Using the LP5562 Four-Channel LED Driver with Programmable Lighting Sequences Evaluation Module

User's Guide

Literature Number: SNVU203A
April 2013–Revised September 2014

Introduction

1.1 Read this First

1.1.1 About This Manual

This user's guide describes the characteristics, operation, and use of the LP5562 Four-Channel LED Driver with Programmable Lighting Sequences evaluation module (EVM). This user's guide includes a schematic diagram and bill of materials (BOM).

1.1.2 Related Documentation from Texas Instruments

LP5562 datasheet SVNS820.

1.1.3 If You Need Assistance

Contact your local TI sales representative.

1.2 General Information

The Texas Instruments LP5562EVM evaluation module (EVM) helps designers evaluate the operation and performance of this device. The LP5562EVM uses the LP5562 LED driver to create special lighting effects for RGB LEDs and/or WLEDs. Information about LED driver characteristics and current ratings of LP5562 can be found in the datasheet.

In order to facilitate ease of testing and evaluation of this circuit, the EVM contains a TI MSP430 microprocessor to provide easy communication via USB. EVM also contains an external power supply connection for the VIN and VEN. Test points for all of the signals can also be found on the evaluation board.

For evaluation purposes, the EVM has been tested over a 2.7V to 5.5V input range. This voltage range is within the absolute maximum input range of the LP5562. Users are cautioned to evaluate their specific operating conditions and choose components with the appropriate voltage ratings before designing this support circuitry into a final product.

Description of LP5562

The LP5562 is a four-channel LED driver designed to produce variety of lighting effects for mobile devices. The device has a program memory for saving programs that enable a variety of automatic lighting sequences. When program memory has been loaded and program set to run mode, the LP5562 can operate independently without processor control.

2.1 Features

- 4 Independently Programmable LED Outputs With 8-bit Current Setting (From 0 mA to 25.5 mA With 100 µA Steps) and 8-bit PWM Control
- Typical LED Output Saturation Voltage 60 mV and Current Matching 1%
- Flexible Control for LED Output PWM
- Automatic Power Save Mode With External Clock
- Three Program Execution Engines With Flexible Instruction Set
- Autonomous Operation With Program Execution Engines
- SRAM Program Memory for Lighting Pattern Programs
- DSBGA 12-bump Package, 0.4 mm Pitch

2.2 Applications

- Fun Lighting
- Indication/Notification Lighting
- Keypad RGB Backlighting and Portable Device Cosmetics

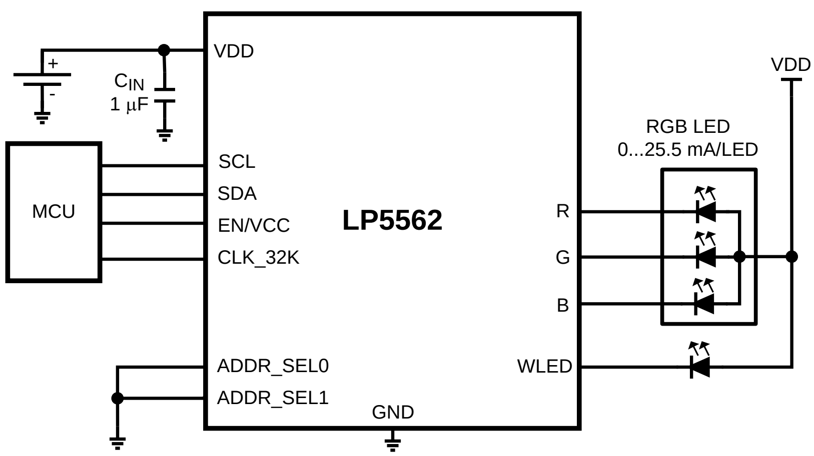


Figure 2-1. Typical Application

2.3 Power Sequences

2.3.1 Startup

The LP5562 is started when VDD is above POR threshold (1.9V typ.), EN signal is high and chip_en bit is set high. Allow 1 ms wait before sending data to the LP5562 after the rising edge of the EN signal. After setting chip_en bit high startup delay is 500 µs (typ).

2.3.2 Shutdown

The LP5562 is shut down if EN signal is set low. When EN signal is set low, chip_en bit is set to 0. If VIN voltage drops below 1.9V (typ.), the device is reset. Reset can also be applied from RESET register by writing FFh.

If the device temperature rises too high, the Thermal Shutdown (TSD) disables the device operation, and the device is in STARTUP mode, until no thermal shutdown event is present.

LP5562 Evaluation Module

3.1 Setup

The LP5562 EVM is connected via USB to the computer. The EVM is controlled with special evaluation software. An MSP430 microcontroller is used with the EVM to provide easy I^2C™ communication, external 32 kHz clock control, and EN -pin control with the LP5562 via USB. The EVM board and LP5562 device are powered by default via USB.

When the board is connected to a computer, Windows should recognize it automatically and start to install the driver. A “Found New Hardware” dialog box will prompt you to locate the missing driver. Select “No, not this time” and continue with “Next”. Select “Install from a list or specific location (Advanced)” to install the driver. Select the directory where the TI_CDC_Virtual_Port driver is. Windows should now install the driver, and the PC can communicate with the evaluation module using a virtual COM port. If Windows cannot find the driver, you need to manually install the TI_CDC_Virtual_Port driver from the Device Manager. There should be a "USB OK" message on the status bar at the bottom of evaluation program, and the red LED should blink on the evaluation board, when the board is recognized. In case the board is not recognized, check the USB address from Windows Control Panel. The USB address should always be less than or equal to 9 (from COM1 to COM9) (see Appendix D). Also switching to another USB port might solve the issue.

A connector for external VDD and VEN is also provided. I^2C communication can be controlled from an external source using pin headers. Test point for all of the signals is provided.

3.2 Evaluation Hardware

The LP5562 evaluation hardware consists basically of two sections:

- LP5562 and the application components: LEDs and input capacitor
- MSP430 microcontroller and its support components

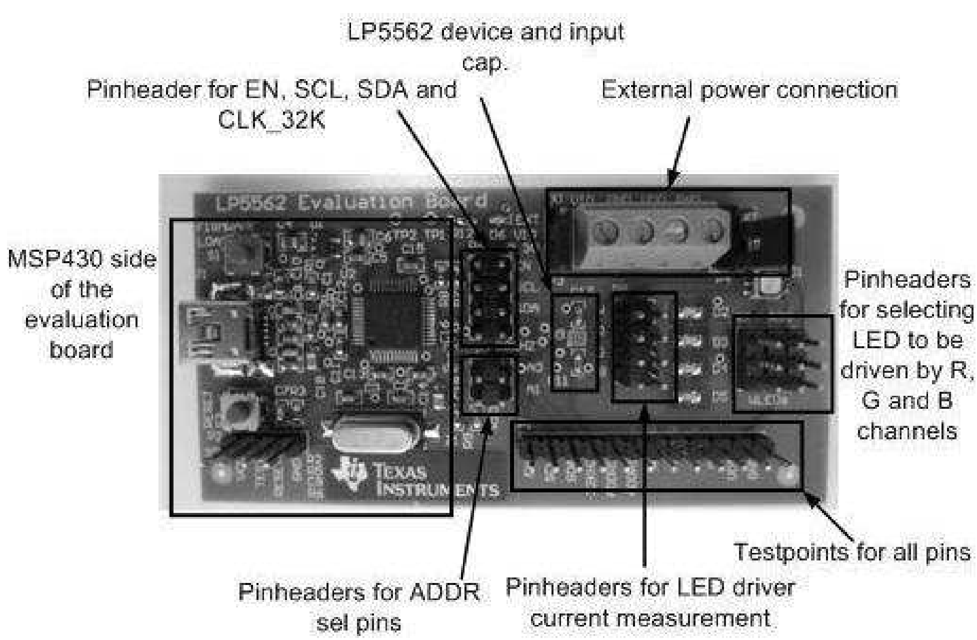


Figure 3-1. Evaluation Hardware

By default the LP5562 is controlled by the MSP430 microcontroller via USB. VDD and VEN voltages come from USB, and the I²C traffic is controlled with microcontroller. The evaluation hardware may also be externally controlled. The VDD and VEN can be fed externally via a connector and with jumper selection. The I²C traffic, EN-pin, and CLK_32K control can be changed from MSP430 control to external control using a pin header. LED driver control can be changed from RGB LED to WLED LEDs, except for the W-LED channel. The pin header enables current measurement to the LED drivers. Device I²C address selection can also be changed with a pin header. For each device pin, there exists a test point (header).

3.3 Evaluation Software

The LP5562 evaluation software helps user to control the evaluation hardware connected to the computer. The evaluation software consists of three sections: tab selection, register selection and register control section. In the tab selection user can switch between Manual, Program and History tabs. In the left-hand side of the evaluation program the register view (see Figure 3-2) is always visible. From this view user can see the register addresses, register names and register values. User can select the register that needs to be changed. Selected register is marked with red X beside the register value. When user selects the register, the selected register can be viewed in detail at the bottom of the evaluation software (see Figure 3-3). This view tells the register address, register name, register default value, register bits and current register value. User can also read and write the register bits by pushing the RD-button (read) and Write-button (write).

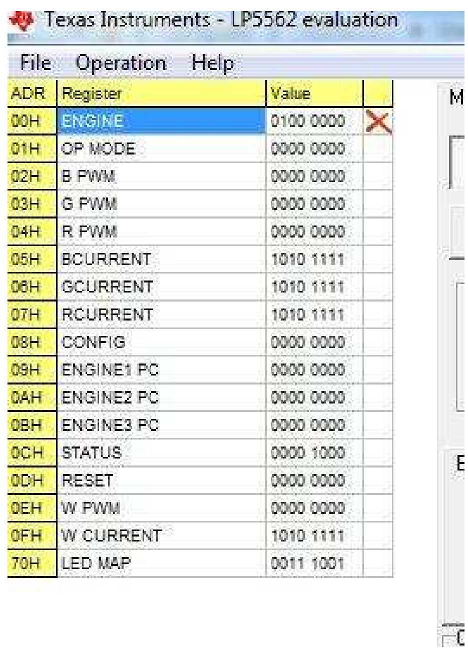


Figure 3-2. Register View

Figure 3-3. Selected Register View

3.3.1 Manual Tab

From the "Manual" tab (see Figure 3-4) user controls all the basic functions of the device:

- "Enable pin" button sets the voltage to EN-pin. When EN is set low, the chip_en bit is cleared, but not other bits.
- "Soft Reset" button resets the device. This creates the RESET register write FFh.
- "Chip enabled" checkbox enables and disables the device with the CHIP_EN bit.
- "I^{2}C address" selection (radio buttons for different addresses). Note that one must adjust the jumper setting on the evaluation board accordingly and that the I^{2}C address is presented in 8-bit format here.
- USB port can be initialized with "Init USB" button.
- "Clock" control (external/internal/automatic clock detection).
- "Powersave" mode enable checkbox enables the powersave mode by setting PS_EN bit.
- "Logarithmic Adjustment" checkbox enables the logarithmic PWM control by setting LOG_EN bit.
- "PWM Clock" frequency selection radio buttons. This sets the PWM_HF bit.
- "Status/Interrupt" bit indicators of external clock and engine interrupts. Values can be updated by RD-button once or checking the scan checkbox for continuous monitoring.
- Control the LED output PWM and current with slider bars. Values are updated when the "Update" button is pushed. These values apply to the I^{2}C registers. The "Clear All" button sets every LED current and PWM value to 0.
- Code memory map is visible here. From this section user can see the SRAM memory contents and alter them. Note that the Read and Write Memory work only when Program Execution Engines are in Load mode.

