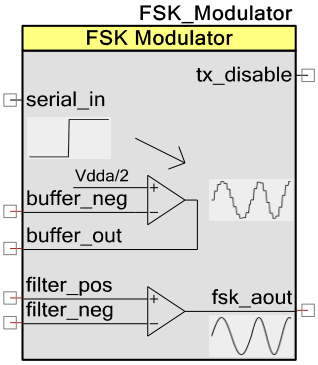
FSK Modulator (FSK_Modulator)

1.0

Features



- Modulates a frequency shift keying (FSK) signal at one of two carrier frequencies (110 kHz or 132 kHz)
- Selectable output wave shape (sine or square)
- Selectable transmit amplitude
- Compliant to FCC/CENELEC conducted emissions standards when used with the recommended coupling circuit
- Compatible with the FSK Demodulator and PLC Network Protocol components, as well as the PSoC® 1 powerline communication solution

General Description

The FSK Modulator component converts a baseband serial signal (for example, from a UART) into an FSK signal. An FSK signal represents each bit of information as one of two frequencies (for example, 133 kHz represents logic ‘0’ and 130 kHz represents logic ‘1’).

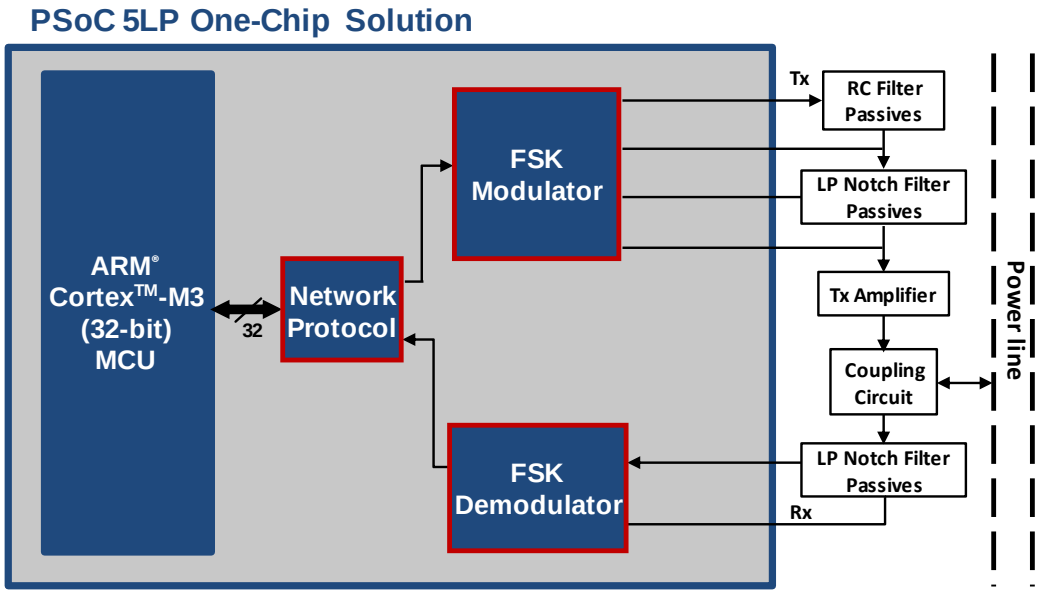
The FSK Modulator component also filters out the harmonics of the FSK signal, so that the transmitted signal is compliant to the FCC/CENELEC conducted emissions standards. The filter is a low-pass notch filter, which uses the PSoC 5LP’s integrated opamp and some external passive components, which are described below in the Schematic Macro Information section.

The FSK Modulator enables the external transmit amplification circuit to drive the FSK signal on the line. When not transmitting, it disables the transmit amplification circuit so that the circuit doesn’t load the line and attenuate the received signal.

When to Use a FSK Modulator

This component is most commonly used for powerline communication (PLC). Figure 1 shows a system diagram of powerline communication. The packet of data is created by the PLC Network Protocol component, which has a UART to convert each byte of data into a serial signal. The FSK Modulator then converts the serial signal into an FSK signal, which can be coupled to the powerline by a coupling circuit (described in application note AN76458 “PSoC 5LP High Voltage (120 – 240 VAC) Powerline Communication”). The signal is received on the other end by the receiver, which uses the FSK Demodulator component to convert the FSK signal back into a serial signal for the Network Protocol.

