UM11380

TJA1145A evaluation board (compatible with Arduino UNO)

Rev. 2 — 14 October 2021 User guide

Document information

Information	Content
Keywords	TJA1145A, high-speed CAN transceiver, partial networking, FlexGUI software interface, Arduino UNO
Abstract	This document describes the operation of the TJA1145A-EVB evaluation board and accompanying FlexGUI software package.

Revision history

Rev	Date	Description
v.2	20211014	text of Important Notice changed (this page); Section 6 added
v.1	20201104	Initial version

IMPORTANT NOTICE

For engineering development or evaluation purposes only

NXP provides the product under the following conditions:

This evaluation kit is for use of ENGINEERING DEVELOPMENT OR EVALUATION PURPOSES ONLY. It is provided as a sample IC pre-soldered to a printed-circuit board to make it easier to access inputs, outputs and supply terminals. This evaluation board may be used with any development system or other source of I/O signals by connecting it to the host MCU computer board via off-the-shelf cables. This evaluation board is not a Reference Design and is not intended to represent a final design recommendation for any particular application. Final device in an application heavily depends on proper printed-circuit board layout and heat sinking design as well as attention to supply filtering, transient suppression, and I/O signal quality.

The product provided may not be complete in terms of required design, marketing, and or manufacturing related protective considerations, including product safety measures typically found in the end device incorporating the product. Due to the open construction of the product, it is the responsibility of the user to take all appropriate precautions for electric discharge. In order to minimize risks associated with the customers' applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards. For any safety concerns, contact NXP sales and technical support services.

1 Introduction

This document is the user guide for the TJA1145A-EVB evaluation board. It is intended for engineers involved in the evaluation, design, implementation, and validation of the TJA1145A high-speed CAN transceiver for partial networking. This guide discusses power supply requirements and the MCU and CAN bus interfaces, and describes how to connect the board into an ECU/CAN network. It also describes how to install and use the accompanying GlexGUI software.

The TJA1145A-EVB evaluation board is designed to facilitate the testing and evaluation of TJA1145A product features in a variety of microcontroller IO interface environments. All MCU interface signals can be accessed in two ways: they are available at a header row on the top side and also at header rows on the bottom side that can be plugged directly into many NXP MCU evaluation boards. The TJA1145A-EVB board is designed to be compatible with the S32K1xx evaluation board series from NXP and to support the use of standard software development tools and drivers.

2 Board overview

Top and bottom views of the TJA1145A-EVB evaluation board are illustrated in Figure 1.

Board dimensions are 45.1 mm × 58.4 mm. Only components needed to support basic TJA1145A functionality are included. The board contains CAN communication, power supply and wake-up circuitry and LEDs indicating when pins BAT and VCC are supplied. It provides several header rows (2.56 mm pitch) for connecting MCU interface and application signals.

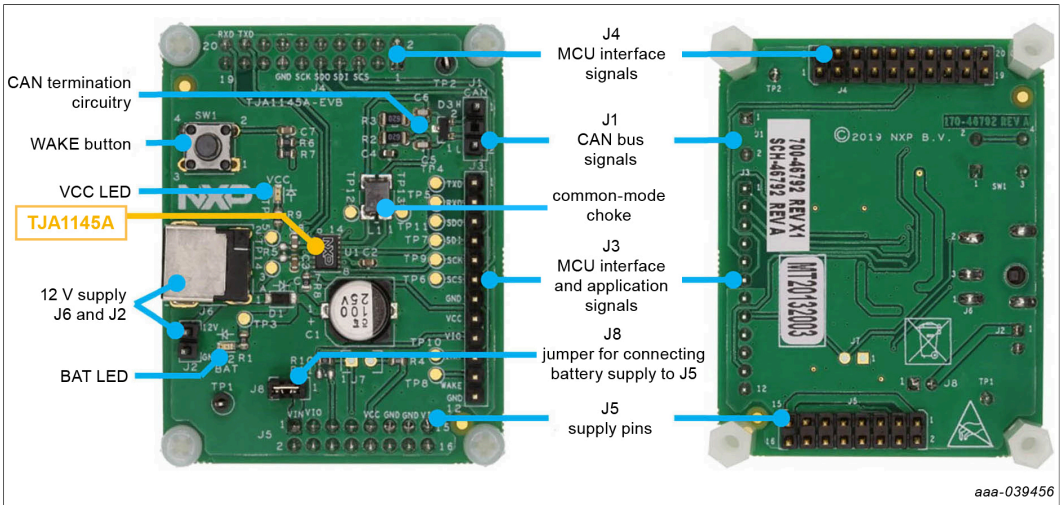


Figure 1. TJA1145A-EVB top (left) and bottom (right) views.

2.1 Ground connections

All ground pins are connected to the ground plane.

Table 1. Ground connections

Ground connections
J2-02
J3-07/12
J4-13
J5-11/13
J6-02/03

2.2 Power supply connections

2.2.1 Battery connections

An external power supply must be connected to either power jack J6 or 2-pin connector J2, as illustrated in Figure 2.

Table 2. TJA1145A/TJA1145A-EVB pins

TJA1145A	TJA1145A-EVB
BAT (pin 10)	J2-01 or J6-01: connect to battery supply

Both supply circuits are routed via polarity protection Schottky diode D1 in order to block reverse currents. Decoupling capacitors C1 and C2 are provided to stabilize the input voltage and remove noise on the battery connection.

Green LED D2 lights up once the 12 V power supply has been connected.

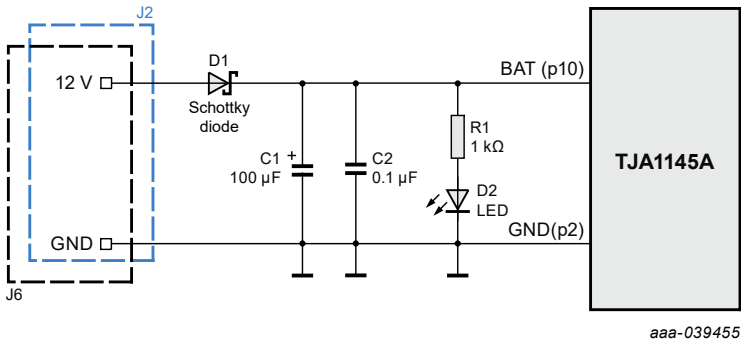


Figure 2. 12 V power supply connection options

2.2.2 VCC/VIO connections

A 5 V VCC supply is needed to operate the CAN transmitter and receiver in Normal and Standby modes. A VIO supply is needed to supply the digital IOs and MCU interface (e.g. SPI pins). The VIO voltage must be aligned with the MCU interface supply voltage. The VCC and VIO voltages are not needed for Sleep mode. Detailed information on the functionality and operation of the TJA1145A can be found in the data sheet and application hints (see Section 7).

Table 3. TJA1145A/TJA1145A-EVB pins

TJA1145A	TJA1145A-EVB
VCC (pin 3)	J3-08 or J5-09: connect 5 V supply voltage
VIO (pin 5)	J3-09 or J5-03: connect MCU-compatible supply voltage
-	J5-01 or J5-15: pin VIN on TJA1145A-EVB board; connected to battery supply by default via jumper J8; remove jumper J8 to disconnect VIN on J5 from the battery supply

The VCC and VIO supplies can be connected to either J3 or J5. J3 is located on the top of the TJA1145A-EVB evaluation board and J5 is mounted on the bottom of the board. The J5 connector pin arrangement follows the Arduino Uno pinout order, allowing the TJA1145A-EVB to be connected directly to a variety of NXP MCU evaluation boards. Decoupling capacitors C3 and C8 are provided to stabilize the input voltages and remove noise on the VCC and VIO inputs. Red LED D4 lights up when the VCC voltage is present.

By default, the TJA1145A-EVB evaluation board battery supply is routed to the MCU board via pin VIN on the Arduino connector, allowing the supply to the entire module to be managed via the TJA1145A-EVB board. This feature can be disabled by removing jumper J8, disconnecting the battery supply from pin VIN.

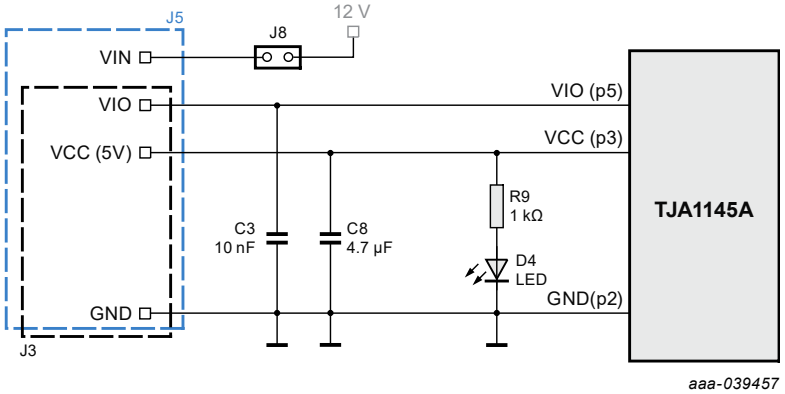


Figure 3. VCC/VIO supply connection options

2.3 CAN communication circuitry

The TJA1145A-EVB evaluation board contains typical CAN communication circuitry. The CANH and CANL bus signals are output on connector J1.

Table 4. TJA1145A/TJA1145A-EVB pins

TJA1145A	TJA1145A-EVB
CANH (pin 13)	J1-01: connect to HIGH-level CAN bus line
CANL (pin 12)	J1-02: connect to LOW-level CAN bus line

Equipped with termination resistors R2 and R3, the TJA1145A-EVB evaluation board is pre-prepared to be used as a termination node in a CAN network. If the CAN network is already terminated at both ends, it is recommended to remove R2 and R3 or replace them with higher value resistors to ensure that the impedance on the bus meets the CAN bus load specification, typically 60 Ω.

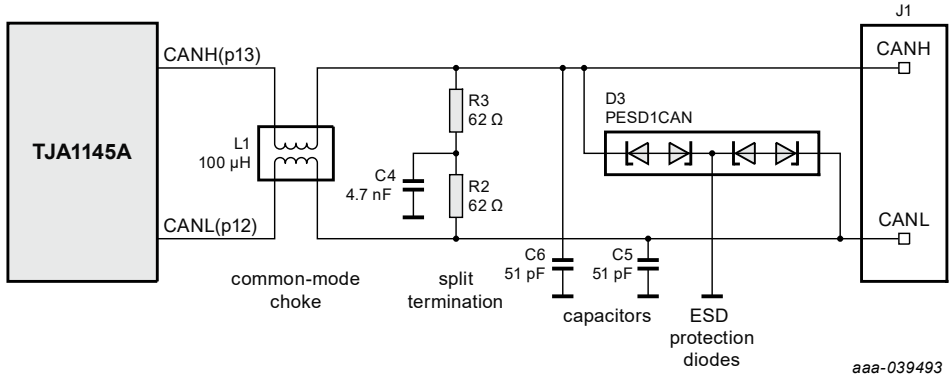


Figure 4. CAN bus interface circuitry

2.4 Wake-up and INH functionality

The TJA1145A supports a Sleep mode for use in energy-sensitive applications. Once in Sleep mode, the device will remain in this low-power mode until a wake-up request is received. A wake-up event can be triggered remotely via a standard pattern or dedicated wake-up frame on the CAN bus, or locally via the WAKE pin (details of wake-up functionality can be found in the TJA1145A data sheet and application hints; see Section 7).