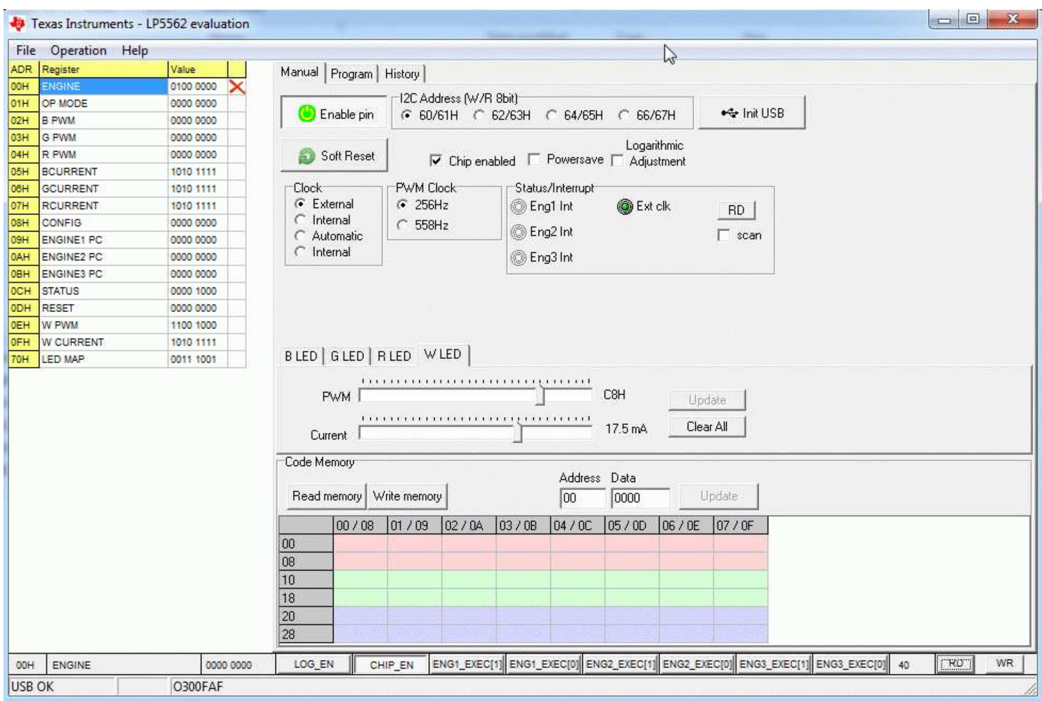


Figure 3-4. Manual Tab of the Evaluation Software

3.3.2 Program Tab

LED lighting programs can be loaded, and engines can be controlled, from the "Program tab" (see Figure 3-5).

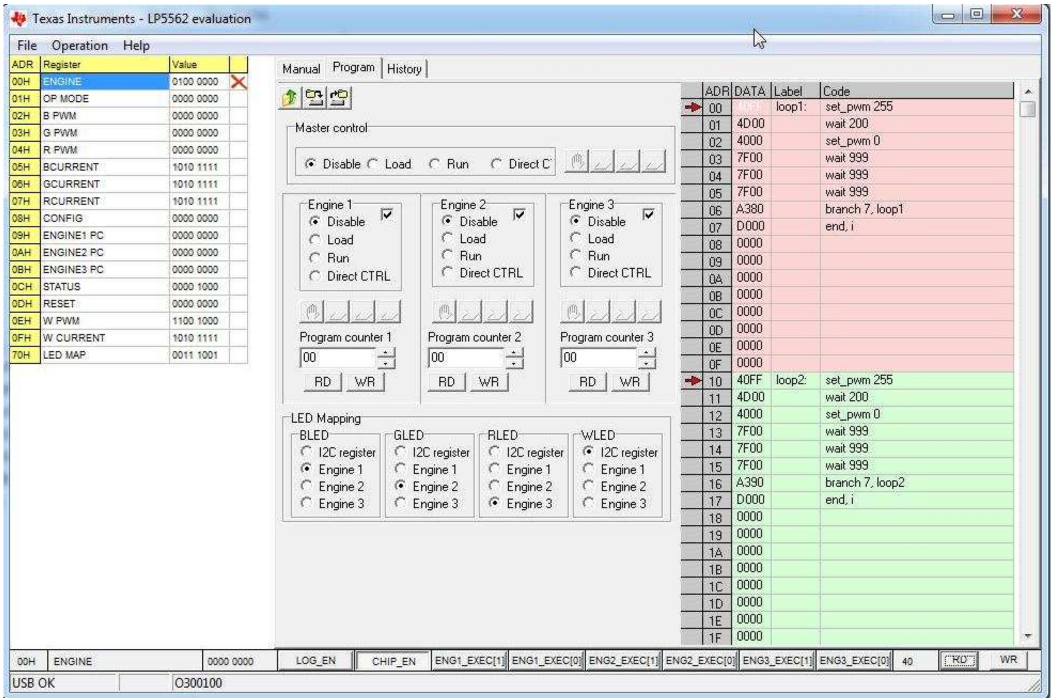


Figure 3-5. Program Tab of the Evaluation Software - Program Is Uploaded Into the LP5562 and Set Running

When downloading a program into the LP5562, user must first select the file to be downloaded. Pushing the leftmost button over the "Master control" section Open file dialog opens (Figure 3-6. User can select the program code (a hex-file) (see Figure 3-7) and push the Open button. After "Open file" dialog disappears, push the Download program into the LP5562 –button (middle button). This automatically sets the Program Execution Engines into right modes and the program gets written into SRAM memory. Also, the Program Execution Engine Program Counters are set to 0. The correct I^{2}C writes for engine modes, and program counters can be seen from the History tab, but the SRAM writes are not visible there. Note that setting the "Load" mode from the Master control section does not download the opened code. If user wants to use the Master control Load option, the "Code Memory" map in Manual tab must be used.

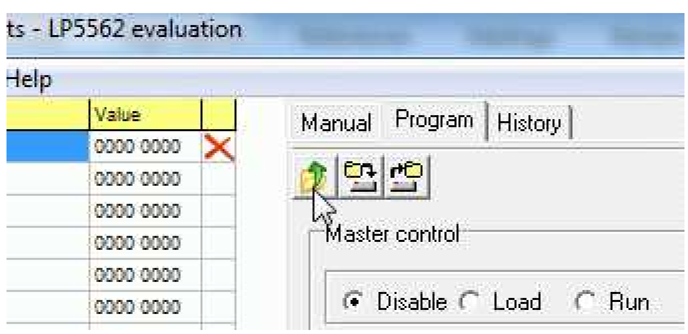


Figure 3-6. Load Source File Button

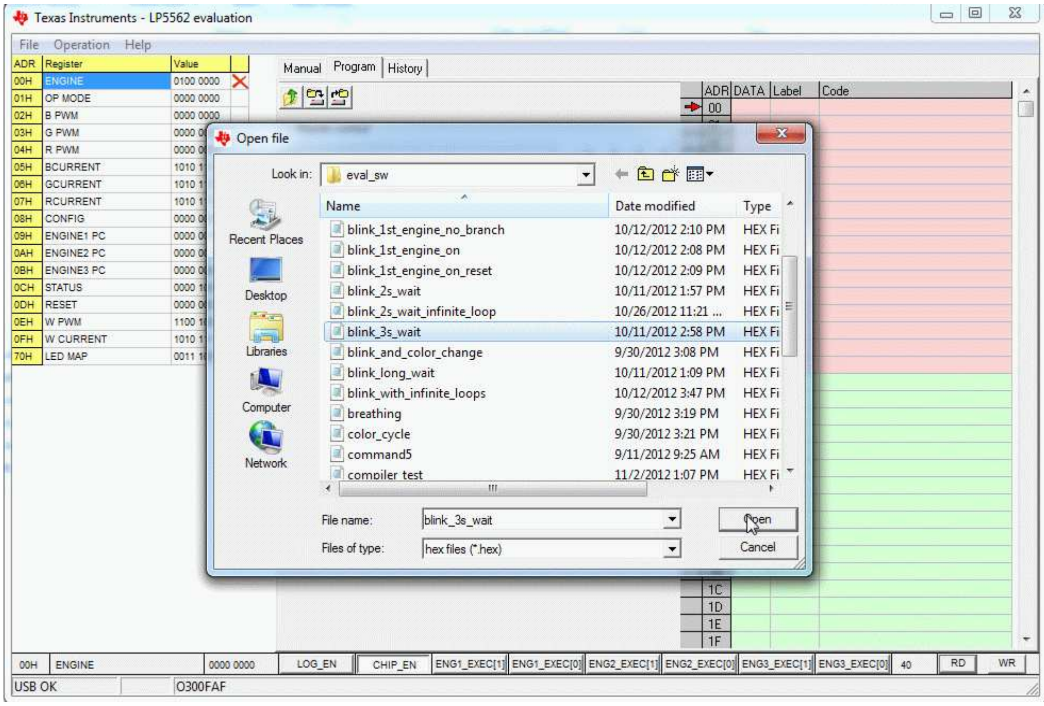


Figure 3-7. Open File Dialog

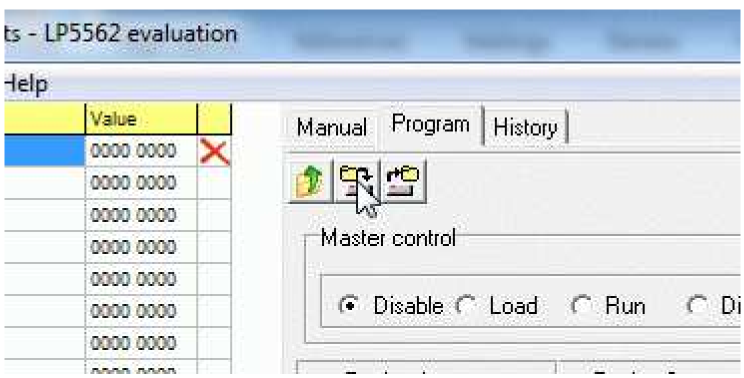


Figure 3-8. Download Program Code Into the LP5562

Once the *.hex code is loaded into the LP5562 the code comes visible at the right-hand side of the "Program" tab. From the code view user can see the code address (note that this is not the SRAM address), code data in hexadecimal labels if they exist in the program, and the code in compiler syntax. The program counter of each engine can be seen as small arrow beside the ADR column. This arrow moves when the Program Counter register is read. The Program Counter register can be read by selecting the register from Register view and pushing the RD-button from the detailed register view at the bottom of the evaluation software. The Program Counter can be read also by pushing RD-button in the Program tab below each engine's Program counter numeric value indicator.

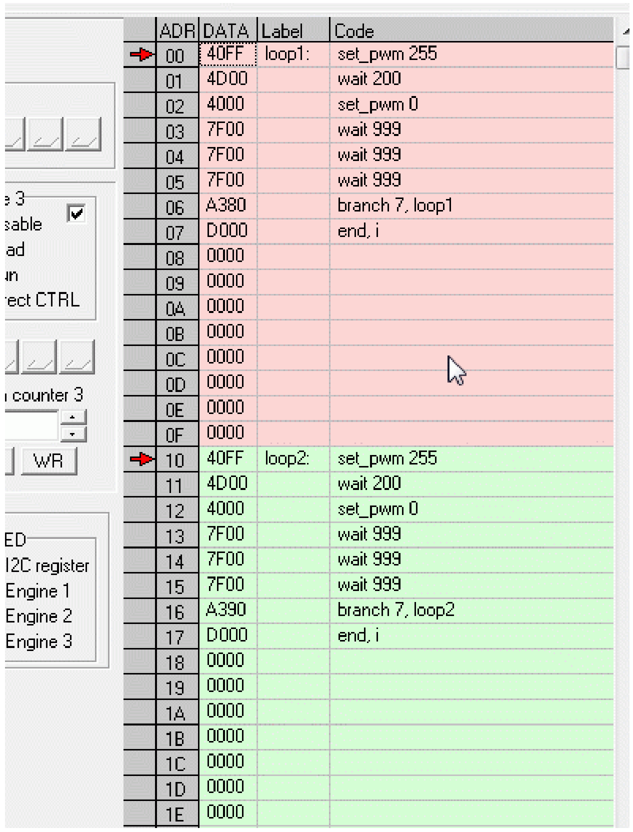


Figure 3-9. Code Section View

In the Program tab there is Master control for all 3 engines and separate controls for each engine. The radio buttons (Disable, Load, Run, Direct Control) control the engine operation mode. The four buttons (Stop, Step, Execute Command and Free Run) control the engine execution states. See tables Table 3-1 and Table 3-2 for operation mode and execution state descriptions.

These same buttons are also available for each individual engine, thus enabling individual control of each engine; i.e., each engine can be run separately. In the same section as the individual engine controls there is a checkbox. When this box is checked the Master control affects that engine. If it is unchecked, the Master control has no effect on that engine.

If user wants to write a program directly to SRAM memory the engine must be set to Load mode. The writes to the SRAM memory are done from the Manual tab. When the address is selected from the table, the data is displayed in the indicator's Address and Data. These can be edited, and the value is updated by pressing the "Update" button. Note that this Update does not write the program memory. The program memory is written by pushing "Write memory" button. With the "Read memory" button, program memory is read to the table. Note that the program memory addresses does not comply with the SRAM I^2C addresses.