Figure 1. Powerline Communication System Diagram



Input/Output Connections

This section describes the various input and output connections for the FSK Modulator. An asterisk (*) in the list of I/Os states that the I/O may be hidden on the symbol under the conditions listed in the description of that I/O.

serial_in – Input

The serial_in terminal is the input for the serial baseband signal (for example, from a UART) that will be modulated.

tx_disable – Output

The tx_disable terminal is the output that drives the shutdown pin of the external transmit amplifier. When asserted, the amplifier will enter a high-impedance state so that it doesn't load the bus when the board is trying to receive data. This is needed for half-duplex systems (for example, powerline communication). The polarity (active high or active low) of can be configured.

buffer_neg – Analog *

The buffer_neg terminal is the negative input of the internal opamp that drives the FSK signal with a low impedance output to the transmit filter. It is present when the FSK Modulator is configured for a sine wave output.

buffer_out – Analog *

The buffer_out terminal is the output of the internal opamp that drives a low impedance output to the transmit filter. It is present when the FSK Modulator is configured for a sine wave output.

filter_pos – Analog *

The filter_pos terminal is the positive input of the internal opamp that filters the FSK signal. It is present when the FSK Modulator is configured for a sine wave output and the transmit filter is enabled.

filter_neg– Analog *

The filter_neg terminal is the negative input of the internal opamp that filters the FSK signal. It is present when the FSK Modulator is configured for a sine wave output and the transmit filter is enabled.

fsk_aout – Analog *

The fsk_aout terminal is the output of the internal opamp that filters the FSK signal. It is present when the FSK Modulator is configured for a sine wave output and the transmit filter is enabled.

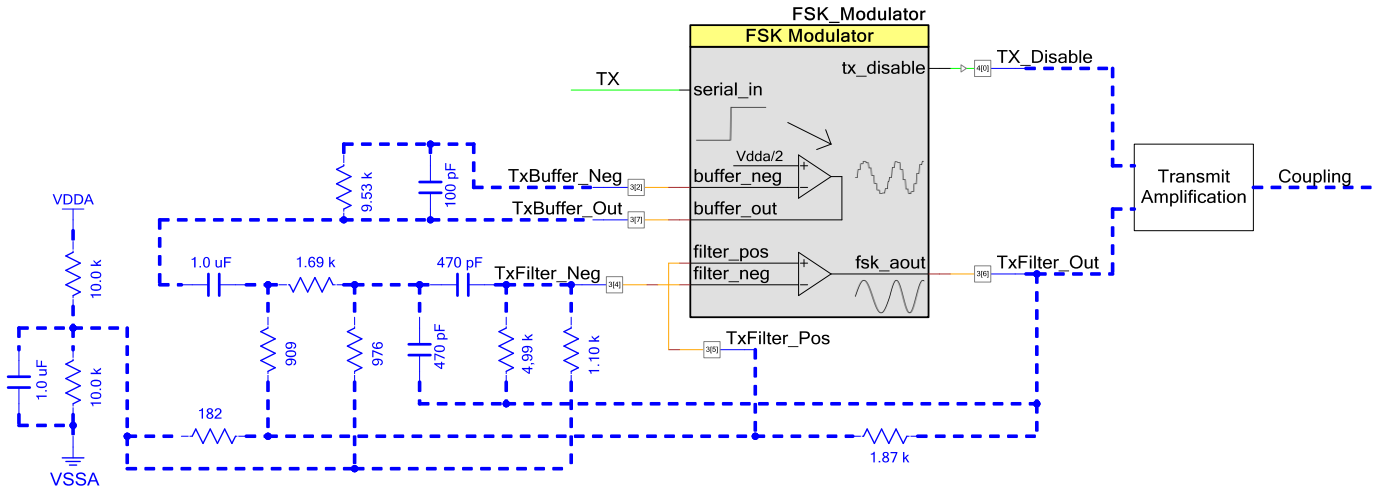
fsk_dout – Output *

The fsk_dout terminal is the square wave output of the FSK signal when the FSK Modulator is configured for a square wave output.