Table 5. TJA1145A/TJA1145A-EVB pins

TJA1145A	TJA1145A-EVB
WAKE (pin 9)	J3-11: connect to wake-up signal
INH (pin 7)	J3-10: connect to control input signal from external regulator(s)

The TJA1145A-EVB evaluation board features local wake-up test circuitry. The WAKE pin is pulled HIGH by default via 10 kΩ resistors R6 and R7. When switch SW1 is pressed, the WAKE pin is pulled LOW. To make use of this feature, falling-edge detection on the WAKE pin must be enabled in the TJA1145A register map (as described in the TJA1145A data sheet).

Pin INH is typically used to control the supply to the MCU and peripherals. In Normal and Standby modes, the level on this pin is equivalent to the voltage on pin BAT. Pin INH is pulled LOW via resistor R8 when the TJA1145A switches to Sleep mode. The MCU can detect when the device is in Sleep mode by monitoring the voltage on this pin.

The WAKE and INH pins are accessible via connector J3 on the top of the board.

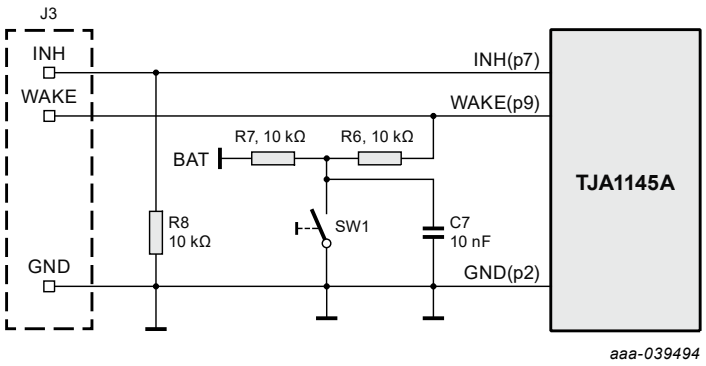


Figure 5. Local wake and INH circuitry

2.5 MCU interface

The digital interface pins are located on the top side connector J3 (J3-01 to J3-06), as well as on the bottom side connector J4. Two of these pins, TXD and RXD, are used for CAN data communication with the MCU. The remaining four pins are used for SPI communication with the MCU.

Table 6. TJA1145A/TJA1145A-EVB pins

TJA1145A	TJA1145A-EVB
TXD (pin 1)	J3-01 or J4-18
RXD (pin 4)	J3-02 or J4-20
SDO (pin 6)	J3-03 or J4-09
SDI (pin 11)	J3-04 or J4-07
SCK (pin 8)	J3-05 or J4-11
SCSN (pin 14)	J3-06 or J4-05

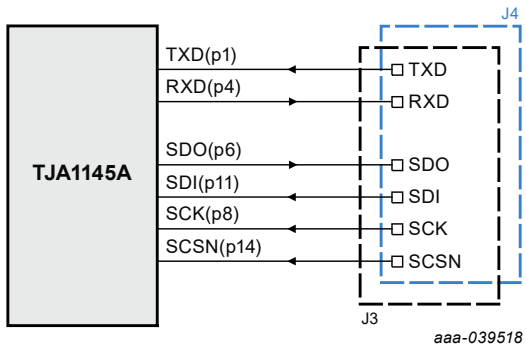


Figure 6. MCU interface

2.6 TJA1145A-EVB to TJA1145A pin summary

Table 7. Overview of TJA1145A-EVB to TJA1145A cross-link

TJA1145A-EVB evaluation board			TJA1145A		
Header name	**EVB pin #: top**	**EVB pin #: bottom**	**Test pad #:**	**Pin #**	**Pin name**
CAN H[1]	J1-01	-	TP12	13	CANH
CAN L[1]	J1-02	-	TP13	12	CANL
TXD	J3-01	J4-18	TP4	1	TXD
RXD	J3-02	J4-20	TP5	4	RXD
SDO	J3-03	J4-09	TP11	6	SDO
SDI	J3-04	J4-07	TP7	11	SDI
SCK	J3-05	J4-11	TP9	8	SCK
SCS	J3-06	J4-05	TP6	14	SCSN
GND	J3-07/12, J2-02, J6-02/03	J4-13, J5-11/13	TP1, TP2	2	GND
VCC	J3-08	J5-09	TP15	3	VCC
VIO	J3-09	J5-03	TP14	5	VIO
INH	J3-10	-	TP10	7	INH
WAKE	J3-11	-	TP8	9	WAKE
-	-	-	TP3	10	BAT
12 V	J2-01, J6-01, J8-02	-	-	-	-
VIN	J8-01	J5-01/15	-	-	-

[1] Common mode choke L1 connects TJA1145A pins CANH/CANL to header pins CAN H/CAN L.

3 Connecting the TJA1145A-EVB to a CAN network

An example of how to connect the TJA1145A-EVB between an MCU and the CAN bus is shown in Figure 7.

The following conditions must be met before powering up the system with a 12 V supply:

- Connect all boards in the ECU to a common GND
- Connect SPI pins to the MCU SPI master:
  - SDI → MOSI
  - SDO → MISO
  - SCK → SCK
  - SCSN → CS
- Connect TXD/RXD pins to the MCU CAN controller TXD/RXD pins
- Connect CANH and CANL to the CAN bus twisted-pair cables
- Connect VCC and VIO to the MCU supply unit; VIO shares the MCU IO supply
- Connect pin INH to the control/enable pin on the ECU supply unit (optional)

Once the above steps have been completed, the ECU/EVB can be powered up using an external battery supply. The TJA1145A starts up in Standby mode, awaiting commands from the MCU via the SPI interface.

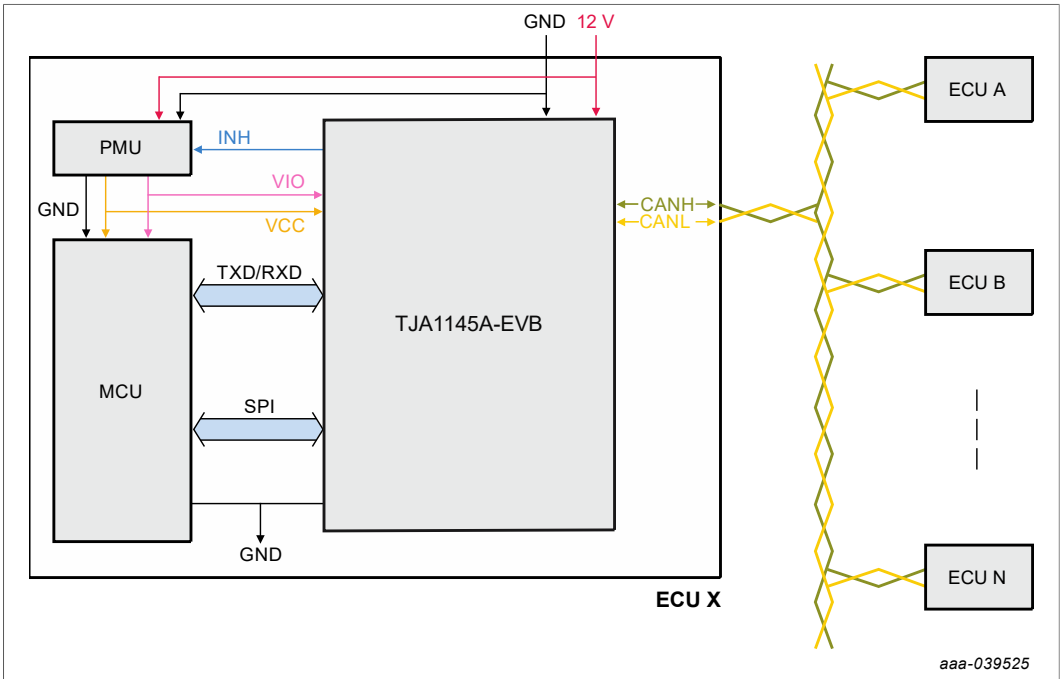
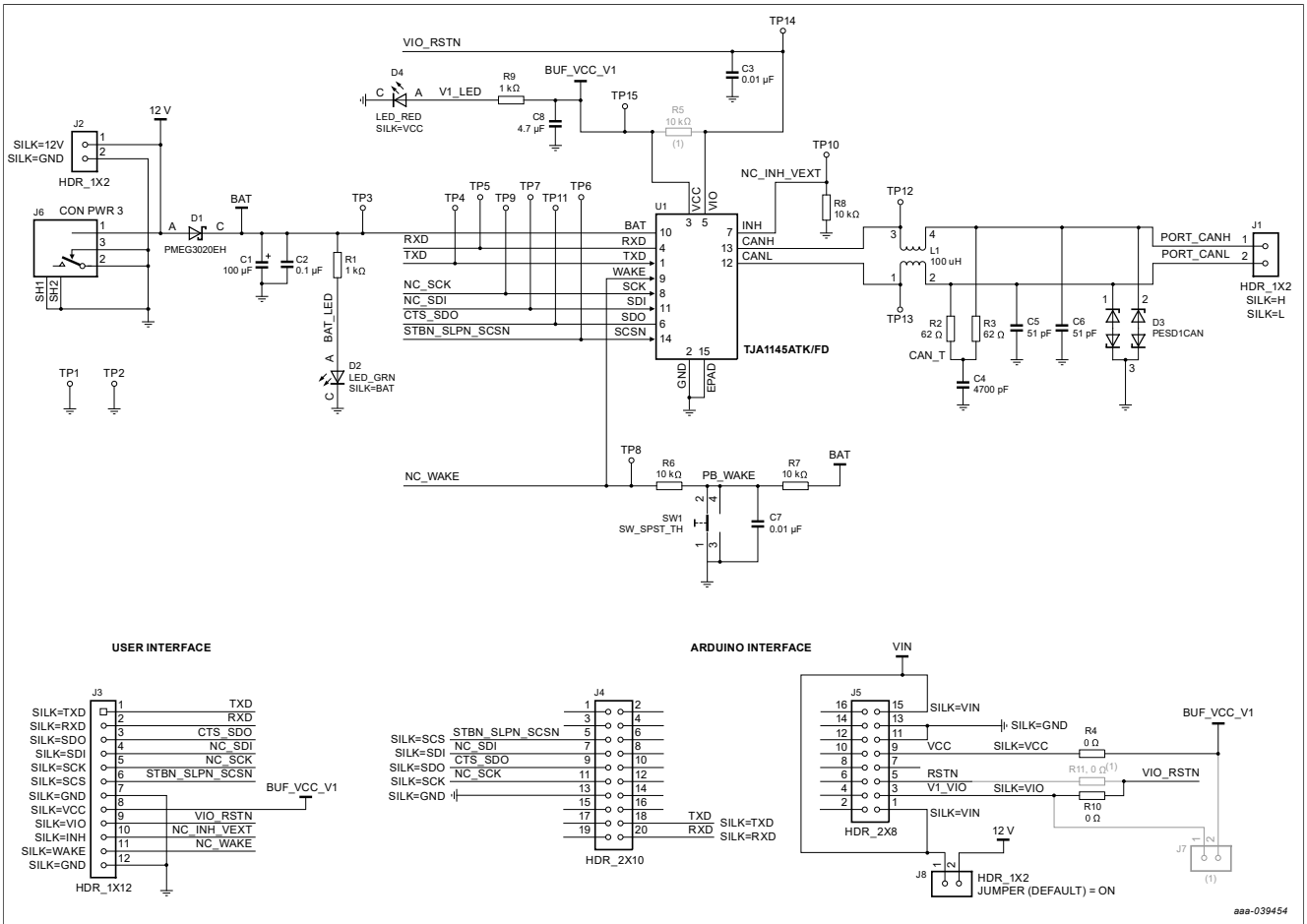