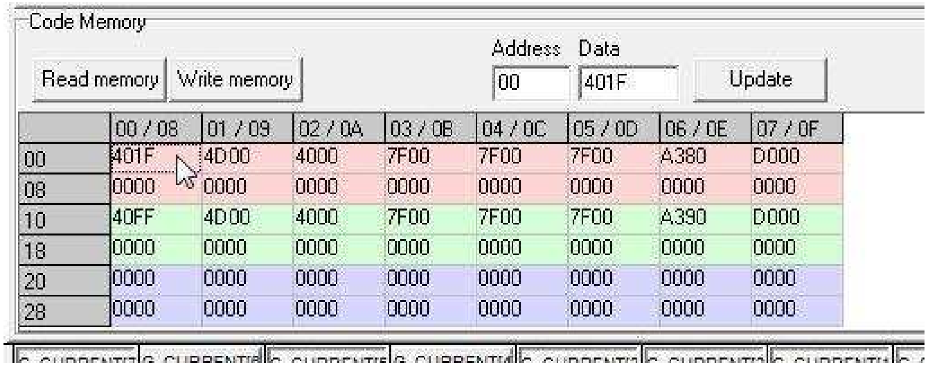


Figure 3-10. Code Memory View

Table 3-1. Operation Modes

Radio button name	Operation Mode	Description
Disable	Disabled	Reset engine program counter
Load	Load	Load program to SRAM, reset engine program counter
Run	Run	Run program defined by engine execution state
Direct Control	Direct Control	Direct control from I^2C PWM register, reset engine program counter

Table 3-2. Engine Execution States

Control button name	Execution State	Description
Stop	Hold	Wait until current command is finished, then stop while execution state is hold. Program counter value can be read or written in this mode
Step	Step	Execute instruction defined by current engine program counter value, increment program counter and change execution state to Hold
Execute Command	Execute instruction	Execute instruction define by current engine program counter value and change execution state to Hold
Free Run	Run	Start program execution from current engine program counter value

LED mapping control is also implemented by the Program tab. LED outputs can be mapped to each of the engines or can be controlled directly from I^2C register (PWM). The selection is done with radio buttons. This selection has an immediate effect, and it is done by writing to LED MAP register 70h.

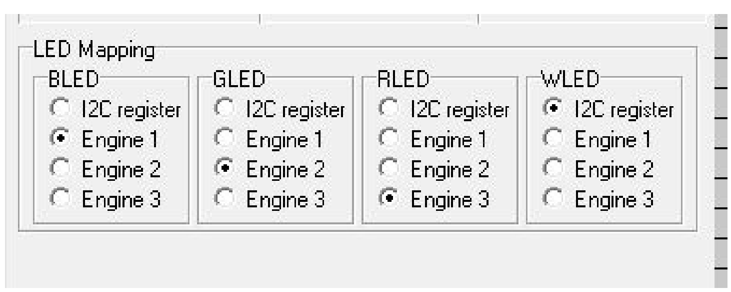


Figure 3-11. LED Mapping

3.3.3 History Tab

The "History" tab (see Figure 3-12) provides information on the I^{2}C writes used to configure/control the LP5562 device. This is a good way of debugging or finding out what writes/reads have been done. However, the history tab does not record the SRAM memory writes when downloading a program.

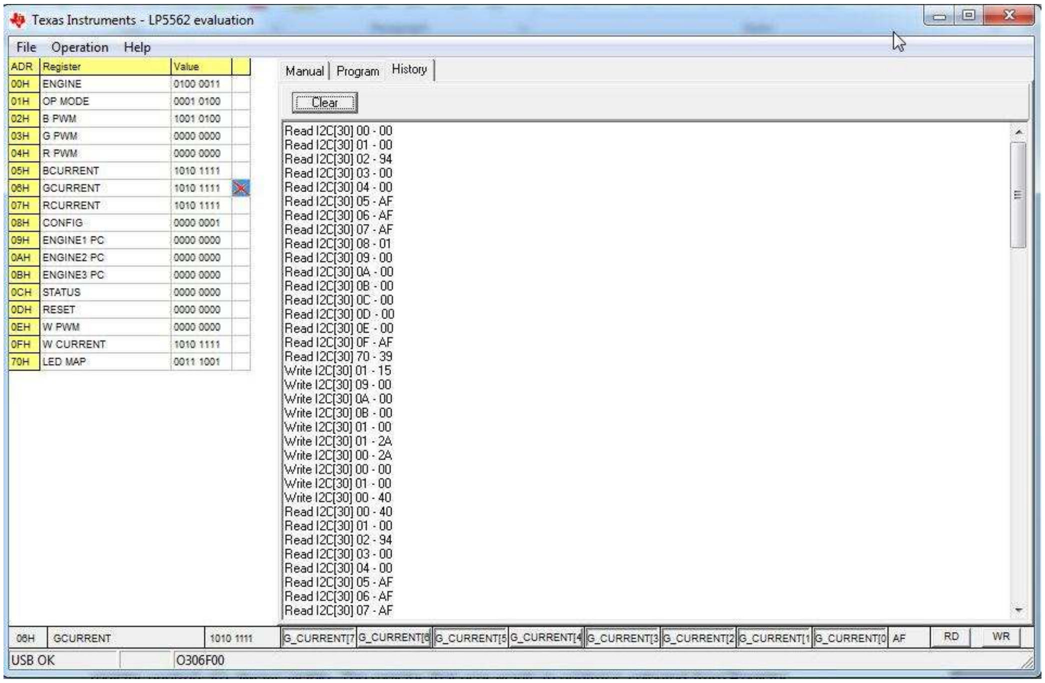


Figure 3-12. History Tab

3.4 Command Compiler

A command compiler is used to write LED sequences for the LP5562. The command compiler is a simple-to-use Windows program with a graphic-user interface.

User can write his own memory files by using command compiler's editor or any text editor (file must have .src extension). Command compiler translates ASCII memory files into binary file (.bin), or hex files (.hex and .he2). In the "Format" menu user can select between .bin, .he2 and .hex formats. Evaluation software uses .hex format, but .bin and .he2 formats can be used for user's own applications. To start command compiler, double-click the compiler icon (compiler.exe). Source file can be opened from File/Open menu. Source files have .src extension. When source tab has been selected, the source code is visible and can be modified, compiled, and saved. File must be created first to be able to edit it using the File/New menu or, for example, to open some demo sequence, and save with another name. Below is the syntax and explanation for command compiler commands. Program can be compiled to binary or hex formats with compile menu. After compilation, command list can be seen by clicking "List" tab. Compilation generates binary/hex/he2 file and list file. Compiled file has 16 commands for each channel for a total of 48 commands. File represents the whole SRAM program memory content.

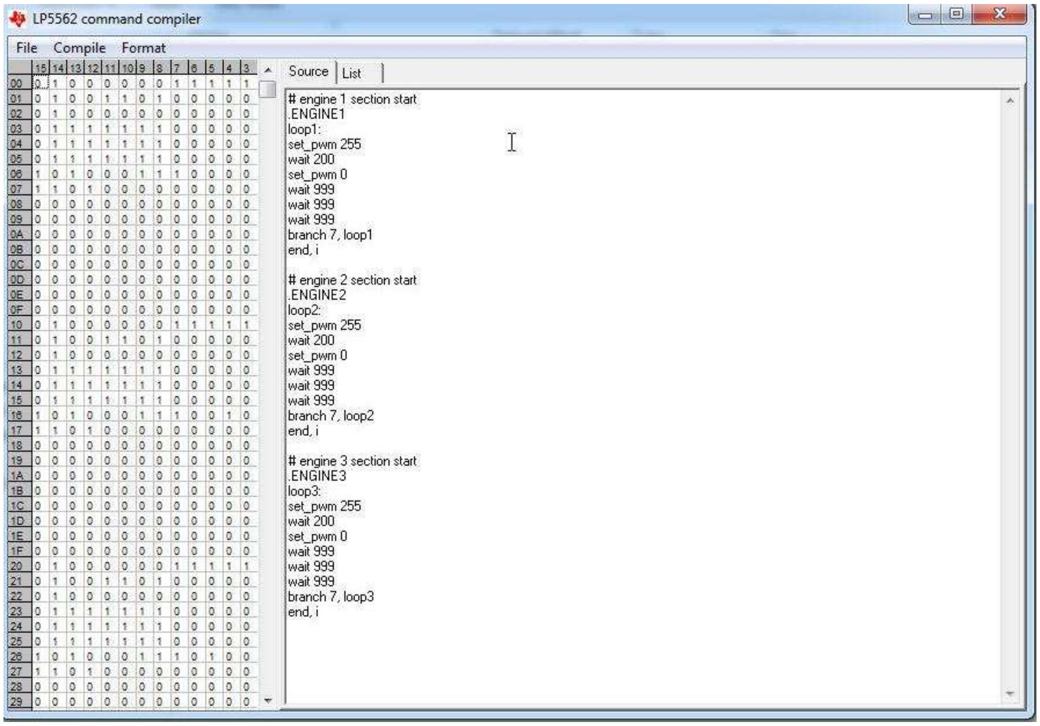


Figure 3-13. Command Compiler Source View

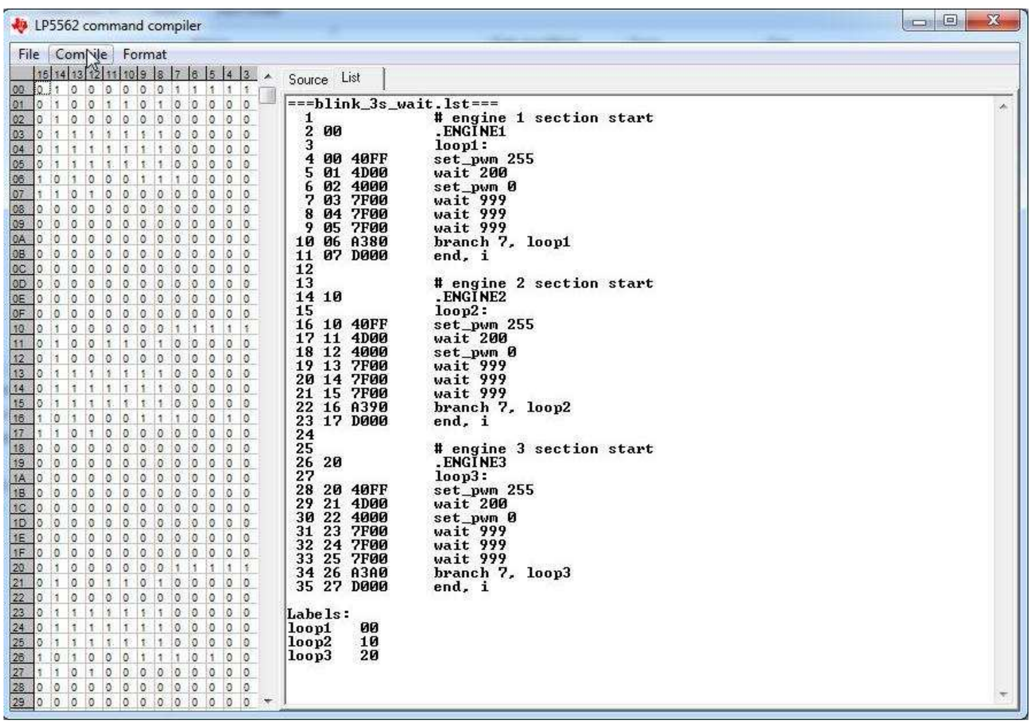


Figure 3-14. Command Compiler List View

3.4.1 Syntax

Comment – Line starting with “#” symbol. Example: # engine 1 section start

Sections – Engine1, engine2 or engine3 section starts with: .ENGINE1, .ENGINE2 or .ENGINE3. Example: .ENGINE1

Labels – Any words with colon (:) as ending symbol. Example: label1:

3.4.2 Commands

Ramp: Ramp command generates a PWM ramp from current value. Ramp command has two parameters – first is time in milliseconds (floating point format, maximum execution time t_{MAX} = (1000 ms x number of steps) - 1 ms) and second is number of steps (positive or negative, integer 2-128) separated with comma. Example: ramp 20.5,6

Wait: With wait command program execution stops for time defined. Command has one parameter, time in milliseconds (floating point format, maximum 999). Example: wait 50.5

Branch: Branch command loads step number to program counter. Branch command has two parameters, loop counter (integer 0-63, 0 means infinite loop) and label separated with comma. Label must be predefined before using in a branch command. The following example loops 5 times commands between label1 and branch command: Example: label1: … branch 5,label1

Set_PWM: Set_pwm command sets PWM output value. Command has one parameter, PWM value (integer 0-255). Example: set_pwm 23

Start: (Go to) Start command resets program counter and continues executing from the beginning of section. No parameters used. Example: start

Trigger: Trigger command sets wait or send trigger. Command has two parameters, wait trigger channel and send trigger channel. Channels are defined as: 1 = engine1, 2 = engine2, 3 = engine3. Examples: trigger s1 => (Send trigger to engine1); trigger s2 => (Send trigger to engine2); trigger s3 => (Send trigger to engine3); trigger s23 => (Send trigger to engine2 and engine3); trigger w1 => (Wait trigger from engine1); trigger s1,w3 => (Send trigger to engine1 and wait trigger from engine3)

End: End command ends program execution. Also interrupt signal can be send or program counter can be reset. Command can have up to two parameters. I = interrupt send, R = reset program counter. Examples: end I; end R; end R,I; end I,R (same as earlier); end

3.4.3 Errors

If there is an error during compilation an error message is generated. Error messages are as follows:

1 = engine section error

2 = syntax error

3 = ramp parameter error

4 = SRAM memory overflow

6 = ramp step error

7 = branch error

9 = set_pwm parameter error

11 = wait parameter error

3.4.4 Files

There are five files that can be created with the command compiler. The source file has .src extension and is in source code format. During compilation listing file with .lst extension is created always. Also binary file with .bin extension or hexadecimal files with .hex and .he2 extension can be generated during compilation. What file from *.bin, *.hex or *.he2 is compiled can be selected from Format menu. File name for the list, binary, and hexadecimal files is the same as for source file name.