Schematic Macro Information

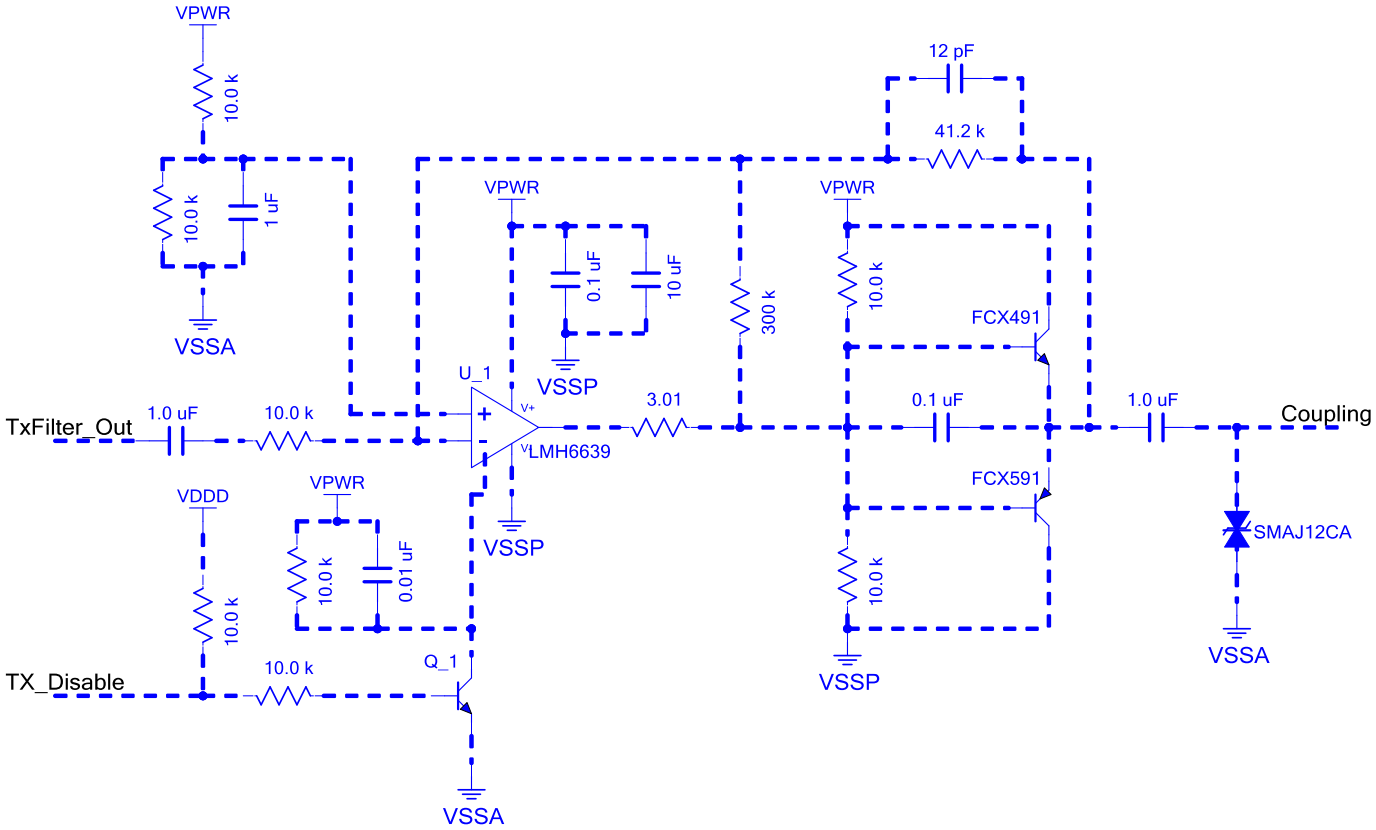
The default FSK Modulator in the Cy_Ref Catalog is a schematic macro using a FSK Modulator component with the default settings. It connects to analog pins for the opamp connections (including the FSK output) and a digital output pin for disabling the external transmit amplifier. It includes the recommended circuit for filtering the FSK signal so that it can pass the conducted emissions requirements of CENELEC EN50065-1 and FCC Part 15. This macro uses the Sine_Wave value for the FSK_Wave_Type parameter and the Enabled value for the Transmit_Filter parameter. If the FSK_Wave_Type parameter is Square_Wave, only the Transmit Amplification block is required for external circuitry. If the FSK_Wave_Type parameter is Sine_Wave and the Transmit_Filter parameter is Enabled, only the 9.53 k resistor and 100 pF capacitor between buffer_neg and buffer_out are required, along with the Transmit Amplification block.