Figure 7. Connecting the TJA1145A-EVB in an ECU/CAN bus network

4 Schematic diagram



(1) Component not populated.

Figure 8. TJA1145A-EVB schematic diagram

5 Bill of Materials

Table 8. Bill of Materials

NXP does not assume liability, endorse, or warrant components from external manufacturers referenced in circuit drawings or tables. While NXP offers component recommendations in this configuration, it is the responsibility of the customer to validate their application.

For critical components, it is vital to use the manufacturer listed.

Item number	Quantity	Schematic label	Value	Description	Part number	Manufacturer name
Active components						
1	1	U1	TJA1145ATK/FD	IC XCVR CAN FD 4.5-5.5 V 5 Mbit/s AEC-Q100 HVSON14	TJA1145ATK/FD	NXP SEMICONDUCTORS
Capacitors						
3	1	C1	100 µF	CAP ALEL 100 µF 25 V 20 % -- SMT	UWX1E101MCL1GB	NICHICON
5	1	C2	0.1 µF	CAP CER 0.1 µF 25 V 10 % X7R AEC-Q200 0603	CGA3E2X7R1E104K080AA	TDK
4	2	C3, C7	0.1 µF	CAP CER 0.01 µF 50 V 10 % X7R AEC-Q200 0603	CGA3E2X7R1H103K080AA	TDK
7	1	C4	4700 pF	CAP CER 4700 pF 50 V 5 % C0G AEC-Q200 0603	CGA3E2C0G1H472J080AA	TDK
2	2	C5, C6	51 pF	CAP CER 51 pF 50 V 5 % C0G 0603	CC0603JRNPO9BN510	YAGEO
6	1	C8	4.7 µF	CAP CER 4.7 µF 16 V 10 % X5R 0603	GRM188R61C475KE11D	MURATA
Diodes						
8	1	D1	SCH/30 V	DIODE SCH PWR RECT 2A 30 V AEC-Q101 SOD123F	PMEG3020EH,115	NEXPERIA
9	1	D2	LED/GRN	LED BRIGHT GRN SGL 30 mA 0603	150060VS75000	WURTH ELEKTRONIK EISOS GMBH & CO. KG
10	1	D3	ESD Prot./24 V	DIODE BIDIR CAN BUS ESD PROTECTION 200 W 24 V AEC-Q 101 SOT23	PESD1CAN,215	NEXPERIA
11	1	D4	LED/RED	LED BRIGHT RED CLEAR SGL 2 V 20 mA SMT 0603	150060RS75000	WURTH ELEKTRONIK EISOS GMBH & CO. KG
Inductors						
12	1	L1	100 µH	IND CHK 100 µH 150 mA -30/+50 % AEC-Q200 1812	B82789C0104N002[1]	EPCOS
Resistors						
13	2	R1, R9	1 kΩ	RES MF 1 kΩ 1/10 W 5 % AEC-Q200 0603	CRCW06031K00JNEA	VISHAY INTERTECHNOLOGY (preferred)
					ERJ-3GEYJ103V	PANASONIC (alternative)
					RK73B1JTTD102J	KOA SPEER (alternative)
14	2	R2, R3	62 Ω	RES MF 62 Ω 1/4W 5 % AEC-Q200 1206	CRCW120662R0JNEA	VISHAY INTERTECHNOLOGY
15	2	R4, R10	0 Ω	RES MF ZERO Ω 1/10 W -- AEC-Q200 0603	ERJ-3GEY0R00V	PANASONIC (preferred)
					CRCW06030000Z0EA	VISHAY INTERTECHNOLOGY (alternative)
16	1	R5	not populated			
17	3	R6, R7, R8	10 kΩ	RES MF 10 kΩ 1/10 W 5 % AEC-Q200 0603	ERJ-3GEYJ103V	PANASONIC (preferred)
					RK73B1JTTD103J	KOA SPEER (alternative)
18	1	R11	not populated			
Switches, Connectors, Jumpers, and Test Points						
19	1	J1	HDR_1X2	HDR 1X2 TH 200 MIL SP 338H SN 100L	TSW-202-07-T-S	SAMTEC
20	2	J2, J8	HDR_1X2	HDR 1X2 TH 100 MIL SP 338H SN 100L	TSW-102-07-T-S	SAMTEC
21	1	J3	HDR_1x12	HDR 1X12 TH 100 MIL SP 344H AU 118L	6130121112	WURTH ELEKTRONIK EISOS GMBH & CO. KG
22	1	J4	HDR_2X10	HDR 2X10 TH 100 MIL CTR 428H AU 110L	TSW-110-14-G-D	SAMTEC
23	1	J5	HDR_2X8	HDR 2X8 TH 100 MIL CTR 433H AU 110L	TSW-108-14-G-D	SAMTEC
24	1	J6	CON 3	CON 3 PWR JACK RA TH 295H -- NI 98L	PJ-051A	CUI INC
25	1	J7	not populated			
26	1	SW1	SPST_SWITCH	SW SPST PB TACT 50MA 12 V TH	430186070716	WURTH ELEKTRONIK EISOS GMBH & CO. KG
27	2	TP1, TP2	TEST_040	TEST POINT BLACK 40 MIL DRILL 180 MIL TH 109L	5001	KEYSTONE ELECTRONICS (preferred)
					TP-105-01-00	COMPONENTS CORPORATION (alternative)
					151-203-RC	KOBICONN (alternative)
28	13	TP3 to TP15	TPAD_059	TEST POINT PAD 59 MIL DIA SMT, NO PART TO ORDER	-	-

[1] NXP used the ACT45B-101-2P from TDK for the latest TJA1145A EMC test report.

6 FlexGUI: interactive register control via USB

When the TJA1145A-EVB is plugged onto a suitable microcontroller evaluation board, the microcontroller board can be used as a USB/SPI interface between the TJA1145A-EVB and a PC. After installing the FlexGUI application on a Windows PC (see Section 6.4), the contents of the TJA1145A registers can be viewed and/or changed interactively.

FlexGUI for TJA1145A-EVB currently supports the following microcontroller boards:

- S32K144EVB, Rev. B (Figure 9)
- FRDM-KL25Z, Rev. E or F (Figure 10)

See www.nxp.com for more information about these boards.

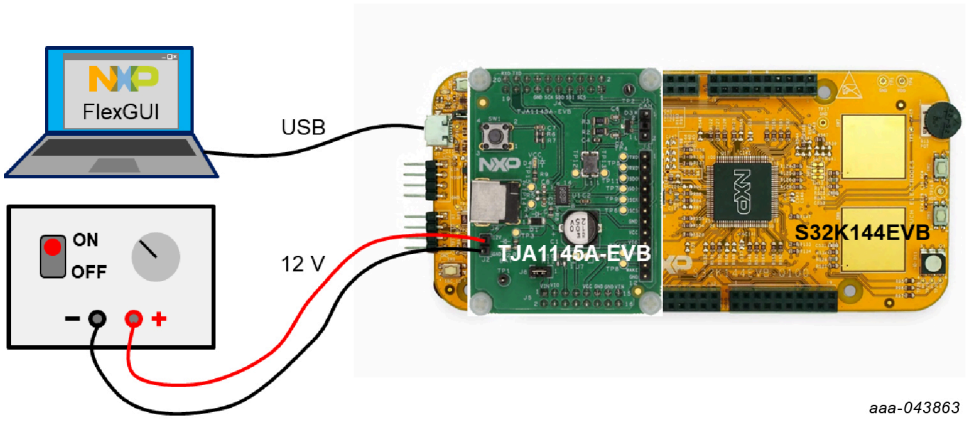


Figure 9. Using FlexGUI with S32K144EVB as USB/SPI interface

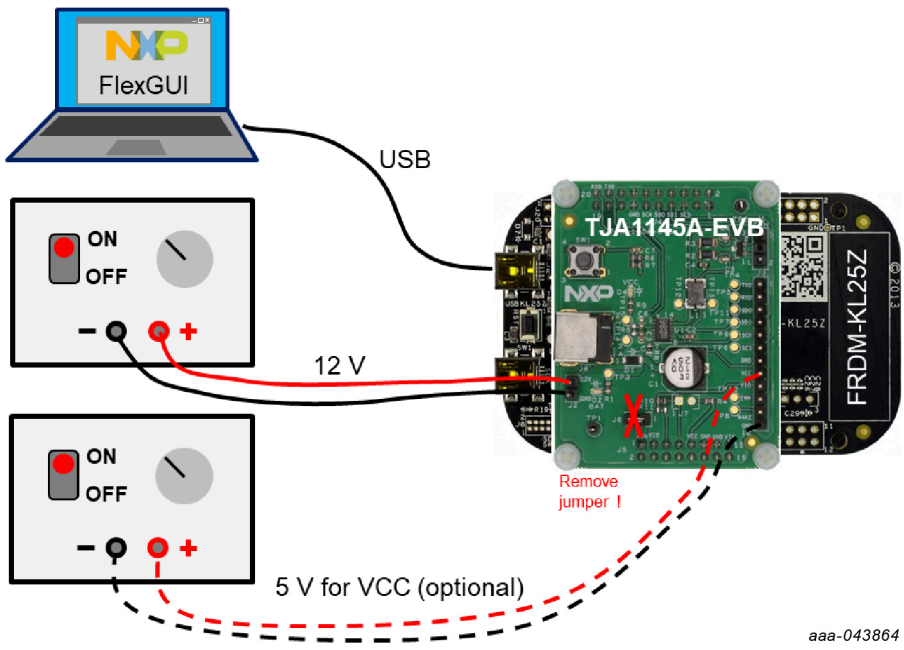


Figure 10. Using FlexGUI with FRDM-KL25Z as USB/SPI interface

6.1 FlexGUI software package overview

The FlexGUI SW package for the TJA1145A-EVB can be downloaded from www.nxp.com. It includes:

- the flexGUI PC installer (see also Section 6.4)
- FlexGUI firmware for all supported microcontroller boards (see also Section 6.2.1 and Section 6.3.1)

6.2 Preparations for using the S32K144EVB as a USB interface

The FlexGUI firmware must be loaded into the S32K144EVB before connecting the TJA1145A-EVB. Note that the jumper settings for firmware programming are different to those for FlexGUI usage.