- List-file (*.lst) contains command address, command in hex format and commands in compiler format. This format can be seen also in evaluation program Program tab, when the code is loaded into the LP5562. Below is an example of a beginning from a program code list file:
  1. ===blink_3s_wait.lst===
  2. 1 # engine 1 section start
  3. 2 00 .ENGINE1
  4. 3 loop1:
  5. 4 00 40FF set_pwm 255
  6. 5 01 4D00 wait 200
  7. 6 02 01F2 ramp 5, -115
- Binary-file (*.bin) contains all commands in a 16-bit binary format. Each command is on its own row. Below is an example of a beginning from a program code binary file (8 commands visible):
  1. 0100000011111111
  2. 0100110100000000
  3. 0100000000000000
  4. 0111111100000000
  5. 0111111100000000
  6. 0111111100000000
  7. 1010001110000000
  8. 1101000000000000
- Hexadecimal-file (*.hex) contains all commands in hexadecimal format. There are four commands in one row. Commands are split to a 8-bit format separated with space character. Below is an example of a beginning from a program code hexadecimal file (12 commands visible):
  1. 40 FF 4D 00 01 F2 40 00
  2. 7F 00 7F 00 7F 00 A3 80
  3. D0 00 00 00 00 00 00 00
- Hexadecimal-file (*.he2) contains all commands in hexadecimal format. There are four commands in one row. Commands are split to a 8-bit format separated with comma and space characters. Below is an example of a beginning from a program code hexadecimal file (12 commands visible):
  1. 0x40, 0xFF, 0x4D, 0x00, 0x40, 0x00, 0x7F, 0x00,
  2. 0x7F, 0x00, 0x7F, 0x00, 0xA3, 0x90, 0xD0, 0x00,
  3. 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,

3.4.5 Example Command File

See more code examples in Appendix B

```
# engine 1 section start
.ENGINE1
ramp 20.5,6
ramp 10,-15
wait 10
# engine 2 section start
.ENGINE2
ramp 10,-15
start
# engine 3 section start
.ENGINE3
ramp 10,15
```

Schematics and Bill Of material

NOTE: Please see the following pages for detailed schematic and Bill of Materials.

A.1 LP5562 Evaluation Board Schematics

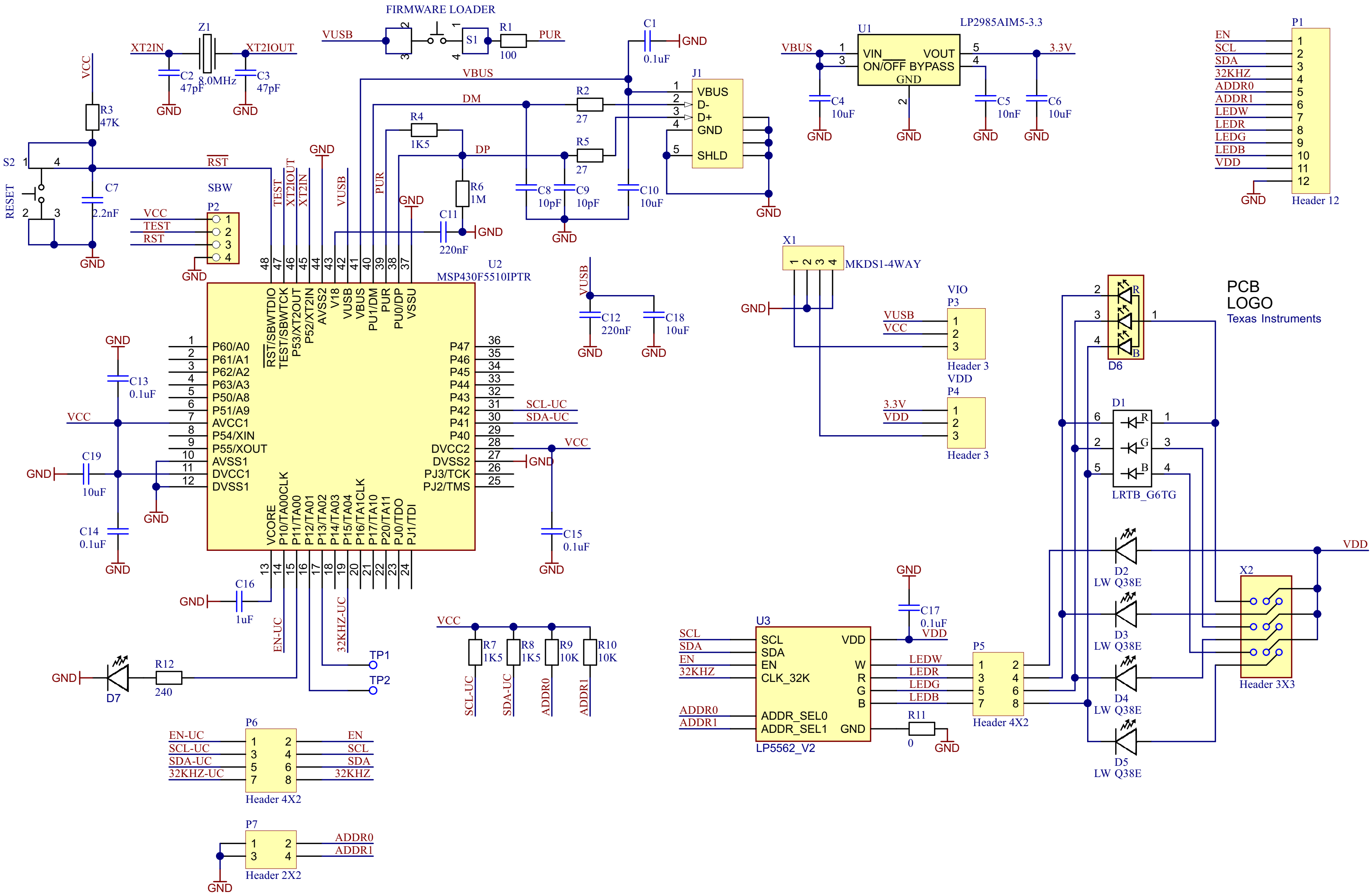


Figure A-1. Evaluation Board Schematics

Table A-1. Evaluation Board Bill of Materials

Component	Description	Designator	Footprint	Manufacturer	Manufacturer Type	Quantity
Ceramic capacitor	2.2nF 5% 50V C0G	C7	0603	Murata Electronics	GRM1885C1H222JA01D	1
Ceramic capacitor	47pF 5% 50V C0G	C2, C3	0603	Murata Electronics	GRM1885C1H470JA01D	2
Ceramic capacitor	220nF 10% 25V X7R	C11, C12	0603	Murata Electronics	GRM188R71E224KA88D	2
Ceramic capacitor	10pF 5% 50V C0G	C8, C9	0603	Murata Electronics	GRM1885C1H100JA01D	2
Ceramic capacitor	10nF 10% 50V X7R	C5	0603	Murata Electronics	GRM188R71H103KA01D	1
Ceramic capacitor	0.1µF 10% 25V X7R	C1, C13, C14, C15, C17	0603	Murata Electronics	GRM188R71E104KA01D	5
Ceramic capacitor	1µF 10% 16V X7R	C16	0603	Murata Electronics	GRM188R61E105KA12D	1
Ceramic capacitor	10µF 10% 16V X5R	C4, C6, C10, C18, C19	0805	Taiyo Yuden	EMK212BJ106KG-T	5
Diode LED	RGB LED	D1	LRTB_G6SF	Osram	LRTBG6SF-V2BA-3E7F-0-0-ZP	
Diode LED	White LED	D2, D3, D4, D5	LW M67C WLED - unipad	Osram	LW Q38E-Q1S2-3K6L-1	4
Diode LED	Red LED	D7	LSL296	Osram	LSL296-P2Q2-1-Z	1
Connector	USB 2.0, RA, SMT, AB Type, Receptacle, 5pos	J1	USB-mini-SMD	Wurth Elektronik	651305142821	1
Connector	Header, 12-Pin	P1	HDR1X12	FCI	68000-112HLF	1
Connector	Header, 4-Pin	P2	HDR1X4	TE connectivity	5-146292-4	1
Connector	Header, 3-Pin	P3, P4	HDR1X3	TE connectivity	5-146292-3	2
Connector	Header, 4-Pin, Dual row	P5, P6	HDR2X4	TE connectivity	5-146256-2	2
Connector	Header, 2-Pin, Dual row	P7	HDR2X2	TE connectivity	5-146256-1	1
Resistor	0R	R11	0603	Yageo	RC0603JR-070RL	1
Resistor	1K5 1%	R4, R7, R8	0603	Yageo	RC0603FR-071K5L	3
Resistor	1M 1%	R6	0603	Yageo	RC0603FR-071ML	1
Resistor	10K 1%	R9, R10	0603	Yageo	RC0603FR-0710KL	2
Resistor	27R 1%	R2, R5	0603	Yageo	RC0603FR-0727RL	2
Resistor	47K 1%	R3	0603	Yageo	RC0603FR-0747KL	1
Resistor	100R 1%	R1	0603	Yageo	RC0603FR-07100RL	1
Resistor	240R 1%	R12	0603	Yageo	RC0603FR-07240RL	1
Switch	Push Button	S1, S2	7914J	Bourns	7914J-1-000E	2
IC	Micropower 250 mA Low-Noise Ultra Low-Dropout Regulator	U1	MF05A	Texas Instruments	LP2985AIM5-3.3	1
IC	Microcontroller	U2	PQFP48	Texas Instruments	MSP430F5510IPTR	1
IC	4 CH LED Driver with Programmable Lighting Sequences	U3	DSBGA12	Texas Instruments	LP5562	1

Table A-1. Evaluation Board Bill of Materials (continued)

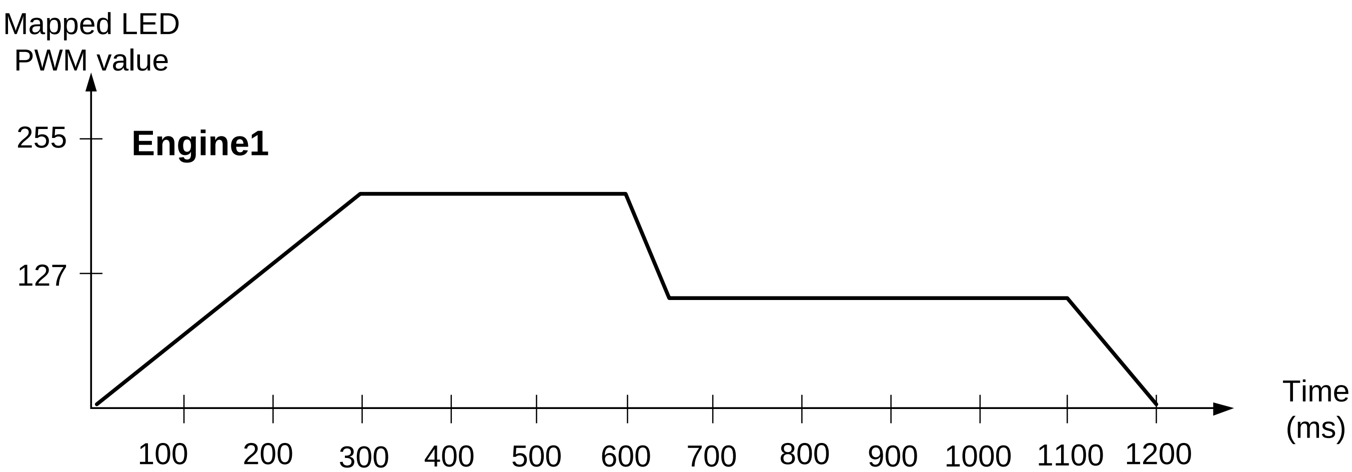
Component	Description	Designator	Footprint	Manufacturer	Manufacturer Type	Quantity
Connector	10A connector 3.81mm pitch 4way	X1	MKDS1-4	Phoenix	1727036	1
Connector	Header, 3-Pin, 3 row	X2	HDR3X3	Samtec	TSW-103-07-L-T	1
Crystal	Crystal SMD	Z1	HC49/4H_SMX	Abracon	ABLS-8.000MHZ-B4-T	1

Programming Considerations

B.1 Wait/Ramp Command

The ramp command generates either an increasing or decreasing PWM ramp, for which execution time and number of steps can be defined. In one ramp command PWM value can be incremented or decremented up to 128 steps from the present PWM value. The maximum PWM value is 255 which means the engine's mapped current source(s) is constantly active. Ramp command maximum execution time t_{MAX} = (1000 ms x number of steps) - 1 ms.

