

PGA450-Q1 Software Development Guide for the UART and LIN Demo 8051 MCU Firmware

The PGA450-Q1 has an integrated 8051 8-bit microcontroller, which is able to run its firmware based program from the internal DEVRAM or OTP memory available on the device. The 8051 is automatically enabled upon power-up to immediately start running the firmware from OTP memory. If the firmware is loaded to the OTP memory, then the 8051 does not require any additional instruction to run the firmware. However, if the firmware is loaded into the DEVRAM memory, then the user must first reprogram the firmware to DEVRAM memory upon power-up. DEVRAM memory is not able to retain the firmware when power-cycled, but offers the flexibility of modifying the firmware for experimentation.

The orderable “TIDA-00151 Automotive Ultrasonic Sensor Interface IC for Park Assist or Blind Spot Detection Systems” reference design (or PGA450Q1EVM-S) is programmed to run from OTP memory, and enables the UART interface for evaluation purposes. The LIN interface is not enabled in this orderable version of the EVM-S, but can be enabled on a new device by modifying the demo firmware.

The demo firmware is a Keil uVision project composed of the following files:

<i>File Name</i>	<i>Description</i>
pga450_main.c	The main function is where the device first enters the firmware program and runs indefinitely by waiting on interrupt commands to execute any additional functions. The Initialization() function is the first function called at startup, and is only called once to configure the PGA450-Q1 ESFRs, UART port, timers, and interrupt routines. The other commands defined in the _main.c file are the detailed functions for the predefined UART and LIN commands, and are only executed upon receiving a valid input from the master controller.
pga450_init.c	The Initialization() function configures the following in the listed order: <ol style="list-style-type: none"> 1. Initialize the UART, GPIO, and LIN ports. 2. The PGA450-Q1 special function registers are initialized to prepare the device ultrasonic data collection when using the Murata MA58MF14-7N transducer. 3. The downsample rate is set to the maximum to enable long range object detection up to approximately 7m. 4. The LIN port is enabled with an enhanced CRC. 5. Timer0 is enabled as a free running timer for accurate time-of-flight data collection. 6. Timer1 is enabled for 8-bit serial UART communication at 19.2kbps. 7. Interrupts are enabled for the serial UART port and the LIN port. 8. Reload EEPROM contents to EEPROM buffer. <p>Once the Initialization() function is completed, the program returns to the main while-loop to wait for an incoming UART or LIN command interrupt.</p>
pga450_isr.c	The Interrupt Service Routines enabled during the Initialization() function configure the device to monitor the UART and LIN ports for incoming commands from a master controller. In the event that a UART command is received, the serial_ISR() first checks if the incoming bytes from the master are valid. Every predefined command requires that the break_byte=0x00 (first byte), the sync_byte=0x55 (second byte), and the addr_nibble (LSB nibble of third byte) be equal to the value at EEPROM address 0x1F.

	<p>If any of these checks are violated, the serial_ISR() function will abort itself. If the received data check is successful, the third byte is further parsed to check which of the eight pre-defined commands is to be run by referring to the MSB nibble of the third byte. The available demo commands include:</p> <ul style="list-style-type: none"> • Command 0 - Test UART communication. • Command 1 - Trigger a short or long distance burst and capture with predefined drive/filter settings, burst, capture, and compare. • Command 2 - Read the latest instance of the threshold comparison results. • Command 3 - Update an EEPROM value. • Command 4 - Read all FIFO data. • Command 5 - Burst, capture, and compare based on the custom EEPROM configuration. • Command 6 - Report threshold time and level values. • Command 7 - Not Used. Reserved for custom function. <p>Each command expects a certain number of bytes to further process the command in the pga450_main.c file. Refer to the “PGA450Q1EVM-S User's Guide and TIDA-00151 UART Demo Instructional User’s Guide” for details on each command’s input. When a valid command is received, the 8051 executes the command in detail on the _main.c file, and will return data (when applicable) to the master in these functions. The same order of events of receiving valid input from the master and retuning data is true for the LIN interface.</p>
pga450_vars.c	<p>The variables file configures pointers to the FIFO data, external RAM, and EEPROM memories accessed by the 8051. Additional arrays are defined for the UART and LIN input and output data.</p>
pga450.h	<p>This header file maps the external special function register (ESFR) names to a numerical address of the PGA450-Q1 device. The 8051 specific register names of the special function register (SFR) and bit registers are also mapped for ease of use throughout the project.</p>
pga450_vars.h	<p>This header file maps the 8051 pins for UART, LIN, and GPIO operation. The external, and external memory locations are defined, and the PGA450-Q1 specific functions such as Initialization() and commandX() are declared.</p>
STARTUP.A51	<p>When compiling the uVision project for the PGA450-Q1, this file determines if the generated output file will target the PGA450-Q1 DEVRAM or OTP memory. Refer to “PGA450Q1EVM User's Guide” for details on configuring this file for the correct memory type.</p>