6.2.1 FlexGUI firmware installation on S32K144EVB

Firmware programming in the S32K144EVB is straightforward:

1. Confirm that the jumpers are in the correct position for firmware programming (Figure 11)
2. Connect the board to the PC with a USB cable (Figure 12)
3. Wait until the PC has launched new drive 'EVB-S32K144' (Figure 13)
4. Copy the firmware file to that drive (Figure 13)

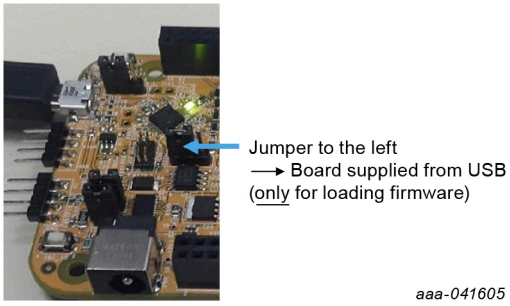


Figure 11. S32K144EVB jumper settings for firmware programming

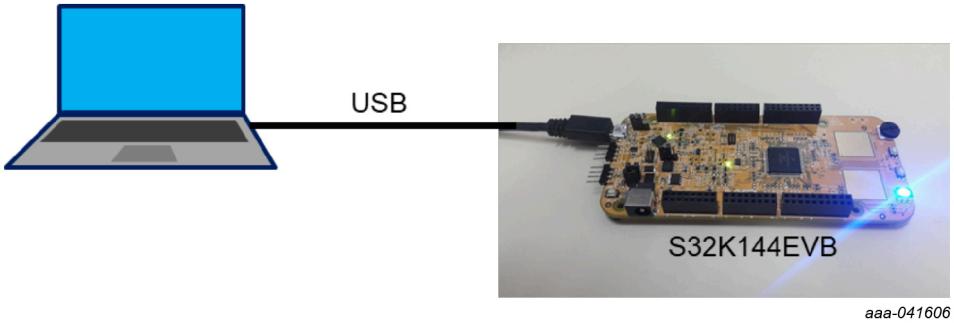


Figure 12. PC connection for firmware programming

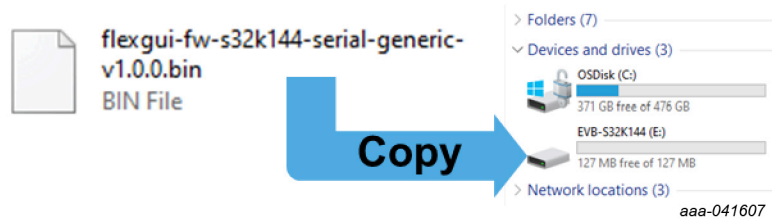


Figure 13. Copying the firmware file to the EVB-S32K144 drive

6.2.2 HW setup for FlexGUI operation

Once the FlexGUI firmware has been installed on the microcontroller board, the jumpers on the boards need to be set as illustrated in Figure 14 before plugging the TJA1145A-EVB into the microcontroller board. The resulting supply and data line interconnections between the boards are shown in Figure 15. Note that only the relevant header interconnections are shown in the schematics extract in Figure 15. See the full board schematics for further details.

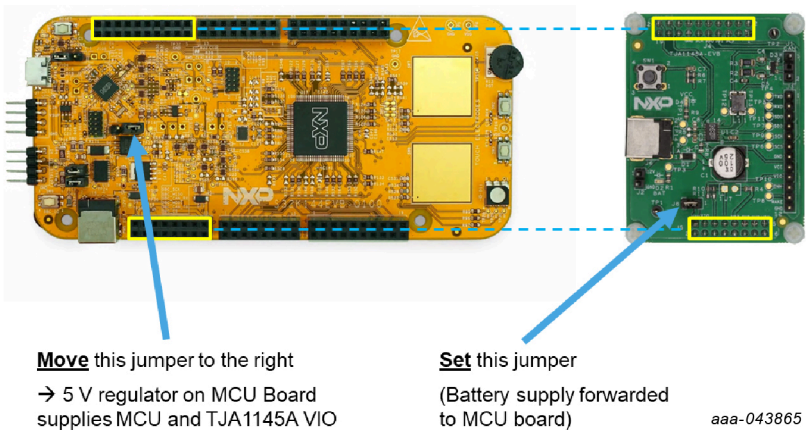


Figure 14. TJA1145A-EVB/S32K144EVB jumper settings for FlexGUI operation

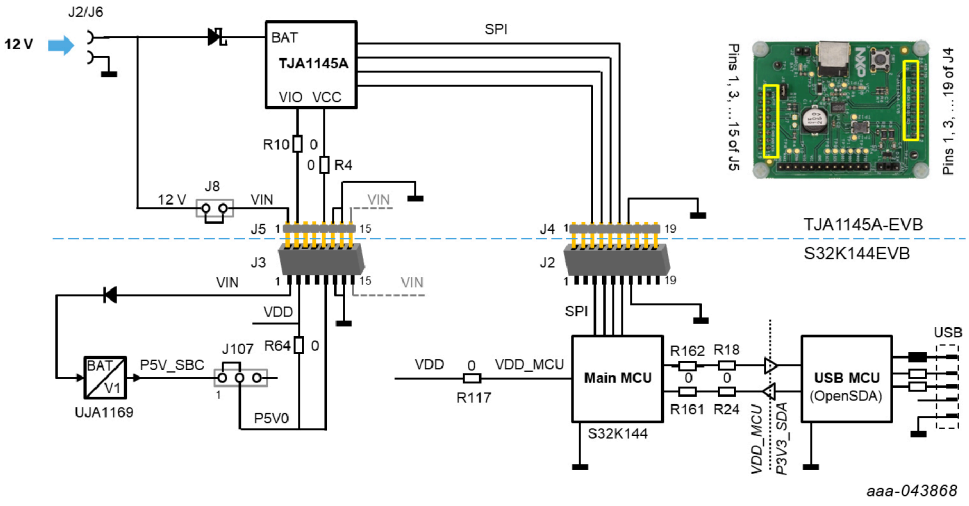


Figure 15. TJA1145A-EVB/S32K144EVB supply and data line interconnections

A 12 V power supply needs to be connected to header J2 or to jack J6 on the TJA1145A-EVB (see also Figure 9). It supplies the BAT pin on the TJA1145A via a polarity protection diode. This supply is also made available at the bottom side header J5 via jumper J8, which is plugged into its counterpart J3 on the S32K144EVB. From there it is rectified and regulated down to a 5 V level. That level is supplied back to the VIO and VCC pins on the TJA1145A-EVB. The main MCU (S32K144) is powered by the same 5 V supply.

The SPI signals are routed directly between the TJA1145A and the main MCU via header J4 and its counterpart J2.

The USB MCU is powered via the USB interface. The signals between the main MCU and the USB MCU are passed through level shifters that serve as a bridge between the two supply domains.

6.3 Preparations for using the FRDM-KL25Z as a USB interface

The FlexGUI firmware must be loaded into the FRDM-KL25Z before connecting the TJA1145A-EVB.

6.3.1 FlexGUI firmware installation on FRDM-KL25Z

Firmware programming in the FRDM-KL25Z is straightforward:

1. Connect the board to the PC with a USB cable (Figure 16), using the 'SDA' USB cable jack
2. Wait until the PC has launched new drive 'FRDM-KL25Z' (Figure 17)
3. Copy the firmware file to that drive (Figure 17)

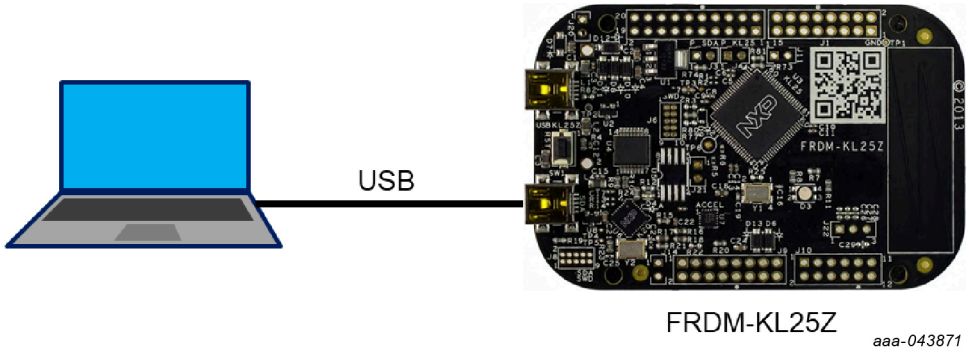


Figure 16. PC connection for firmware programming

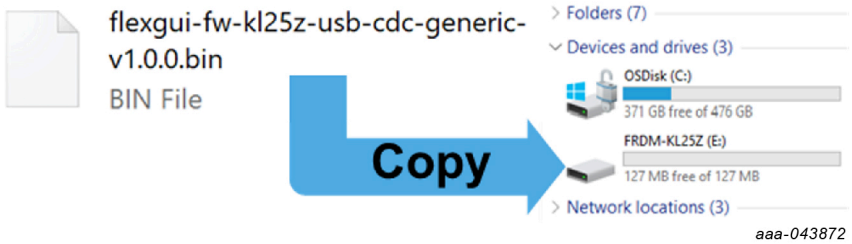


Figure 17. Copying the firmware file to the EVB-S32K144 drive

6.3.2 HW setup for FlexGUI operation

The required jumper settings on the TJA1145A-EVB and the configuration of the interconnection headers on the FRDM-KL25Z are illustrated in Figure 18. Attention should be paid to two important details:

- As further explained below, the end pins of FRDM-KL25Z socket J9 should not be populated, i.e. only a 6-pin single-row socket (or a 2x6-pin dual-row socket) should be used, leaving pins 2 and 16 empty.
- The inner rows of the TJA1145A-EVB bottom headers must be mated with the outer rows of their counterpart sockets on the FRDM-KL25Z. Accordingly, it is sufficient to solder single-row sockets on the outer rows of J2 and J9 on the FRDM-KL25Z.