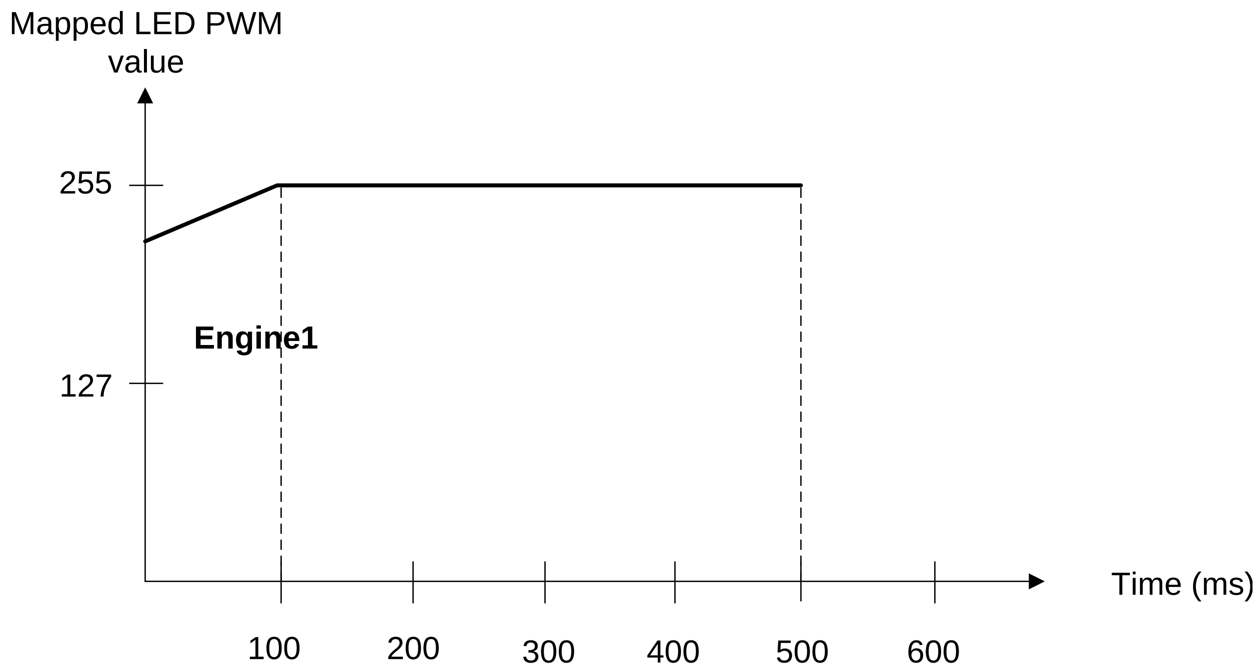
Figure B-1 illustrates ramp and wait command usage. The program has been made with the command compiler. The first ramp command increases PWM value from 0 to 100 in 150 ms. The second ramp command will increase PWM value from 100 to 200 in 150 ms. PWM value is kept constant during the next 300 ms wait cycle. Program then continues by ramping down PWM from 200 to 100 in 50 ms. During the 450 ms wait cycle PWM value is kept constant, and the last command decreases PWM value from 100 to 0 during 100 ms.



Engine1 program:
ramp 150, 100
ramp 150, 100
wait 300
ramp 50, -100
wait 450
ramp 100, -100

Figure B-1. Ramp and Wait Command Example

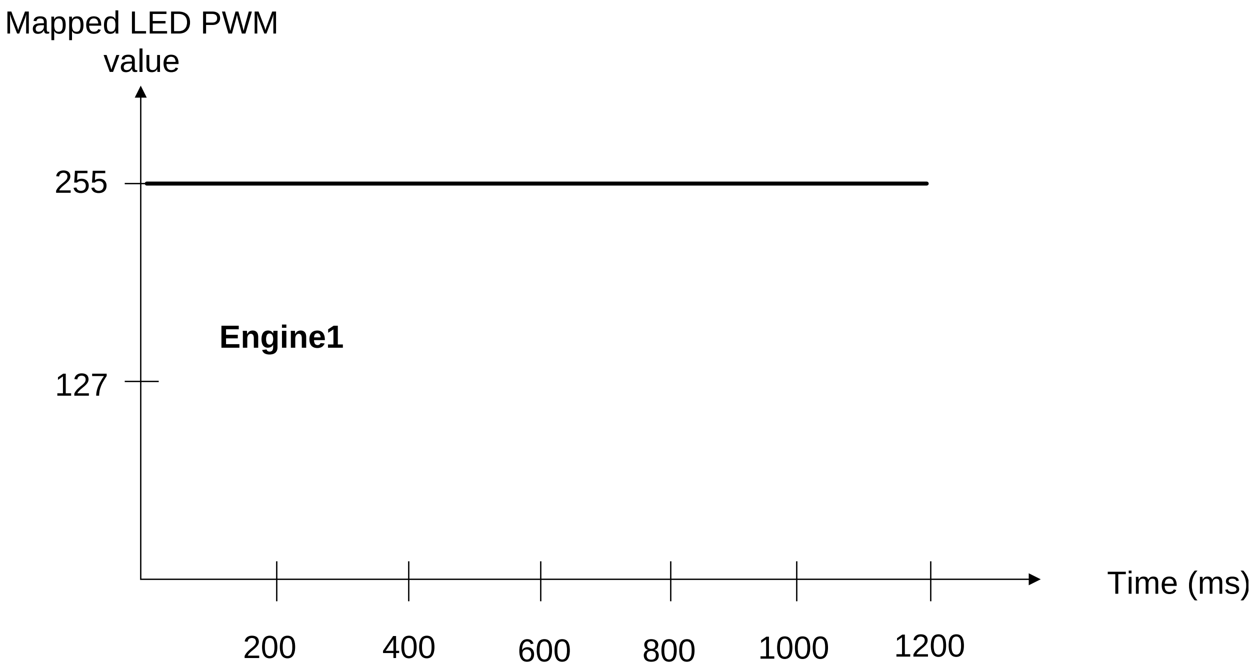
By combining the ramp and wait commands the number of program commands can be reduced. Figure B-2 illustrates this situation. Ramp begins from PWM value 231 and saturates to 255 in 100 ms. After PWM is saturated to maximum, ramp command changes to wait command for the rest 450 ms. Falling ramp saturates similar way to 0 PWM value. After the saturation, the rest of the command execution will be wait time.



Engine1 program:
set_pwm 231
ramp 500, 120

Figure B-2. Combined Ramp and Wait Command Example

Wait command maximum time is 999 ms on the command compiler; if longer wait time is required, then branching (looping) can be used. Branch command is explained in this application note. Also ramp command can produce the desired wait time if PWM level during wait cycle is either 0 or 255. Ramp command maximum execution time t_{MAX} = (1000 ms x number of steps) - 1 ms on the command compiler, so maximum wait time with ramp command is 127 999 ms. On Figure B-3, PWM is first set to 255, and 1200 ms wait cycle is produced with a ramp command that increases PWM level by 127 in 1200 ms. Since PWM is already at maximum, this command will only produce wait time. Similarly, falling ramp can be used as a wait command if PWM level is 0.



Engine1 program:
set_pwm 255
ramp 1200,127

Figure B-3. Using Ramp Command as Wait

B.2 Set PWM Command

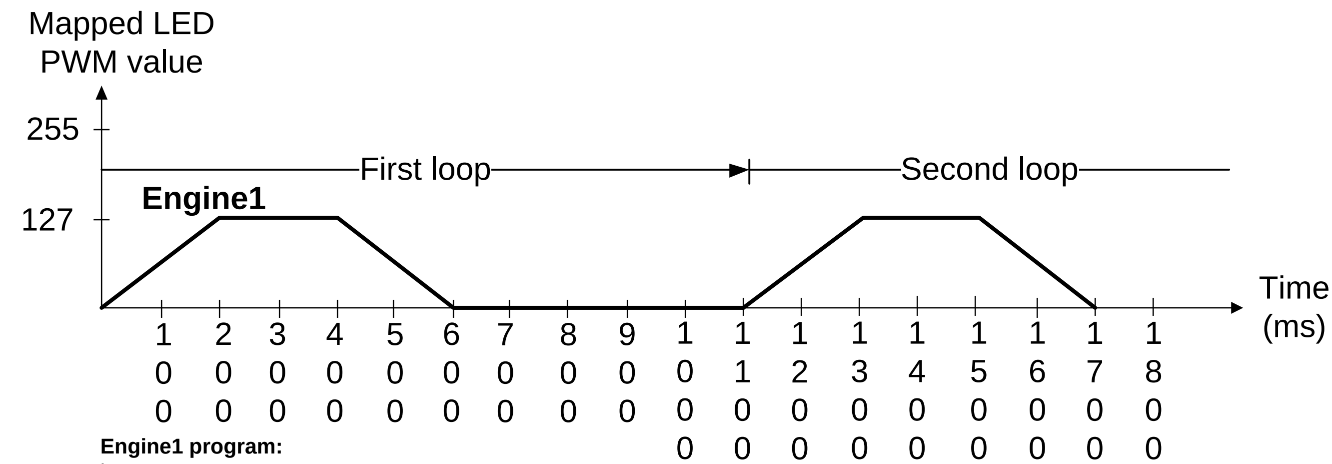
Set_pwm command adjusts the PWM level with 8-bit control from 0 to 255. PWM level is adjusted to new value in 0.488 ms (typ.).

B.3 Go-To-Start Command

Go-to-start command resets program counter, and program execution will be started from the beginning of the program. Go-to-start can be interpreted as an infinite loop. By default, all program memory locations are reset to zeros, which implies go-to-start command. In command compiler syntax this command is Start. If program memory is fully occupied, and last command is ramp, wait, set_pwm or trigger, program execution will be continued from the beginning of the program.

B.4 Branch Command

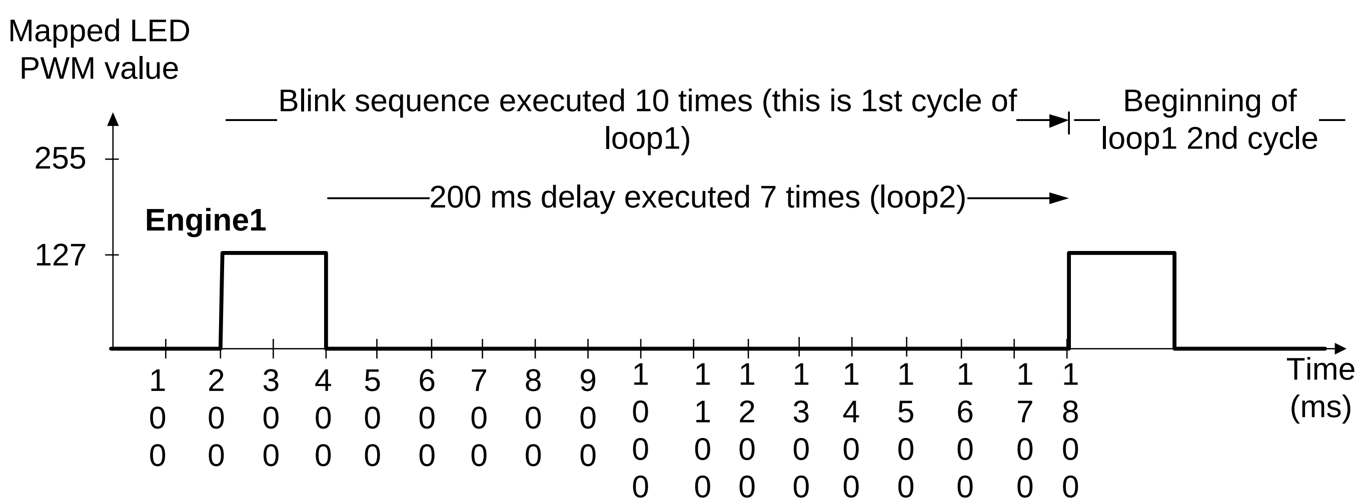
Branch command can be used to loop certain sequences in program. Figure B-4 illustrates branch command. Program ramps up PWM level from 0 to 127 in 200 ms. PWM level is kept at a constant 200 ms, then ramped down to 0 in 200 ms. PWM is kept at 0 during the next 500 ms. The whole sequence is executed 6 times. Loop start location is defined with label (loop1).