Figure 2. FSK Modulator Macro



The transmit amplification circuit is shown in Figure 3. The application note AN76458 “PSoC 5LP High Voltage (120 – 240 VAC) Powerline Communication” provides a detailed description of the key components.

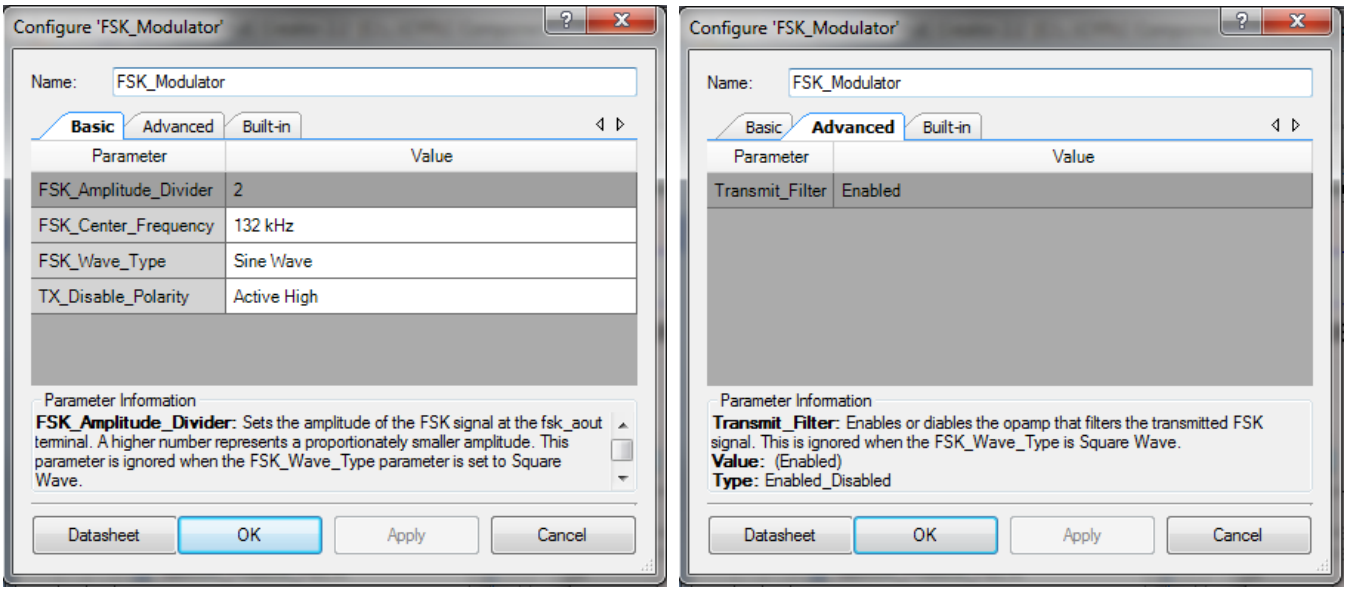
Figure 3. Transmit Amplification Schematics



Component Parameters

Drag a FSK Modulator onto your design and double click it to open the Configure dialog. Figure 4 shows the Configure dialog.

Figure 4. Configure FSK Modulator Dialog



The FSK Modulator provides the following parameters.

Basic Options

FSK_Amplitude_Divider

This parameter sets the amplitude of the FSK signal at the fsk_aout terminal. A higher number represents a proportionately smaller amplitude. This parameter is ignored when the FSK_Wave_Type parameter is set to Square Wave. The default value is 2. A minimum value of 4 should be used to pass the CENELEC EN50065-1 and FCC Part 15 conducted emissions tests.