The resulting supply and data line interconnections between the boards once the TJA1145A-EVB has been plugged into the microcontroller board are illustrated in Figure 19. Note that only the relevant header interconnections are shown in the schematic extract in Figure 19. See the full board schematics for further details.

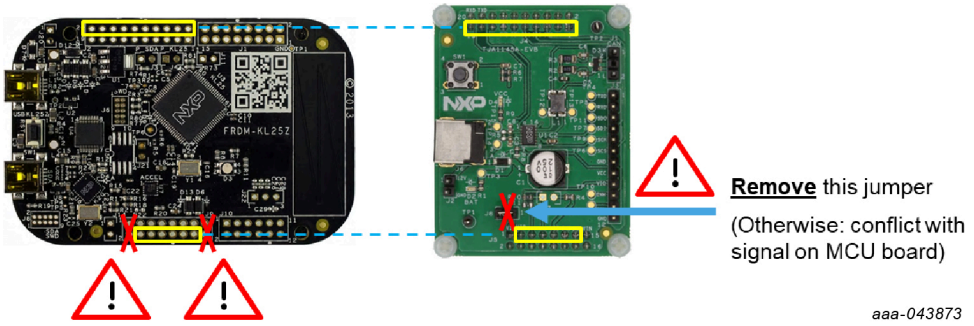


Figure 18. TJA1145A-EVB/FRDM-KL25Z jumper settings for FlexGUI operation

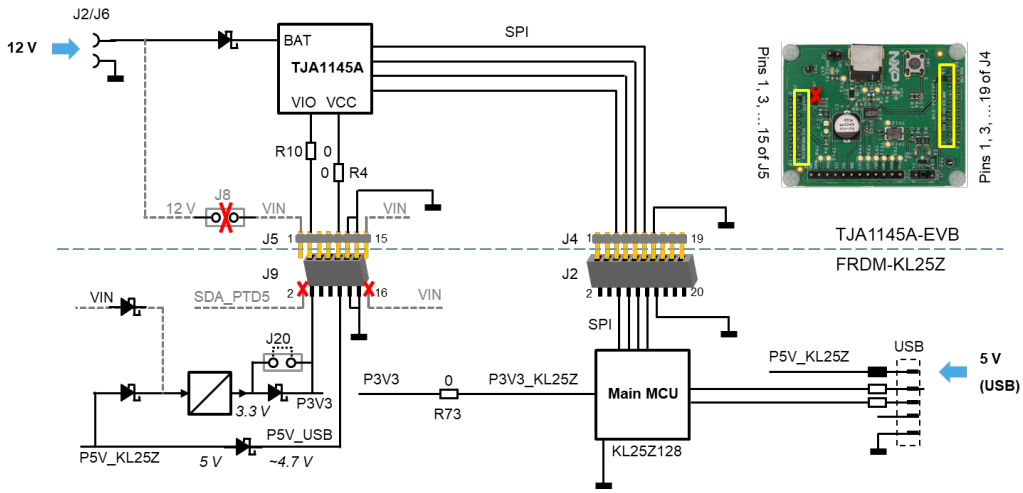


Figure 19. TJA1145A-EVB/FRDM-KL25Z supply and data line interconnections

A 12 V power supply needs to be connected to header J2 or to jack J6 on the TJA1145A-EVB (see also Figure 9). It supplies the BAT pin on the TJA1145A via a polarity protection diode.

On the FRDM-KL25Z, pin 2 of header J9 is routed to a microcontroller port pin, and pin 16 is reserved for the battery supply to the board. However, on the TJA1145A-EVB, the corresponding two pins on header J5 are shorted. It is important, therefore, to prevent interconnections between these pins on the two boards. This can be achieved by using a 6-pin (or 2x6-pin) socket, leaving pins 2 and 16 'empty' on the FRDM-KL25. And to be on the safe side, jumper J8 on TJA1145A-EVB should not be set.

The main microcontroller (KL25Z128) is powered from a 3.3 V voltage regulator, which is itself powered via the USB interface. The Schottky diode in the 3.3 V path can be bypassed with jumper J20 to have the full 3.3 V level on the microcontroller supply pins. The microcontroller supply is also forwarded to the VIO pin on the TJA1145A.

The VCC pin on the TJA1145A is connected to the 5 V USB supply via a Schottky diode. The voltage drop across this diode may cause the VCC level to become too low for proper CAN bus operation. Applying an accurate 5 V level to VCC, with the help of an external power supply, would solve this problem (see also Figure 10).

The SPI signals are routed directly between the TJA1145A and the main MCU via header J4 and its counterpart J2.

Since the main MCU on the FRDM-KL25Z already features USB functionality, the secondary MCU is not involved in FlexGUI operation

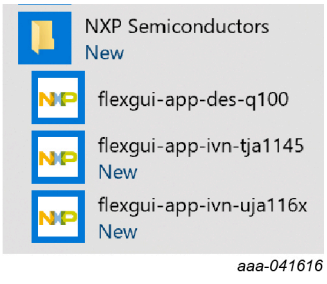
6.4 Installing the FlexGUI on a PC

Double-click on file **flexgui-app-ivn-tja1145-1.0.0.exe** to begin the installation. The FlexGUI application starts automatically after a successful installation.

6.5 Using the FlexGUI

6.5.1 Starting the FlexGUI application

FlexGUI can be started via the Windows Start menu or the shortcut symbol on the desktop (Figure 20).



a. Windows Start menu

b. Desktop shortcut

Figure 20. FlexGUI start options

During start-up, the name of evaluation board(s) supported by FlexGUI installation is displayed in the launch window (Figure 21), along with the short-form type number of its on-board transceiver. The MCU board connected to the USB/SPI interface needs to be selected from the 'Target MCU' drop-down list in the Advanced Settings section of the window.

Click **OK** after selecting the appropriate interface. A temporary pop-up window indicates the status while the FlexGUI configuration is being loaded (Figure 22). Once loading is

complete, the FlexGUI start-up window is displayed (Figure 23). The red text in the lower left corner of the window indicates that the application has not yet established a logical connection to the board. Section 6.5.2 explains how to establish a connection.