Engine1 program:
loop1:
ramp 200, 127
wait 200
ramp 200, -127
wait 500
branch 5, loop1

Figure B-4. Branch Command Example

The maximum loop count is 63 in one branch command, but the LP5562 supports loop inside loop i.e. nested looping. Nested looping is illustrated in the following example. 1600 ms blinking cycle is repeated 10 times. LED is active 200 ms during the cycle and rest of the cycle 1400 ms is wait time. The program has two loops loop1 and loop2. Loop1 repeats the whole sequence 10 times and loop2 creates 1400 ms wait time.



Engine1 program:
wait 200
loop1:
set_pwm 127
wait 200
set_pwm 0
loop2:
wait 200
branch 6, loop2
branch 9, loop1

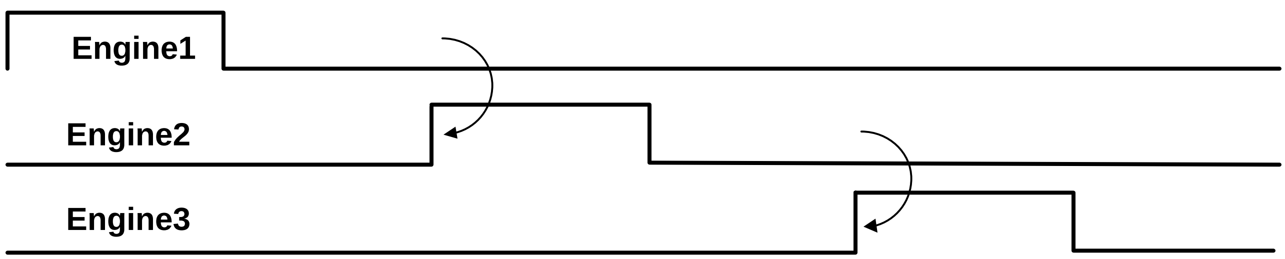
Figure B-5. Branch Command Example

B.5 End Command

End command stops program execution. There are two parameters which can be defined with an end command: interrupt and reset. Interrupt can be used to notify the processor that program execution is at the end. Status bits in register address 0CH inform which engine (1, 2, 3) has caused the interrupt. Status bits will be cleared when status register 0CH is read. Reset parameter resets program counter to 0, changes engine mode to hold from run mode, and sets PWM output to 0. If no parameters are defined, engine will be changed to hold mode and PWM value will remain.

B.6 Trigger Command

Triggering is efficient way of controlling program execution between LP5562 engines (1,2,3). The following example describes basic triggering concept. Each engine (1,2,3) have identical 100 ms LED pulse and 100 ms delay period after the pulse. Engine 1 begins the sequence by generating pulse and at the end sends trigger to engine 2. Engine 2 continues program execution, and at the end of the program sends trigger to engine 3 program.



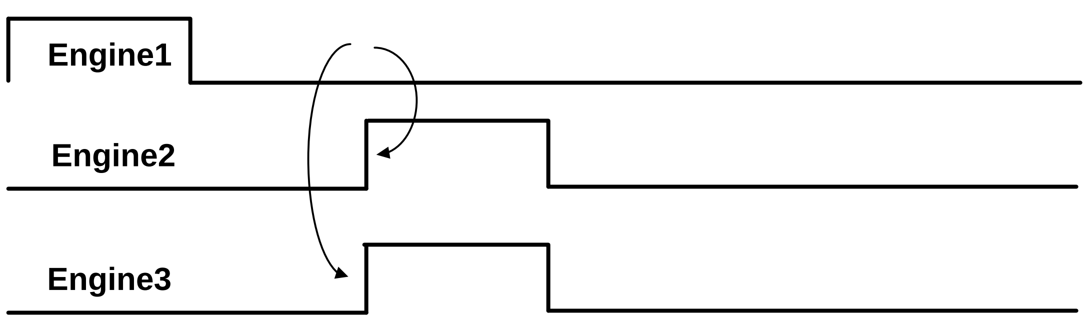
Engine1 program:
set_pwm 255
wait 100
set_pwm 0
wait 100
trigger s2
end

Engine2 program:
trigger w1
set_pwm 255
wait 100
set_pwm 0
wait 100
trigger s3
end

Engine3 program:
trigger w2
set_pwm 255
wait 100
set_pwm 0
end

Figure B-6. Basic Triggering Example

One engine can send multiple triggers in one command. Figure B-7 shows that engine 1 triggers both engine 2 and 3. Engine 2 and 3 programs are identical.



Engine1 program:
set_pwm 255
wait 100
set_pwm 0
wait 100
trigger s23
end

Engine2 program:
trigger w1
set_pwm 255
wait 100
set_pwm 0
end

Engine3 program:
trigger w1
set_pwm 255
wait 100
set_pwm 0
end

Figure B-7. Sending Multiple Triggers Example

Automatic Power-Save Functionality

C.1 Power Save With Engine Execution

Automatic power-save mode is enabled when PS_EN bit in register address 08H is 1. Almost all analog blocks (including internal oscillator) are powered down in power save, if external clock is used. However, if internal clock has been selected, only LED drivers are disabled during power save. Program execution engine remains active during power-save mode.

During program execution the LP5562 can enter power save if there is no PWM activity in LED outputs mapped to engines. To prevent short power-save sequences during program execution, the LP5562 has command look-ahead filter. In every instruction cycle engine commands are analyzed, and if there is sufficient time left with no PWM activity, device will enter power save. In power save program execution continues uninterruptedly. When a command that requires PWM activity is executed, fast internal startup sequence will be started automatically. Table C-1 describes commands and conditions that can activate power save. All engines need to meet power-save condition in order to enable power save. Note that if a LED output is mapped to the I^2C register control and has PWM, it prevents power save even though engine execution would allow power save.

Table C-1. Power-Save Condition With Operation Mode

LED controller operation mode (ENG1/ENG2/ENG3_MODE)	Power save condition
00b	Disabled mode enables power save
01b	Load program to SRAM prevents power save
10b	Run program mode enables power save if there is no PWM activity and command look ahead filter condition is met
11b	Direct control mode enables power save if there is no PWM activity

Table C-2. Power-Save Condition With Commands

Command	Power save condition
Wait	No PWM activity and current command wait time longer than 50 ms. If prescale = 1 then wait time needs to be longer than 80 ms. (see Figure C-1 and Figure C-4)
Ramp	Ramp command PWM value reaches minimum 0 and current command execution time left more than 50 ms. If prescale = 1 then time left needs to be more than 80 ms (see Figure C-2)
Trigger	No PWM activity during wait for trigger command execution (see Figure C-3)
End	No PWM activity or Reset bit = 1 (see Figure C-3)
Set PWM	Enables power save if PWM set to 0 and next command generates at least 50 ms wait
Other commands	No effect to power save

