose Register can be erroneously overwritten the software must take care that the read SBC-ID is correct in a fail-safe application. This can be done by evaluating the Device

Identification Code bit (DIC, bit 11 in the General Purpose Register 0). This bit is set to “0” at power-on. As soon as the General Purpose 0 Register is overwritten this bit is automatically set to “1”. The bit itself cannot be overwritten. As long as this bit is “0” the software can be sure that rest of the General Purpose Register shows the true SBC identification code.

5. How to check if the heat sink is properly connected?

The heat sink of the SBC package should be soldered to the Ground plane of the PCB to ensure optimal heat transfer from the device to the environment. For checking if the heat sink is properly connected a simple method can be applied.

Between the pin BAT14 and the pin V1 there is a reverse diode; see Figure A-5. If the pin BAT14 is shorted to GND and a current is fed into V1 the device heats up. The chip temperature depends on the power dissipation and the thermal resistance to ambient air. The power dissipation can be simply measured. The chip temperature can be estimated by determining the I-to-V characteristic of the diode. Both measurements can be used for calculating the thermal resistance and this value indicates whether the heat sink is properly soldered or not. Some 100mA can be used to heat-up the SBC with the reverse current for limited time.

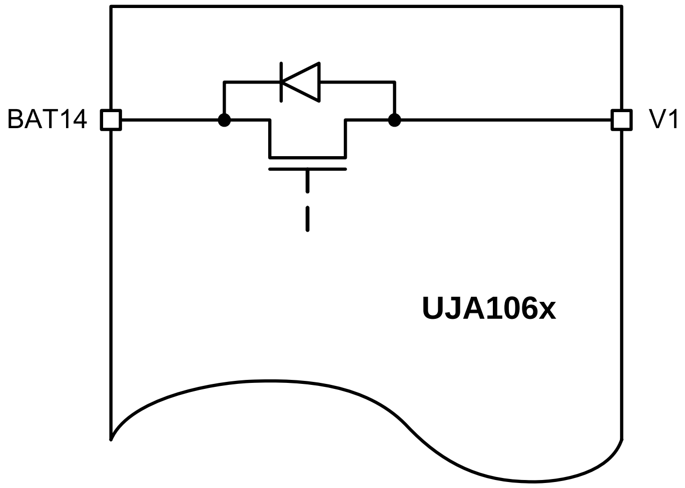


Figure A-5: Using V1 regulator diode for measuring the thermal resistance

6. How to measure the battery voltage with the AD-converter of the microcontroller?

Automotive control units often permanently observe the level of the battery voltage. This is usually done by measuring the voltage with an AD-converter in the microcontroller. Because of this there is a connection via a voltage divider between the battery supply and the microcontroller. In case the fail-safe SBC is in Sleep or Fail-safe Mode the microcontroller is normally unsupplied. To avoid that there is a current flowing from battery to the un-powered microcontroller it is necessary to insert a switch between the battery pin and the voltage divider. The switch must be open when the UJA106x is in Sleep or Fail-safe Mode.

For such applications the INH/LIMP pin of the fail-safe SBC can ideally be used as shown in Figure A-6. The INH/LIMP pin can be set on battery level by setting the ILEN and the ILC bit to “1” in the Configuration Register. As soon as the fail-safe SBC enters Sleep or Fail-safe Mode the INH/LIMP pin is automatically set to GND level or the pin is floating.