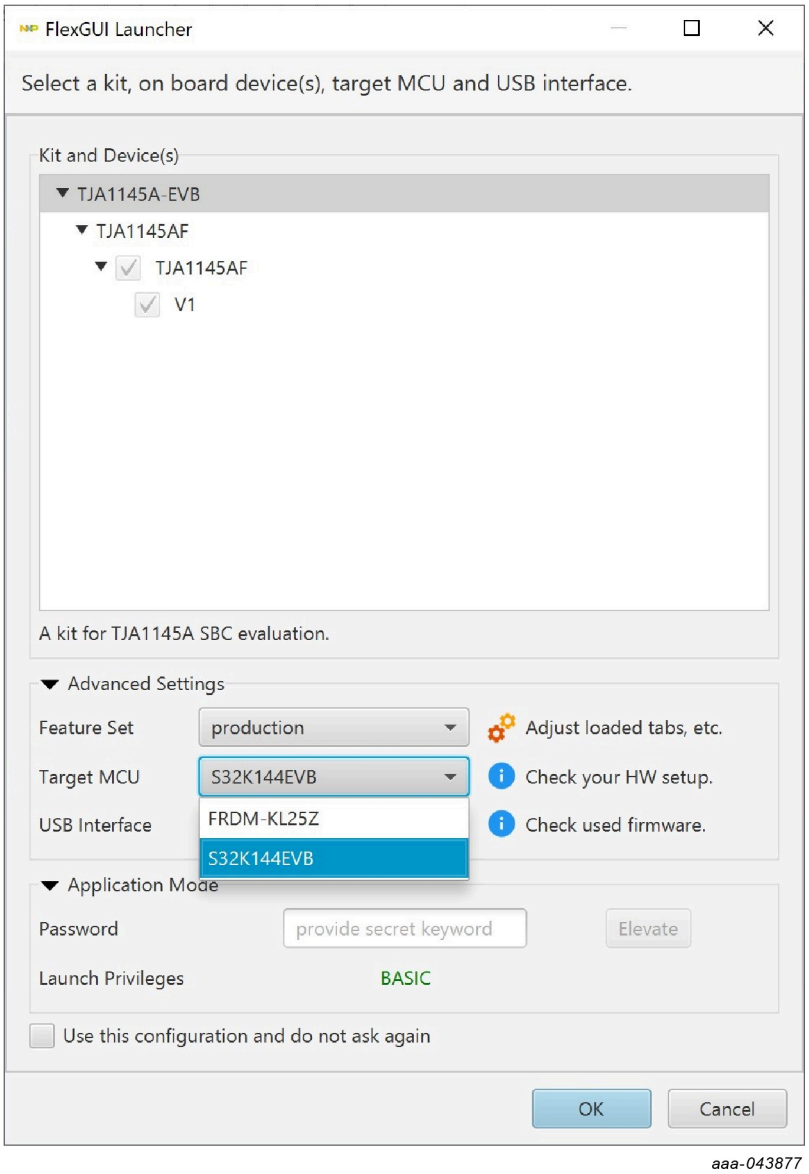


Figure 21. Launch window - EVB selection

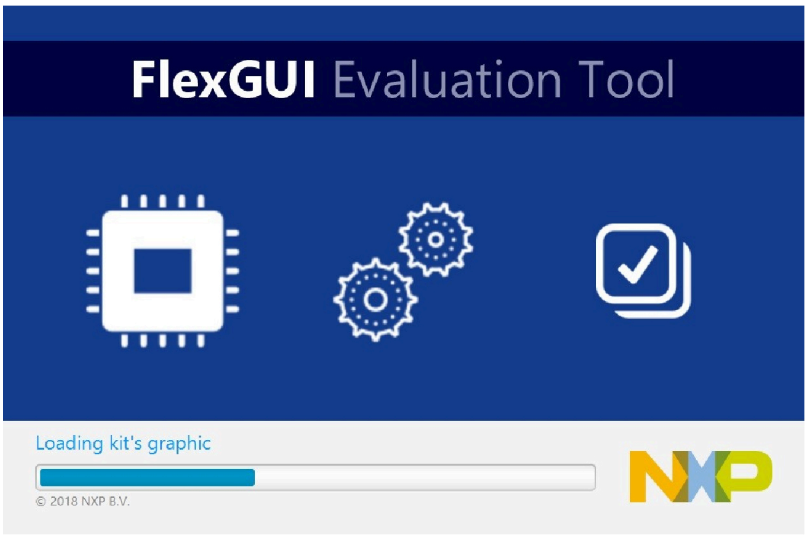


Figure 22. Window displayed during start-up

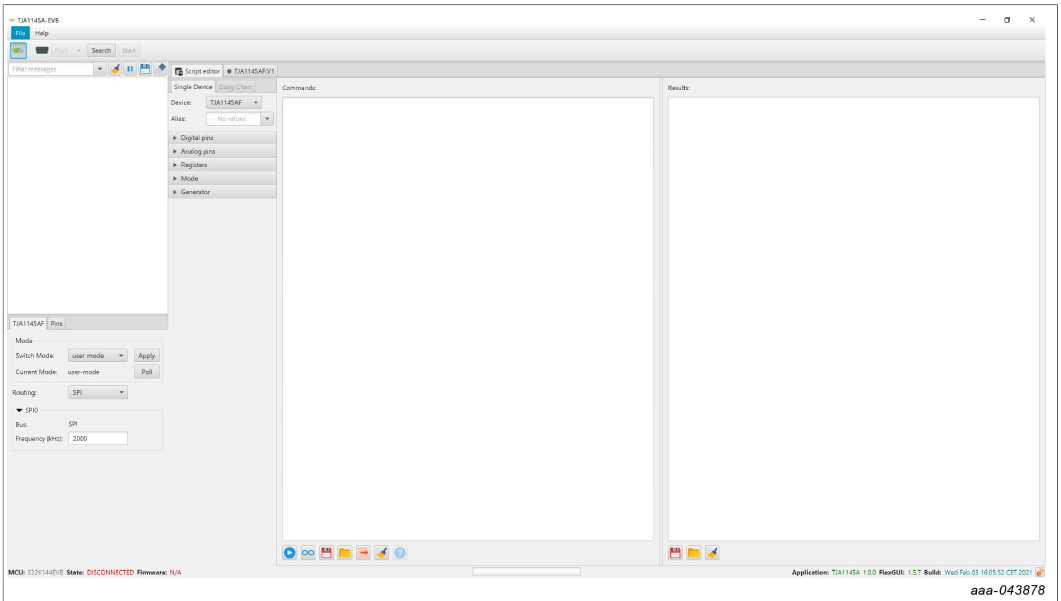


Figure 23. FlexGUI start-up window; boards not yet connected

6.5.2 Establishing a connection between the FlexGUI and the hardware

To establish a connection between the FlexGUI and the hardware, the microcontroller board needs to be connected to the PC with a USB cable (see Figure 9 and Figure 10). When using S32K144EVB, a battery supply needs to be connected to the TJA1145A-EVB before a USB connection can be extablished (see Figure 9). Note that it may take a few seconds for the PC operating system to detect the connection and locate the appropriate USB driver.

Once the connection has been established, a communication session can be started over the USB link:

- Click the **Search** button in the upper left corner of the FlexGUI window (Figure 24) to detect all available serial connections.
- Identify and select the COM port of the board. It is usually the last item on the list if no other USB cables have been connected to the PC since the board was plugged in.
- Click **Start** to enable the connection.

The text in the lower left corner of the window should turn from red to green to indicate that the session has started successfully (Figure 24). The FlexGUI functionality can now be accessed, as described in the following sections.

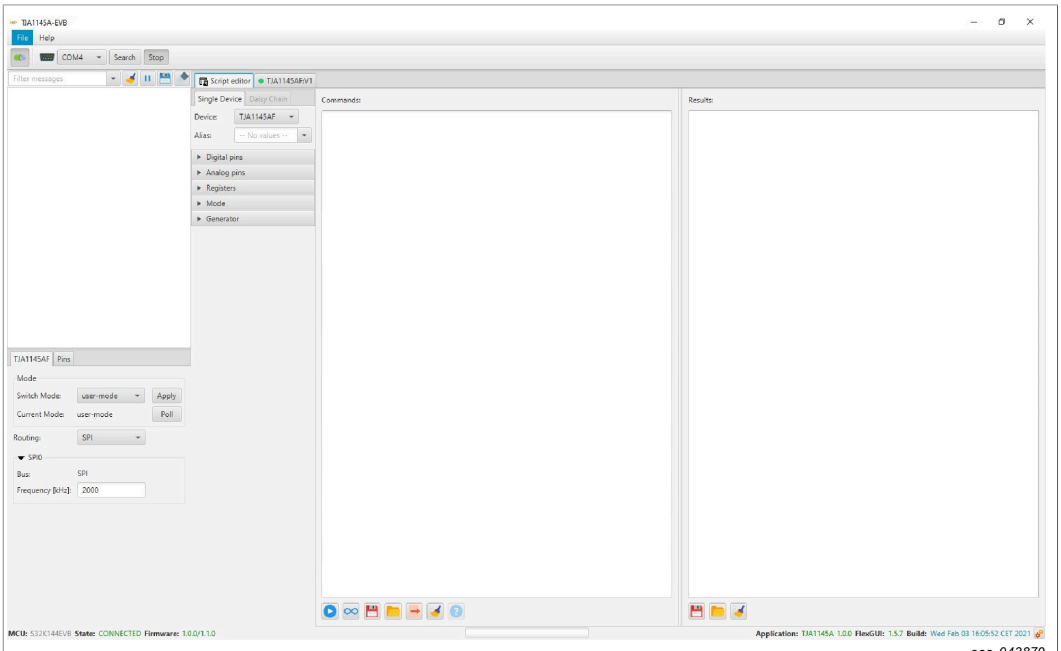


Figure 24. FlexGUI, USB connection to MCU board established

If FlexGUI shuts down during a connection attempt, it may be caused by a conflict due to an obsolete jssc (java simple serial connector) library in the user cache. This problem can be solved by removing that library from the cache, e.g. with the command:

del "%USERPROFILE%\.jssc\windows\jSSC-2.8_x86_64.dll".

When this command is executed (e.g. by double-clicking on a text file that includes this line and has a file extension .cmd), the obsolete library is removed from the cache and a later version of the library is cached the next time FlexGUI starts up.

6.5.3 SPI speed selection

The SPI speed (frequency) can be specified in the lower-left section of the FlexGUI window (if that section of the window is not visible, click the slider symbol under the 'File' menu). Note that the SPI speed needs to be reduced when the TJA1145A is in Sleep mode; the default FlexGUI value may be too high for this mode.

6.5.4 Interactive control of the RGB LED on the microcontroller board

Select the **Pins** tab in the FlexGUI window to access the drop-down window shown in Figure 25. This window contains selection boxes for the microcontroller pins that control the red, green and blue color components of the RGB LED on the microcontroller board. A 'Low' value selects a component; a 'High' value turns it off.

Note that the blue LED on the FRDM-KL25Z board cannot be controlled (it is always on) because the corresponding port pin is already being used for the SPI clock signal to the TJA1145A.

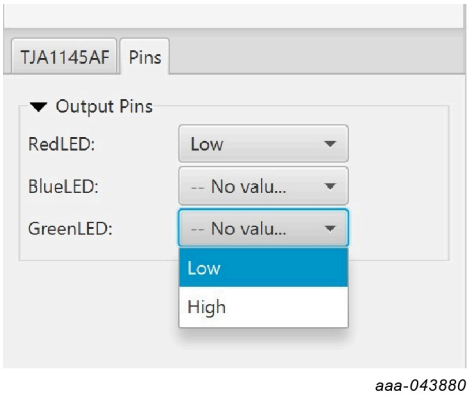


Figure 25. Interactive control of the RGB LED on the microcontroller board

6.5.5 Register map

When the FlexGUI window opens, the 'Script editor' tab is selected by default. Click on **TJA1145AF:V1** tab to display the register map of the selected board. Device registers can be read or written to interactively via this window.

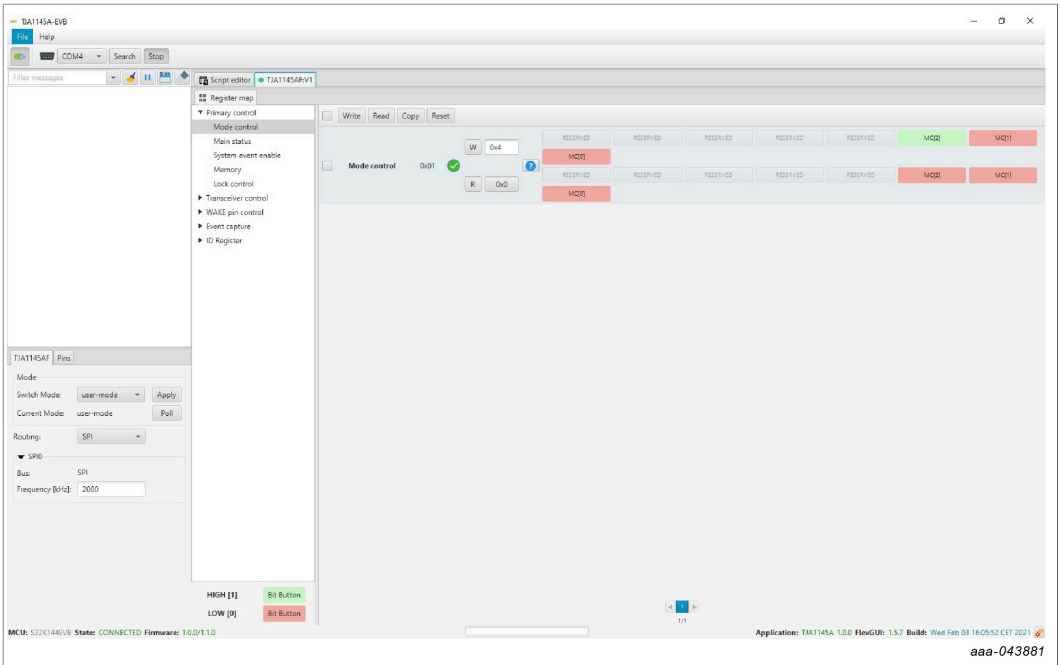


Figure 26. Register map tab

The registers are divided into groups. A register group can be selected in the left column. If option **Tree View** is selected, a single register may be selected (Tree View is selected via the FlexGUI pop up window accessed under File/Settings; see Figure 27).