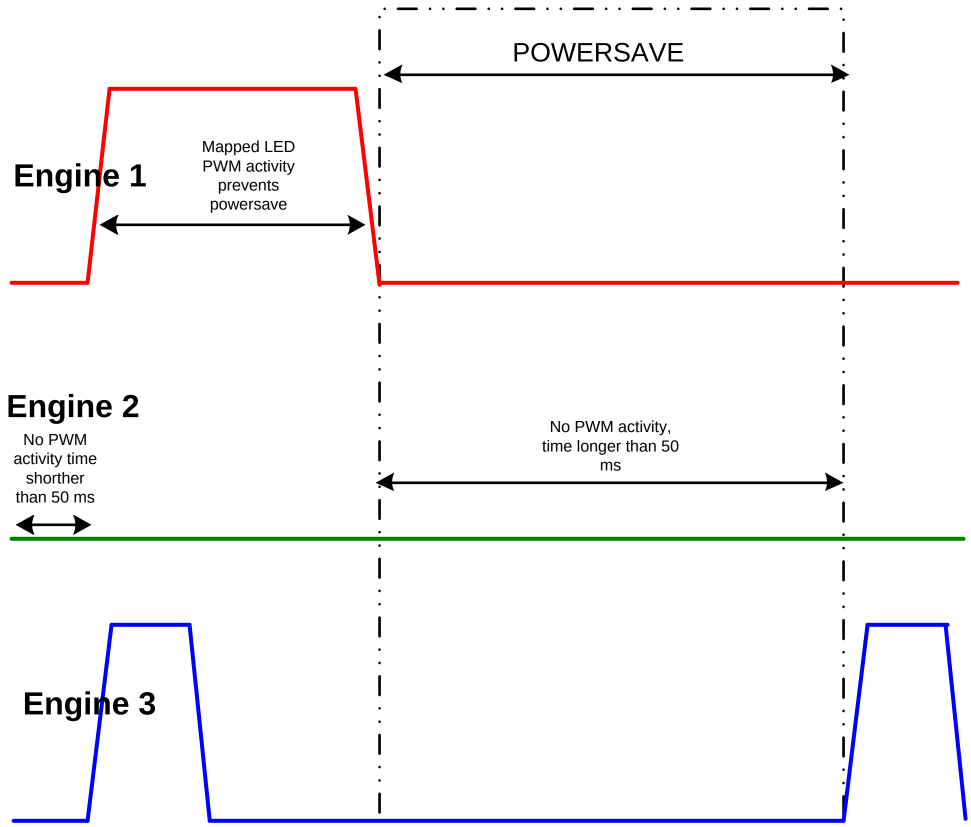


Figure C-1. Basic Power-Save Sequence With Long Inactive Period

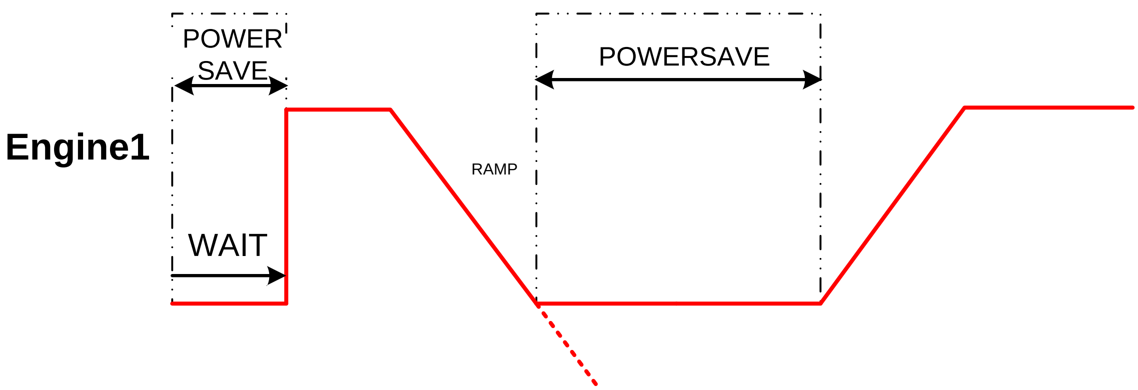


Figure C-2. Power-Save Sequence With a Ramp Command Used as Combined Ramp/Wait

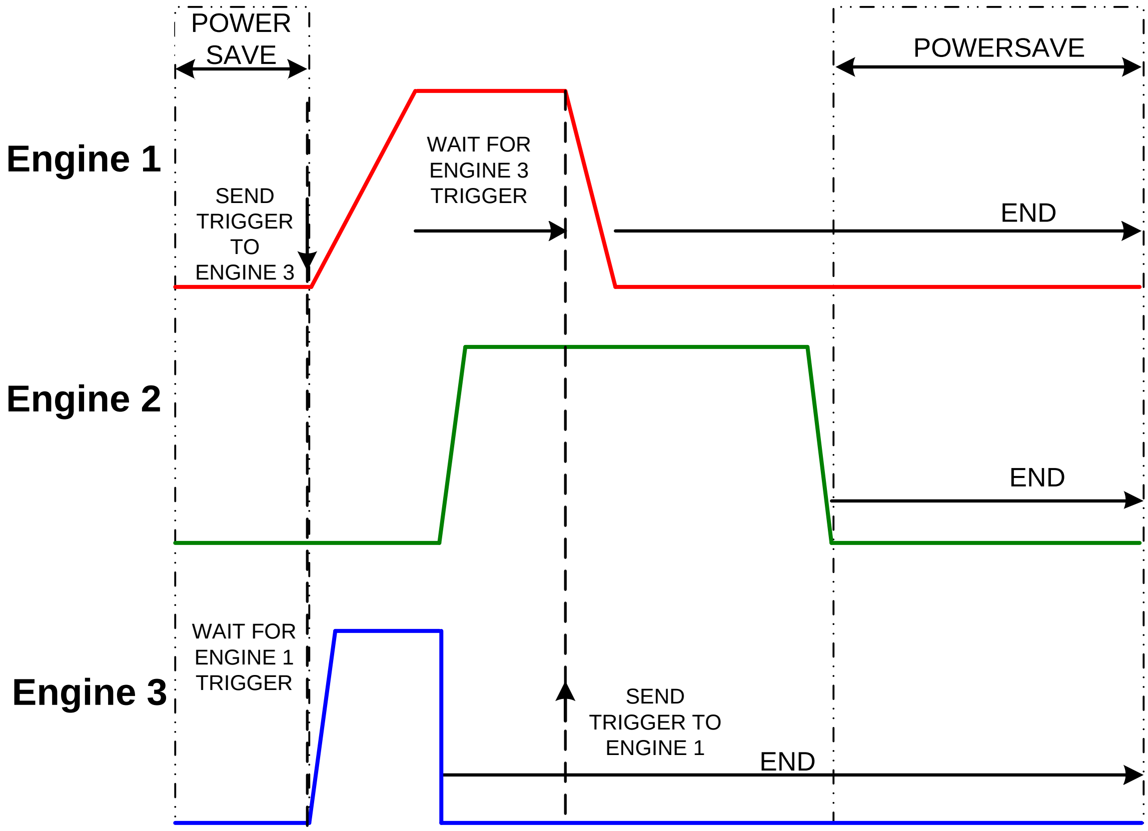


Figure C-3. Power-Save Sequence With Wait for Trigger Command

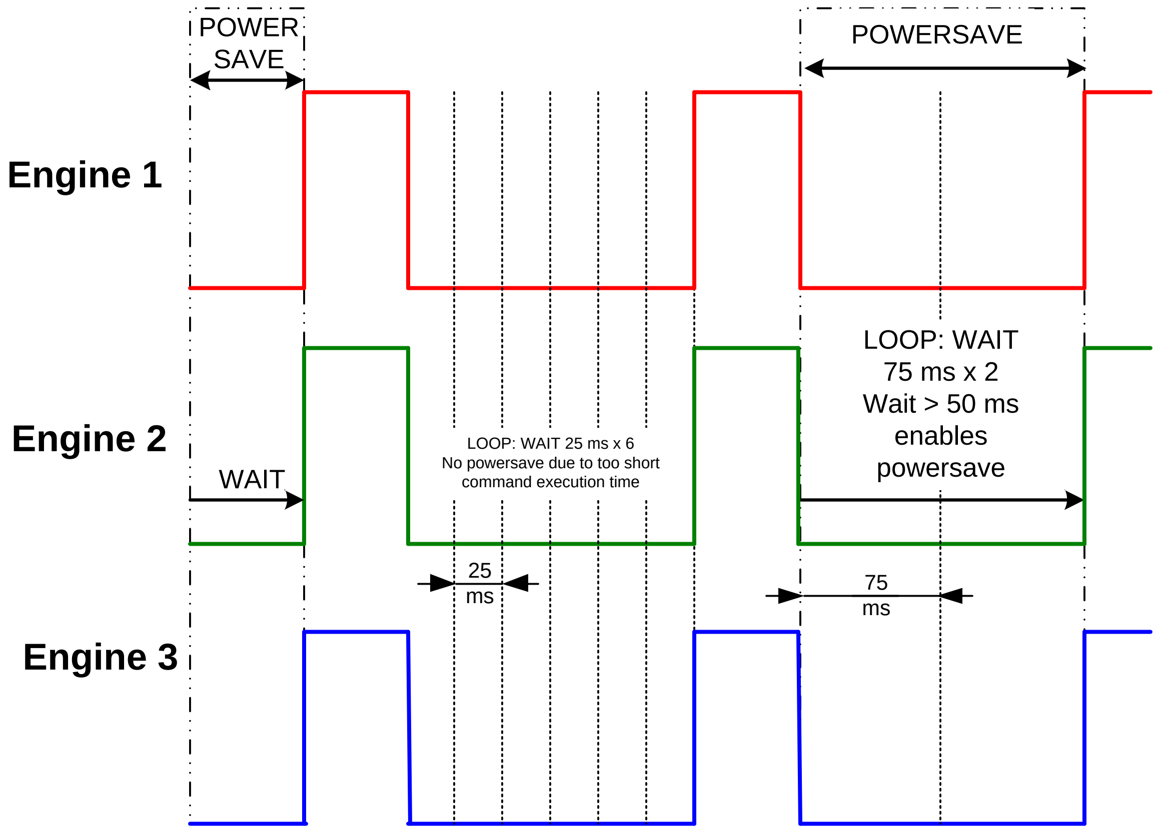


Figure C-4. Power-Save Sequence With Long and Short Wait Commands

USB PORT CONFIGURATION

D.1 Configuring USB Port Number

When the USB COM port number is bigger than 9, the evaluation program is not able to recognize the board. COM port number can be manually changed from Windows Device Manager. The below figures describe this sequence in Windows7. The Device Manager can be found from the Control Panel. Note that one may need to have Administrator rights to do the changes.

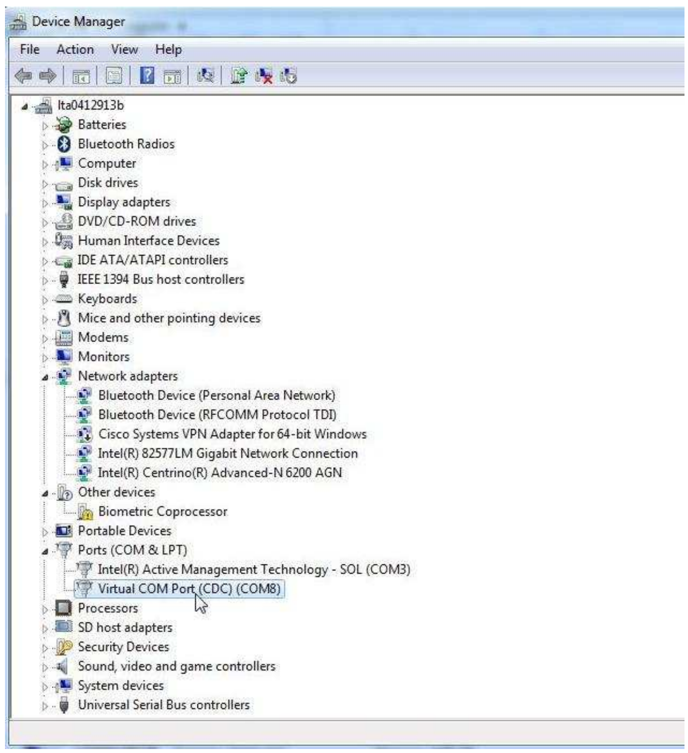


Figure D-1. Device Manager View. Select the Virtual COM Port

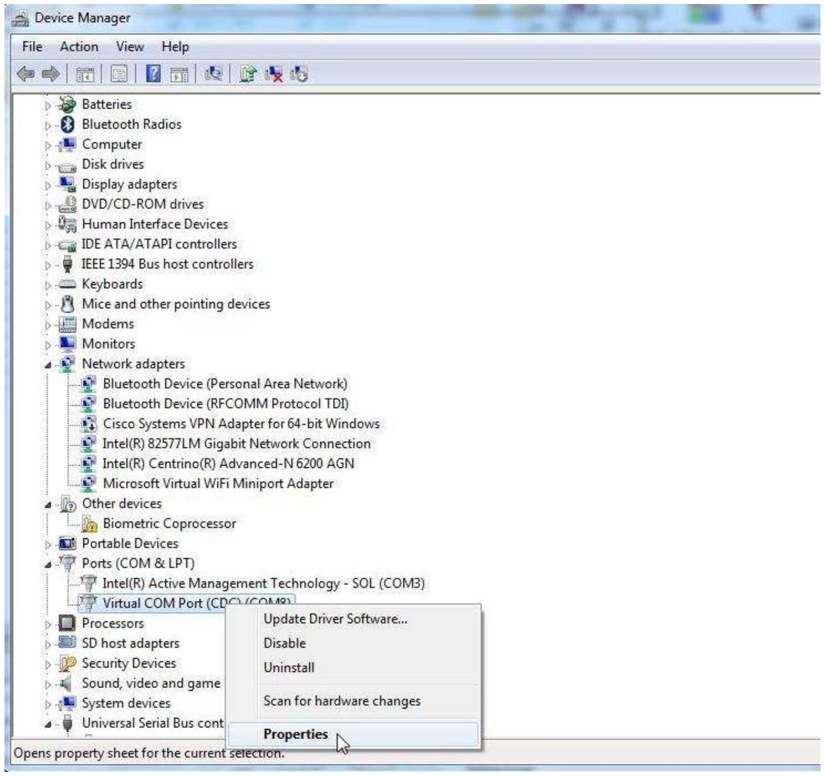


Figure D-2. Open Properties by Clicking Right Mouse Button on Virtual COM Port

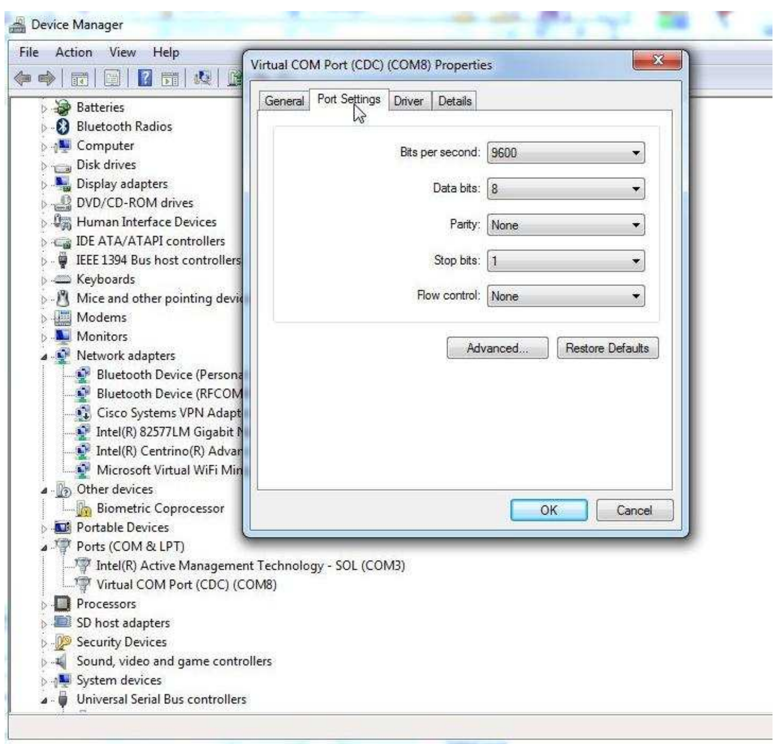


Figure D-3. Select Port Settings from The Virtual COM Port Properties

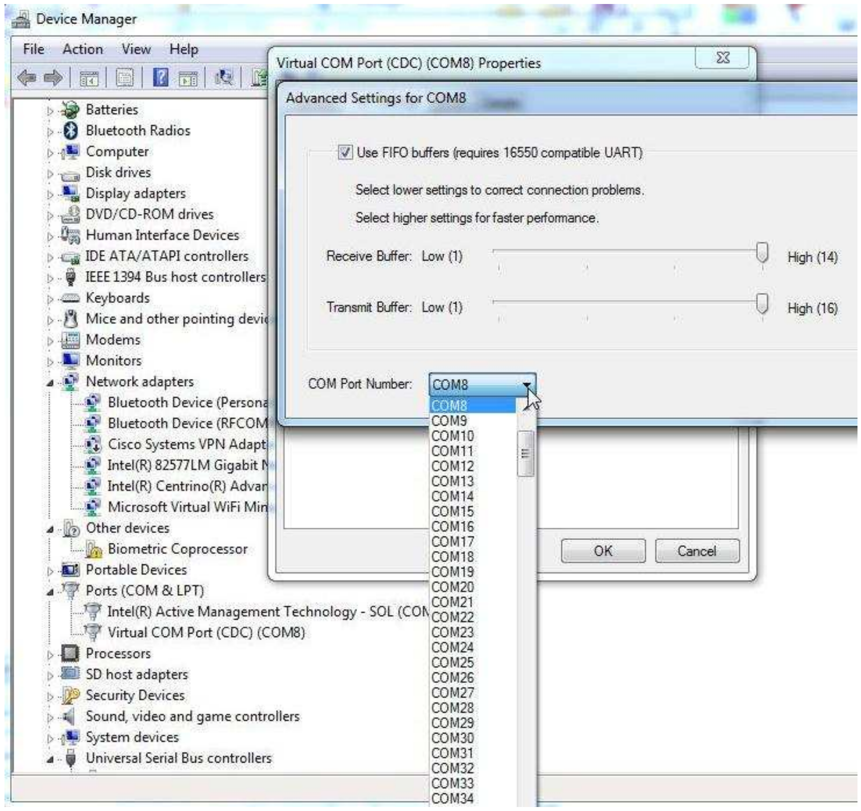


Figure D-4. Select Advanced from Virtual COM Port Properties and Select COM Port Number