FSK_Center_Frequency

This parameter sets the center frequency of the FSK signal. When set to 132 kHz, a logic ‘0’ input is modulated to 133.333 kHz and a logic ‘1’ input is modulated to 130.434 kHz. When set to 110 kHz, a logic ‘0’ input is modulated to 111.111 kHz and a logic ‘1’ input is modulated to 109.091 kHz. The default value is 132 kHz.

FSK_Wave_Type

This parameter sets the wave type of the FSK signal. When set to Sine Wave, the FSK signal is an analog signal that is sinusoidal and has a configurable amplitude, which is set by the Tx_Amplitude_Divider parameter. When set to Square Wave, the FSK signal is a digital square wave signal. The Sine Wave option is required to pass the CENELEC EN50065-1 and FCC Part 15 conducted emissions tests. The Square Wave option uses fewer PSoC resources and fewer external components. The default value is Sine Wave.

TX_Disable_Polarity

This parameter sets the polarity of the tx_disable terminal when the FSK Modulator shuts down the external transmit amplifier. A value of Active Low means that the FSK Modulator will drive a logic ‘0’ value when not transmitting and logic ‘1’ when transmitting. For half-duplex systems (for example, powerline communication), the FSK modulator shuts down the external transmit amplifier so that the amplifier doesn’t pull the line low when the board is trying to receive an FSK signal. The default value is Active Low.

Advanced Options

Transmit_Filter

This parameter enables or disables the opamp that filters the transmitted FSK signal. This is ignored when the FSK_Wave_Type is Square Wave. The default value is Enabled. It should be Enabled in order to pass the CENELEC EN50065-1 and FCC Part 15 conducted emissions tests.

Clock Selection

The master clock for the project must be set to 48 MHz to generate the correct FSK frequencies. In order to have accurate FSK frequencies that can be received by the FSK Demodulator , the PLL must be sourced by an external 24 MHz crystal with a maximum stability/tolerance of +/-50 ppm.

Resources

The FSK Modulator uses the following device resources. The number of resources depends on the value of the Wave_Type and Transmit_Filter parameters.

Wave_Type	Transmit_Filter	Digital Blocks				Analog Blocks		DMA Channels	Digital Clocks	Pins
		Datapath Cells	Macro cells	Status Cells	Control Cells	VIDAC Blocks	Opamps			
Sine Wave	Enabled	2	13	3	1	1	2	1	1	6
Sine Wave	Disabled	2	13	3	1	1	1	1	1	3
Square Wave	N/A	0	1	1	2	0	0	0	0	2

API Memory Usage

The component memory usage varies significantly, depending on the number of APIs used and component configuration. The following table provides the memory usage for all APIs available in the given component configuration.

The measurements were performed with the associated compiler configured in Release mode and with optimization set for Size. For a specific design, the map file generated by the compiler can be analyzed to determine the memory usage.

Wave_Type	API Memory (Bytes)	
	Flash	RAM
Sine Wave	1832	24
Square Wave	784	0

Application Programming Interface

Application Programming Interface (API) routines allow you to configure the component using software. The following table lists and describes the interface to each function. The subsequent sections cover each function in more detail.

By default, PSoC Creator assigns the instance name "FSK_Modulator_1" to the first instance of a component in a given design. If using the Network Protocol component, then you must name this component “FSK_Modulator”. If not using the Network Protocol component, you can rename it to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is "FSK_Modulator."

Functions

Function	Description
FSK_Modulator_Start()	Starts transmitting an FSK signal and de-asserts the tx_disable output

FSK_Modulator_Stop()	Stops transmitting the FSK signal and asserts the tx_disable output
FSK_Modulator_Init()	Initializes the default configuration provided by the customizer
FSK_Modulator_SetAmplitude()	Sets the amplitude of the FSK signal
FSK_Modulator_SetFskFrequency()	Sets the FSK center frequency
FSK_Modulator_EnableAmplifier()	De-asserts the tx_disable output
FSK_Modulator_DisableAmplifier()	Asserts the tx_disable output

Global Variables

Function	Description
FSK_Modulator_initVar (static)	The initVar variable indicates the initial configuration of this component. This variable is prepended with the component name. The variable is initialized to zero and set to 1 the first time FSK_Modulator_Start() is called. This allows for component initialization without reinitialization in all subsequent calls to the FSK_Modulator_Start() routine.

void FSK_Modulator_Start(void)

Description: This is the preferred method to begin component operation. This function turns on the modulator with the center frequency and amplitude from the settings provided during the configuration or the current values after FSK_Modulator_Stop() has been called. It also turns on the opamps for the filter and de-asserts the tx_disable output. On the first call of this function, it sets the FSK_Modulator_initVar variable and calls the FSK_Modulator_Init() function.