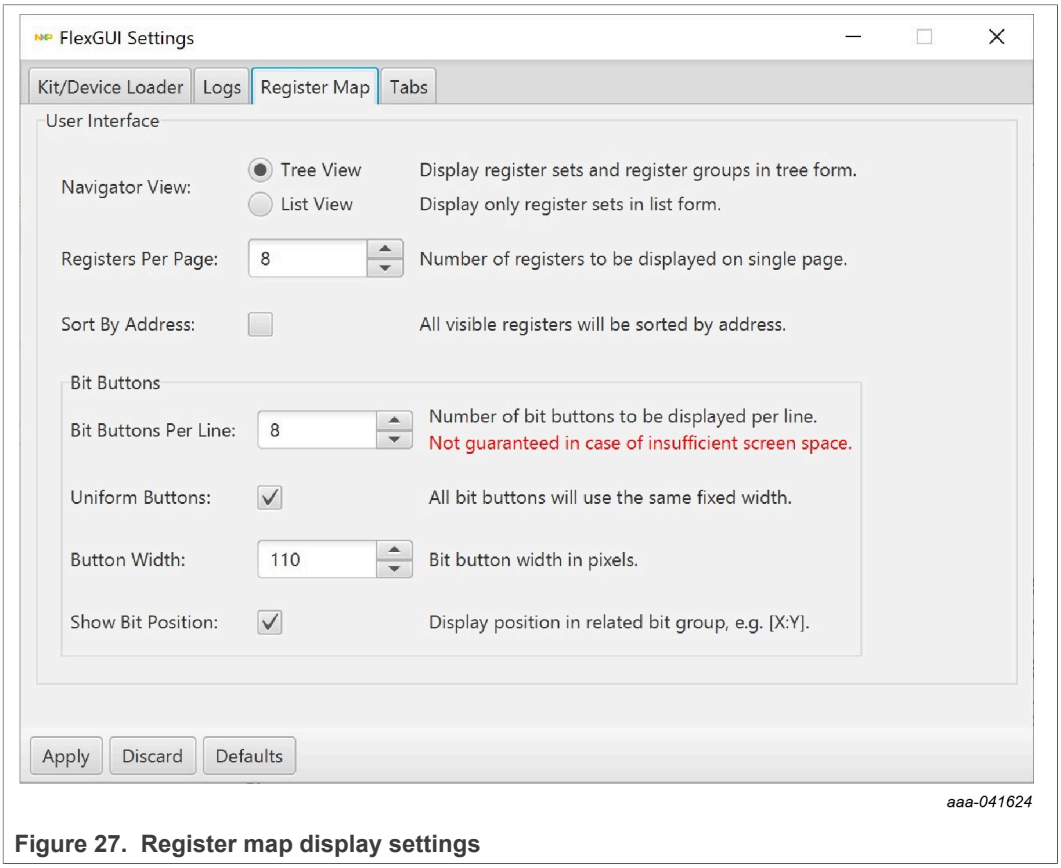


Figure 27. Register map display settings

The contents of the selected register, or register group, are displayed in the main window. Register data can be edited in the top row in preparation for writing to the register. When option **Use Register Init Value** is selected, the editor is initialized with the default values at start-up and reset (Figure 28). If this option is not selected, all bits will be 0 at start-up and reset.

Actual register contents from a prior read access are shown in the bottom row.

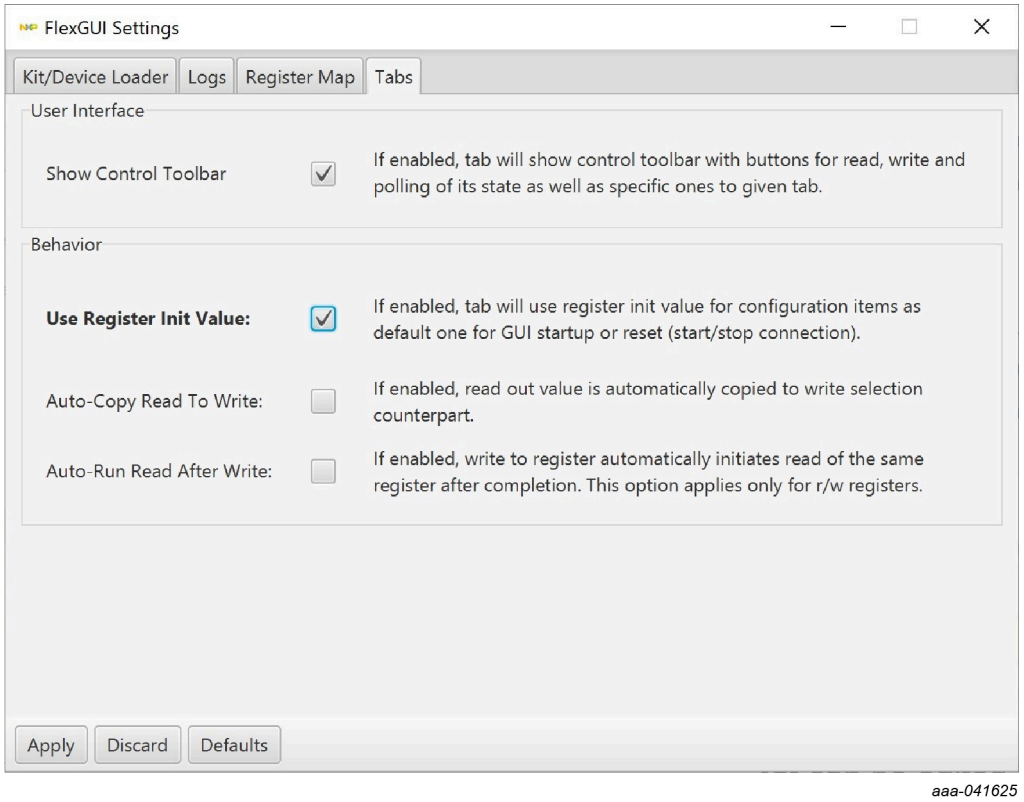


Figure 28. Advanced options for register map

Register data is displayed in three formats:

- As a single hexadecimal value for the entire register
- In text format, when clicking on the question mark symbol
- A color-coded button is provided for each register bit:
  - red = 0
  - green = 1

  When the bit buttons do not fit on a single row (as in Figure 26), try de-selecting checkbox **Uniform Buttons** (see Figure 27). The width of the buttons is then minimized to fit the bit names (Figure 29).

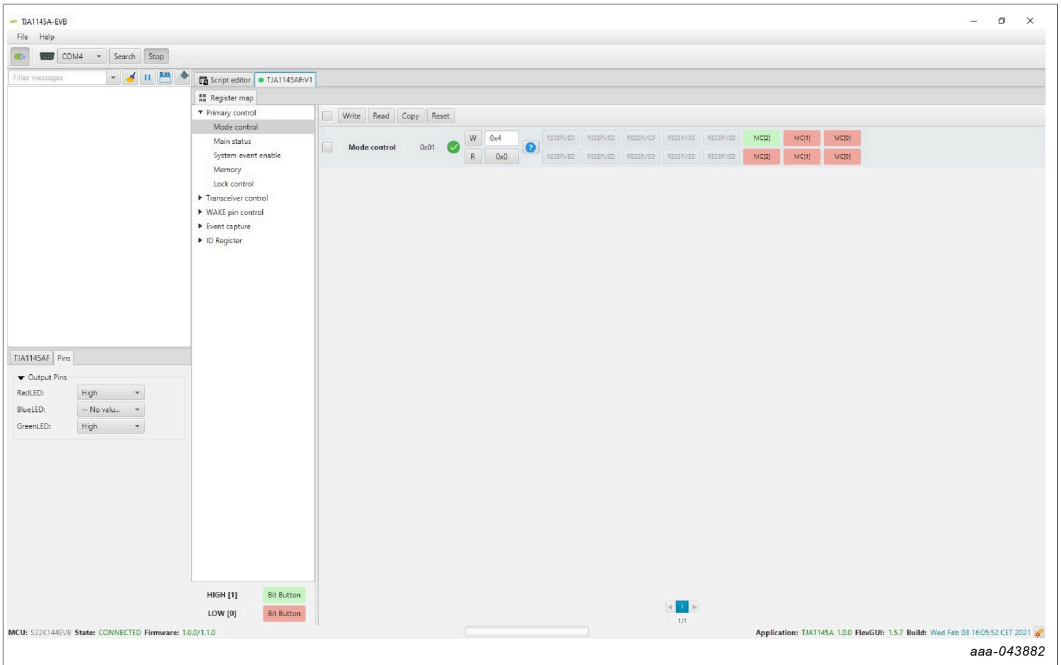


Figure 29. Register map with option 'Uniform Buttons' not selected.

If there are more registers in a group than can be displayed on screen, the registers are distributed over two or more pages and the active page can be selected at the bottom of the main window. The user can also choose the maximum number of registers displayed via control field **Registers Per Page** (see Figure 27). Figure 30 shows an example Register map view displaying the second page of the Transceiver control register group with 11 registers per page.

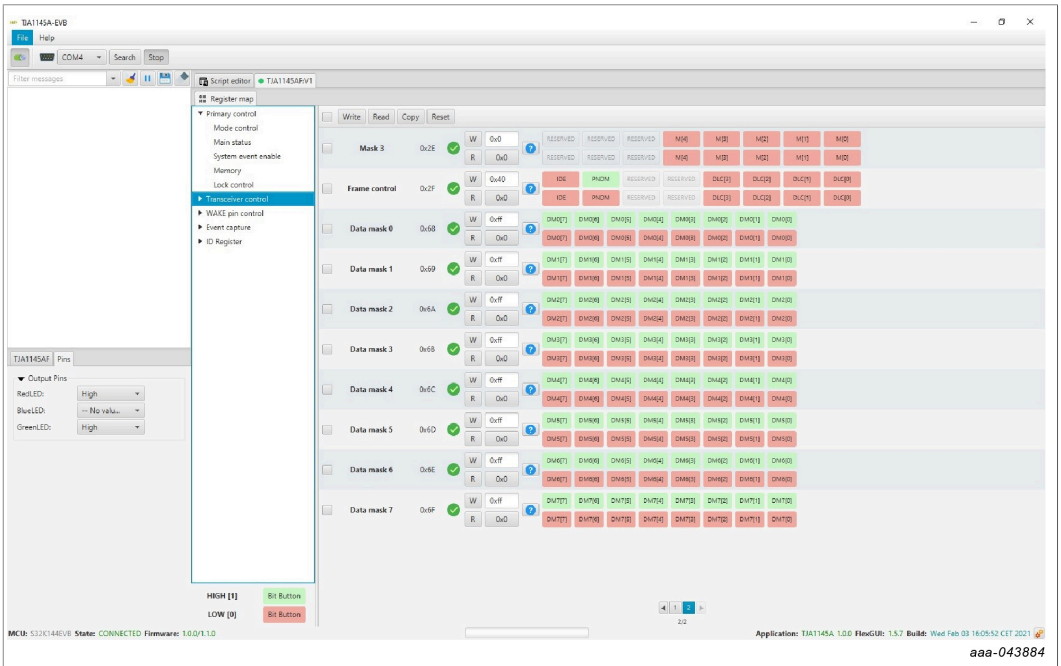


Figure 30. Register map display example with 11 registers per page selected

For each register, read and write operations can be triggered using the **R** and **W** buttons.

Multiple registers can be selected using the check boxes to the left of the register names. The selected registers will be included in later multi-register operations. Four associated buttons are provided:

- Write and read operations can be triggered with the **Write** and **Read** buttons.
- The **Copy** button can be used to copy data from the 'read' row(s) to the selected 'write' row(s).
- Clicking the **Reset** button undoes changes made to the 'write' row(s) since the most recently executed write action(s) on the associated register(s). If a register has not been previously written to, the selected rows are re-initialized (with the default values as selected via the **Use Register Init Value** check box; see Figure 28).