Revision History

Changes from Original (April 2013) to A Revision	Page
• Changed schematic drawing	18

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

ADDITIONAL TERMS AND CONDITIONS, WARNINGS, RESTRICTIONS, AND DISCLAIMERS FOR EVALUATION MODULES

Texas Instruments Incorporated (TI) markets, sells, and loans all evaluation boards, kits, and/or modules (EVMs) pursuant to, and user expressly acknowledges, represents, and agrees, and takes sole responsibility and risk with respect to, the following:

1. User agrees and acknowledges that EVMs are intended to be handled and used for feasibility evaluation only in laboratory and/or development environments. Notwithstanding the foregoing, in certain instances, TI makes certain EVMs available to users that do not handle and use EVMs solely for feasibility evaluation only in laboratory and/or development environments, but may use EVMs in a hobbyist environment. All EVMs made available to hobbyist users are FCC certified, as applicable. Hobbyist users acknowledge, agree, and shall comply with all applicable terms, conditions, warnings, and restrictions in this document and are subject to the disclaimer and indemnity provisions included in this document.
2. Unless otherwise indicated, EVMs are not finished products and not intended for consumer use. EVMs are intended solely for use by technically qualified electronics experts who are familiar with the dangers and application risks associated with handling electrical mechanical components, systems, and subsystems.
3. User agrees that EVMs shall not be used as, or incorporated into, all or any part of a finished product.
4. User agrees and acknowledges that certain EVMs may not be designed or manufactured by TI.
5. User must read the user's guide and all other documentation accompanying EVMs, including without limitation any warning or restriction notices, prior to handling and/or using EVMs. Such notices contain important safety information related to, for example, temperatures and voltages. For additional information on TI's environmental and/or safety programs, please visit www.ti.com/esh or contact TI.
6. User assumes all responsibility, obligation, and any corresponding liability for proper and safe handling and use of EVMs.
7. Should any EVM not meet the specifications indicated in the user’s guide or other documentation accompanying such EVM, the EVM may be returned to TI within 30 days from the date of delivery for a full refund. THE FOREGOING LIMITED WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY TI TO USER AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. TI SHALL NOT BE LIABLE TO USER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RELATED TO THE HANDLING OR USE OF ANY EVM.
8. No license is granted under any patent right or other intellectual property right of TI covering or relating to any machine, process, or combination in which EVMs might be or are used. TI currently deals with a variety of customers, and therefore TI’s arrangement with the user is not exclusive. TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services with respect to the handling or use of EVMs.
9. User assumes sole responsibility to determine whether EVMs may be subject to any applicable federal, state, or local laws and regulatory requirements (including but not limited to U.S. Food and Drug Administration regulations, if applicable) related to its handling and use of EVMs and, if applicable, compliance in all respects with such laws and regulations.
10. User has sole responsibility to ensure the safety of any activities to be conducted by it and its employees, affiliates, contractors or designees, with respect to handling and using EVMs. Further, user is responsible to ensure that any interfaces (electronic and/or mechanical) between EVMs and any human body are designed with suitable isolation and means to safely limit accessible leakage currents to minimize the risk of electrical shock hazard.
11. User shall employ reasonable safeguards to ensure that user’s use of EVMs will not result in any property damage, injury or death, even if EVMs should fail to perform as described or expected.
12. User shall be solely responsible for proper disposal and recycling of EVMs consistent with all applicable federal, state, and local requirements.

Certain Instructions. User shall operate EVMs within TI’s recommended specifications and environmental considerations per the user’s guide, accompanying documentation, and any other applicable requirements. Exceeding the specified ratings (including but not limited to input and output voltage, current, power, and environmental ranges) for EVMs may cause property damage, personal injury or death. If there are questions concerning these ratings, user should contact a TI field representative prior to connecting interface electronics including input power and intended loads. Any loads applied outside of the specified output range may result in unintended and/or inaccurate operation and/or possible permanent damage to the EVM and/or interface electronics. Please consult the applicable EVM user's guide prior to connecting any load to the EVM output. If there is uncertainty as to the load specification, please contact a TI field representative. During normal operation, some circuit components may have case temperatures greater than 60°C as long as the input and output are maintained at a normal ambient operating temperature. These components include but are not limited to linear regulators, switching transistors, pass transistors, and current sense resistors which can be identified using EVMs’ schematics located in the applicable EVM user's guide. When placing measurement probes near EVMs during normal operation, please be aware that EVMs may become very warm. As with all electronic evaluation tools, only qualified personnel knowledgeable in electronic measurement and diagnostics normally found in development environments should use EVMs.

Agreement to Defend, Indemnify and Hold Harmless. User agrees to defend, indemnify, and hold TI, its directors, officers, employees, agents, representatives, affiliates, licensors and their representatives harmless from and against any and all claims, damages, losses, expenses, costs and liabilities (collectively, "Claims") arising out of, or in connection with, any handling and/or use of EVMs. User’s indemnity shall apply whether Claims arise under law of tort or contract or any other legal theory, and even if EVMs fail to perform as described or expected.

Safety-Critical or Life-Critical Applications. If user intends to use EVMs in evaluations of safety critical applications (such as life support), and a failure of a TI product considered for purchase by user for use in user’s product would reasonably be expected to cause severe personal injury or death such as devices which are classified as FDA Class III or similar classification, then user must specifically notify TI of such intent and enter into a separate Assurance and Indemnity Agreement.

RADIO FREQUENCY REGULATORY COMPLIANCE INFORMATION FOR EVALUATION MODULES

Texas Instruments Incorporated (TI) evaluation boards, kits, and/or modules (EVMs) and/or accompanying hardware that is marketed, sold, or loaned to users may or may not be subject to radio frequency regulations in specific countries.

General Statement for EVMs Not Including a Radio

For EVMs not including a radio and not subject to the U.S. Federal Communications Commission (FCC) or Industry Canada (IC) regulations, TI intends EVMs to be used only for engineering development, demonstration, or evaluation purposes. EVMs are not finished products typically fit for general consumer use. EVMs may nonetheless generate, use, or radiate radio frequency energy, but have not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC or the ICES-003 rules. Operation of such EVMs may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

General Statement for EVMs including a radio

User Power/Frequency Use Obligations: For EVMs including a radio, the radio included in such EVMs is intended for development and/or professional use only in legally allocated frequency and power limits. Any use of radio frequencies and/or power availability in such EVMs and their development application(s) must comply with local laws governing radio spectrum allocation and power limits for such EVMs. It is the user's sole responsibility to only operate this radio in legally acceptable frequency space and within legally mandated power limitations. Any exceptions to this are strictly prohibited and unauthorized by TI unless user has obtained appropriate experimental and/or development licenses from local regulatory authorities, which is the sole responsibility of the user, including its acceptable authorization.

U.S. Federal Communications Commission Compliance

For EVMs Annotated as FCC – FEDERAL COMMUNICATIONS COMMISSION Part 15 Compliant

Caution

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation. Changes or modifications could void the user's authority to operate the equipment.

FCC Interference Statement for Class A EVM devices

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at its own expense.

FCC Interference Statement for Class B EVM devices

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Industry Canada Compliance (English)

For EVMs Annotated as IC – INDUSTRY CANADA Compliant:

This Class A or B digital apparatus complies with Canadian ICES-003.

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Concerning EVMs Including Radio Transmitters

This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Concerning EVMs Including Detachable Antennas

Under Industry Canada regulations, this radio transmitter may only operate using an antenna of a type and maximum (or lesser) gain approved for the transmitter by Industry Canada. To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (e.i.r.p.) is not more than that necessary for successful communication.

This radio transmitter has been approved by Industry Canada to operate with the antenna types listed in the user guide with the maximum permissible gain and required antenna impedance for each antenna type indicated. Antenna types not included in this list, having a gain greater than the maximum gain indicated for that type, are strictly prohibited for use with this device.

Canada Industry Canada Compliance (French)

Cet appareil numérique de la classe A ou B est conforme à la norme NMB-003 du Canada

Les changements ou les modifications pas expressément approuvés par la partie responsable de la conformité ont pu vider l'autorité de l'utilisateur pour actionner l'équipement.

Concernant les EVMs avec appareils radio

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes : (1) l'appareil ne doit pas produire de brouillage, et (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Concernant les EVMs avec antennes détachables

Conformément à la réglementation d'Industrie Canada, le présent émetteur radio peut fonctionner avec une antenne d'un type et d'un gain maximal (ou inférieur) approuvé pour l'émetteur par Industrie Canada. Dans le but de réduire les risques de brouillage radioélectrique à l'intention des autres utilisateurs, il faut choisir le type d'antenne et son gain de sorte que la puissance isotrope rayonnée équivalente (p.i.r.e.) ne dépasse pas l'intensité nécessaire à l'établissement d'une communication satisfaisante.

Le présent émetteur radio a été approuvé par Industrie Canada pour fonctionner avec les types d'antenne énumérés dans le manuel d'usage et ayant un gain admissible maximal et l'impédance requise pour chaque type d'antenne. Les types d'antenne non inclus dans cette liste, ou dont le gain est supérieur au gain maximal indiqué, sont strictement interdits pour l'exploitation de l'émetteur.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Important Notice for Users of EVMs Considered “Radio Frequency Products” in Japan

EVMs entering Japan are NOT certified by TI as conforming to Technical Regulations of Radio Law of Japan.

If user uses EVMs in Japan, user is required by Radio Law of Japan to follow the instructions below with respect to EVMs:

1. Use EVMs in a shielded room or any other test facility as defined in the notification #173 issued by Ministry of Internal Affairs and Communications on March 28, 2006, based on Sub-section 1.1 of Article 6 of the Ministry's Rule for Enforcement of Radio Law of Japan,
2. Use EVMs only after user obtains the license of Test Radio Station as provided in Radio Law of Japan with respect to EVMs, or
3. Use of EVMs only after user obtains the Technical Regulations Conformity Certification as provided in Radio Law of Japan with respect to EVMs. Also, do not transfer EVMs, unless user gives the same notice above to the transferee. Please note that if user does not follow the instructions above, user will be subject to penalties of Radio Law of Japan.

http://www.tij.co.jp

【無線電波を送信する製品の開発キットをお使いになる際の注意事項】 本開発キットは技術基準適合証明を受けておりません。 本製品のご使用に際しては、電波法遵守のため、以下のいずれかの措置を取っていただく必要がありますのでご注意ください。

1. 電波法施行規則第6条第1項第1号に基づく平成18年3月28日総務省告示第173号で定められた電波暗室等の試験設備でご使用いただく。
2. 実験局の免許を取得後ご使用いただく。
3. 技術基準適合証明を取得後ご使用いただく。。

なお、本製品は、上記の「ご使用にあたっての注意」を譲渡先、移転先に通知しない限り、譲渡、移転できないものとします

上記を遵守頂けない場合は、電波法の罰則が適用される可能性があることをご留意ください。

日本テキサス・ インスツルメンツ株式会社

東京都新宿区西新宿６丁目２４番１号

西新宿三井ビル

http://www.tij.co.jp

Texas Instruments Japan Limited

(address) 24-1, Nishi-Shinjuku 6 chome, Shinjuku-ku, Tokyo, Japan

IMPORTANT NOTICE

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265