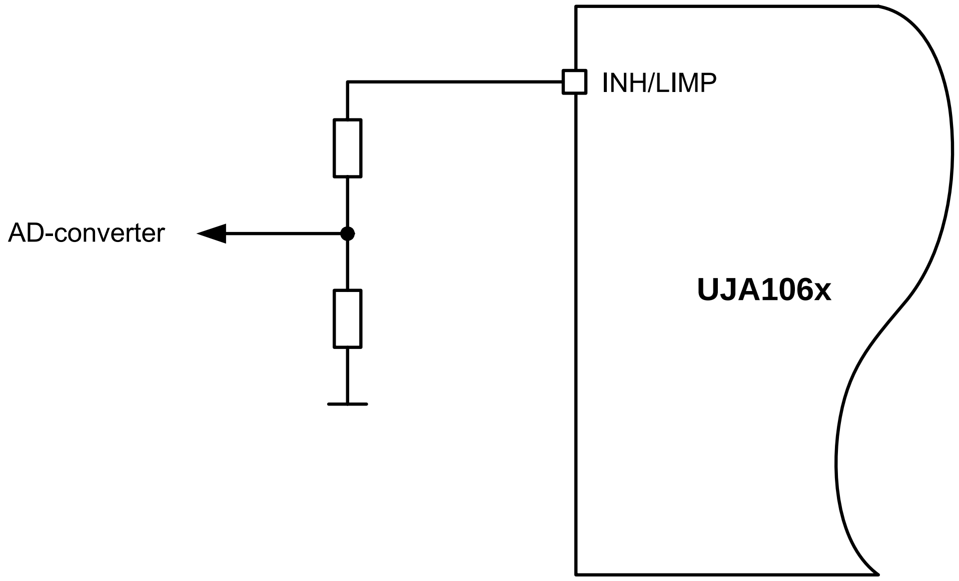


Figure A-6: Measuring battery voltage with an AD-converter via INH/LIMP pin

7. *How can it be distinguished in which mode the SBC currently is?*

In Forced Normal Mode the SPI communication is not possible. Pin SDO is in high-resistive state.

In Flash Mode: GPR Registers can be written.

8. *Is it possible to flash microcontrollers via a BDM module with an external Voltage Source?*

The idea is to supply the microcontroller with an external Voltage Source during Flash programming. The problem is that the SBC pulls down the RESET pin as soon as it is supplied backwards via the V1 output. Hence using the BDM becomes impossible. For initial Flash programming it is recommended to use the Forced Normal Mode (see 2.10). It is explicitly not recommended to use a series resistor in the reset line!

9. *Why are there spikes on SDO?*

The reason for the occurrence of these spikes is that the first three bits of the SPI message determine which register is read, and therefore it is not clear which data has to be pushed out before the third bit has been received. The spike results from switching the internal multiplexers. However these spikes only occur at the rising edge of SCK whereas data is sampled at the falling edge. At the falling edge the signal is always stable. Therefore no error can occur.

10. *Why does the SBC not leave Fail-Safe mode after a hard shortcut of V1 to GND?*

If V1 is shorted to GND with a very low resistance (<< 1Ω) while there is communication on CAN or LIN, it is possible that the SBC suppresses any further wake-up signals on the affected channel (CAN or LIN). After leaving Fail-Safe mode suppression of wake-ups on the affected channel discontinues.

A.7 Device Identification Codes

During power-up, the bits of the General Purpose Register 0 (GP0) will be loaded with a ‘Device Identification Code’ (DIC), consisting of the SBC type and SBC version. Bit 11 of GP0 reflects a logical ‘0’ indicating the content of this register to be the ‘Device Identification Code’ after Power-on. With writing to this register, bit 11 will be set to a permanent logic ‘1’ indicating that the device code has been overwritten with application-specific information. Table A-3 shows the different ‘Device Identification Codes’ for the SBC family engineering samples listed above.

Table A-3: Device Identification Codes

		GP0.11	GP0.10	GP0.9	GP0.8	GP0.7	GP0.6	GP0.5	GP0.4	GP0.3	GP0.2	GP0.1	GP0.0	
Device		**DIC**	**Silicon status**				**SBC type**							**Hex.**
UJA1061	**N1C**	0	0	0	1	1	0	0	0	0	0	0	1	0x181
	N1D	0	0	1	0	0	0	0	0	0	0	0	1	0x201
	N1E	0	0	1	0	1	0	0	0	0	0	0	1	0x281
UJA1065	**N1A**	0	0	0	0	1	0	0	0	0	1	0	1	0x085
	N1B	0	0	0	1	0	0	0	0	0	1	0	1	0x105
	N1C	0	0	0	1	1	0	0	0	0	1	0	1	0x185
	N1D	0	0	1	0	0	0	0	0	0	1	0	1	0x205
UJA1066	**N1A**	0	0	0	0	1	0	0	0	0	1	1	0	0x086
	N1B	0	0	0	1	0	0	0	0	0	1	1	0	0x106
	N1C	0	0	1	0	0	0	0	0	0	1	1	0	0x206
UJA1069	**N1A**	0	0	0	0	1	0	0	0	1	0	0	1	0x089
	N1B	0	0	0	1	0	0	0	0	1	0	0	1	0x109

B References

[1] UJA1065 High-speed CAN/LIN fail-safe system basis chip Rev. 07 - 25.02.2010 Product data sheet

[2] UJA1061 Fault-tolerant CAN/LIN fail-safe system basis chip Rev. 06 - 09.03.2010 Product data sheet

[3] UJA1069 LIN fail-safe system basis chip Rev. 04 – 28.10.2009 Product data sheet

[4] UJA1066 High-speed CAN fail-safe system basis chip Rev. 03 – 17.0.2010 Product data sheet

C Legal information

C.1 Disclaimers

General — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of a NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is for the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

C.2 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.