For each register, an 'OK' (✓) or 'pencil' (✎) symbol is displayed to the left of the W/R buttons. The ✓ symbol indicates that the data currently in the editable text field matches the data previously written to the register (or the default initialization values if no previous write operation was executed). A ✎ symbol indicates that the data in the editable text field differs from the data previously written to the register (or from the default values).

6.5.6 Working with the script editor

Selecting tab 'Script editor' opens a tool for creating, executing, loading and saving command sequences ('scripts') used for reading from or writing to registers. RGB LED settings (see Section 6.5.4) can also be included in such scripts.

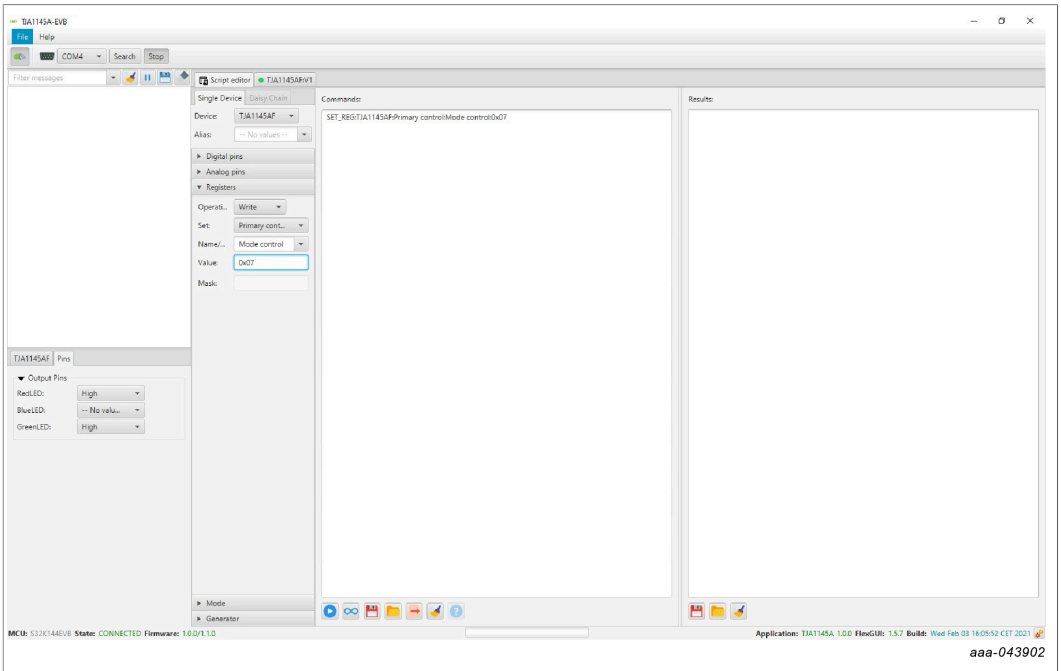


Figure 31. Script editor window

The commands available for the TJA1145A-EVB are listed in Table 9. Commands can be typed directly into the 'Commands' window, or constructed step-by-step using the selector tools in the left column. A script can also be loaded from a file.

Once a script is complete, it can be saved to a file and/or executed once by clicking on the corresponding button (and/or ▸). Help text is displayed when the mouse pointer is hovered above these buttons.

Script execution is logged in the 'Results' window.

If the infinity option (∞) is selected when the script is executed, it runs continuously in a loop. The ▸ button changes to ✋ when a script is running. Execution continues until halted by clicking the ✋ icon. The ∞ option should not be used when the script includes a PAUSE command. If this happens by accident, it may be necessary to abort the FlexGUI application with the help of Windows Task Manager.

Table 9. Syntax for script editor commands

Command	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Purpose
SET_REG	device name	register group	register name	value	write to a register
GET_REG	device name	register group	register name	n/a	read from a register
SET_DPIN	device name	MCU pin name	pin value	n/a	control RGB LED
PAUSE[1]	message text	n/a	n/a	n/a	wait for user
//	comment text	n/a	n/a	n/a	comment

[1] The PAUSE command should not be used when the auto-repeat option (∞) has been selected.

Example scripts using all available commands:

```
// This is an example script
// Do not run this script with the auto-repeat option,
//   because the script includes a PAUSE command

// write value 0x07 to Mode-control register
SET_REG:TJA1145AF:Primary control:Mode control:0x07

// read Global event status register
GET_REG:TJA1145AF:Event capture:Global event status

// turn on red LED
SET_DPIN:TJA1145AF:RedLED:LOW

// give user time to identify the current LED color
PAUSE:RGB LED will change from red to green

// turn off red LED & turn on green LED
SET_DPIN:TJA1145AF:RedLED:HIGH
SET_DPIN:TJA1145AF:GreenLED:LOW
```

6.5.7 Logging read and write operations

Each executed read or write access is logged in the upper left corner of the FlexGUI window. The logged data can be saved to a log file at any time.

A number of **Log Level** filter options are available to tailor the logged data to the needs of the user (see Figure 32). When 'FINEST' is selected, all bits of signals SDI ('out') and SDO ('in') are displayed for each SPI transfer (see script execution example in Figure 33).

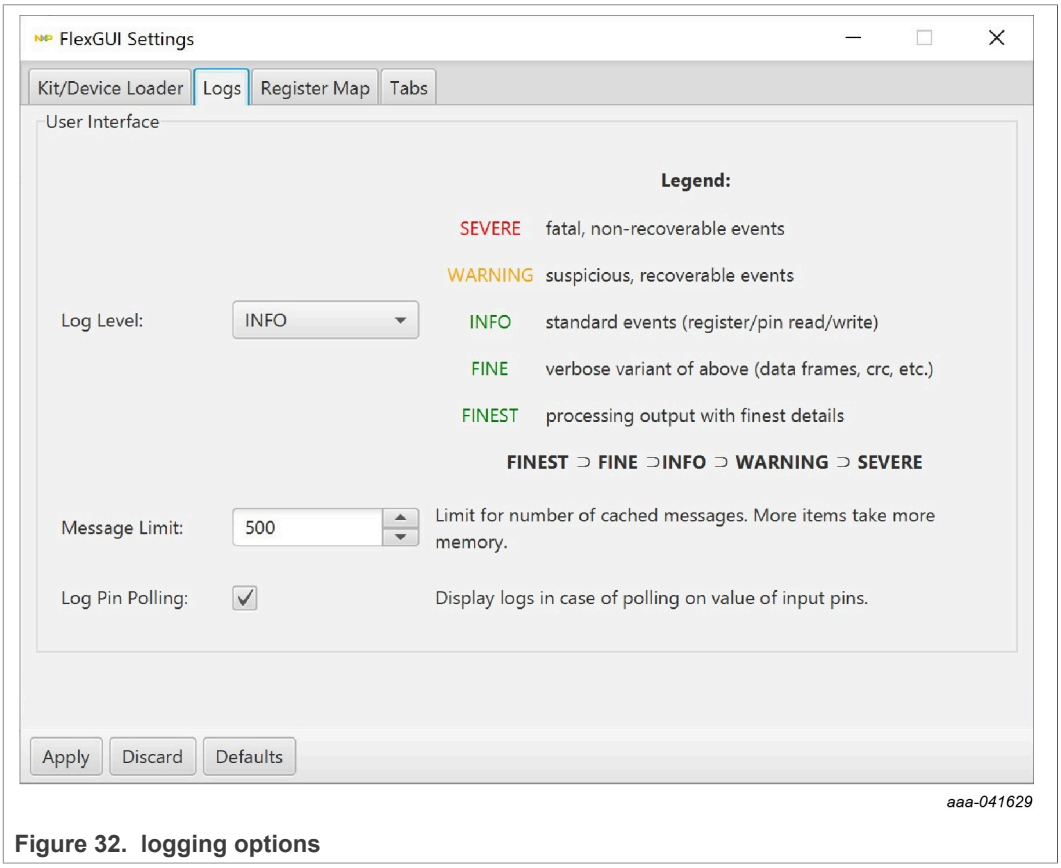
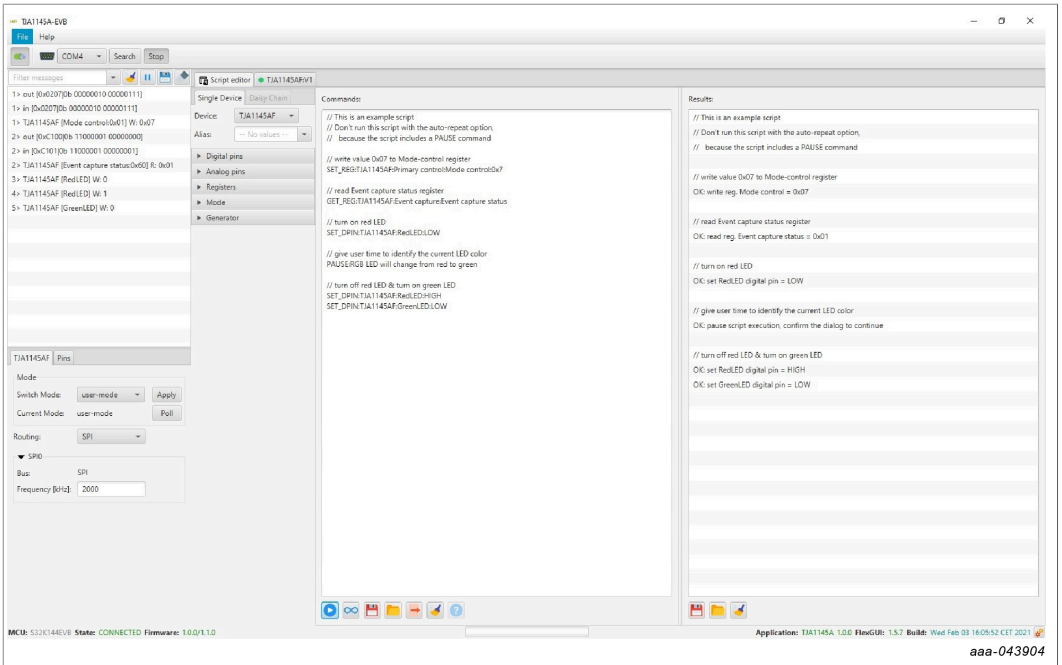


Figure 32. logging options



aaa-043904

Figure 33. FlexGUI 'FINEST' detail level selected for log window

If the log window is not displayed, click on the slider symbol under the "File' menu.

7 References

[1] **TJA1145A data sheet** — High-speed CAN transceiver for partial networking
http://www.nxp.com/TJA1145A

[2] **AH1903 Application Hints** — High-speed CAN transceiver for partial networking TJA1145A, available from NXP Semiconductors

8 Legal information

8.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

8.2 Disclaimers

Limited warranty and liability — 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. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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 life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at 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. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

Translations — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

8.3 Trademarks

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

NXP — wordmark and logo are trademarks of NXP B.V.

Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 14 October 2021
Document identifier: UM11380