Parameters: None

Return Value: None

Side Effects: None

void FSK_Modulator_Stop(void)

Description: This function turns off the modulator, turns off the opamps for the filter and asserts the tx_disable output.

Parameters: None

Return Value: None

Side Effects: None

void FSK_Modulator_Init(void)

Description: Initializes or restores the component according to the customizer Configure dialog settings. It is not necessary to call FSK_Modulator_Init() because the FSK_Modulator _Start() API calls this function and is the preferred method to begin component operation.

Parameters: None

Return Value: None

Side Effects: None

void FSK_Modulator_SetAmplitude(uint8 txAmplitudeDivider)

Description: Sets the amplitude of the FSK signal based on the divider factor. This is only available when the FSK_Wave_Type parameter is set to Sine Wave.

Parameters: uint8 txAmplitudeDivider: the amount that the FSK signal's amplitude is divided by.

Return Value: None

Side Effects: None

void FSK_Modulator_SetFrequency(uint8 centerFrequency)

Description: Sets the center frequency of the FSK signal.

Parameters: uint8 centerFrequency: one of the following center frequencies of the FSK signal

Options	Description
FSK_Modulator_110K	The center frequency is 110 kHz. Logic '0' is modulated as 111.111 kHz and Logic '1' is modulated as 109.091 kHz
FSK_Modulator_132K	The center frequency is 132 kHz. Logic '0' is modulated as 133.333 kHz and Logic '1' is modulated as 130.434 kHz

Return Value: None

Side Effects: None

void FSK_Modulator_EnableAmplifier(void)

Description: This de-asserts the tx_disable output. It does not affect the output of the modulator or the filter opamps. It is not necessary to call this function for normal operation, since the FSK_Modulator_Start function de-asserts the tx_disable output.

Parameters: None

Return Value: None

Side Effects: None

void FSK_Modulator_DisableAmplifier(void)

Description: This asserts the tx_disable output. It does not affect the output of the modulator or the filter opamps. It is not necessary to call this function for normal operation, since the FSK_Modulator_Stop function asserts the tx_disable output.

Parameters: None

Return Value: None

Side Effects: None

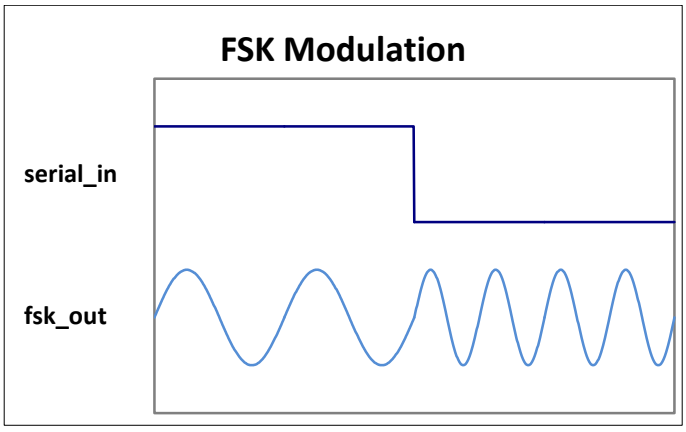
Sample Firmware Source Code

The code example in the application note AN76458 “PSoC 5LP High Voltage (120 – 240 VAC) Powerline Communication” contains the FSK Modulator, along with the FSK Demodulator and Network Protocol components. This forms a complete powerline communication (PLC) system and demonstrates how to read an analog voltage with an ADC, transmit that value over the powerline, then receive the value and display it on an LCD.

Functional Description

The FSK Modulator component converts a baseband serial signal (for example, from a UART) into a FSK signal. A FSK signal represents each bit of information as one of two frequencies. Figure 5 provides an example where logic ‘1’ is represented by a 100 kHz sine wave and logic ‘0’ is represented by a 200 kHz sine wave.

Figure 5. FSK Modulation Example



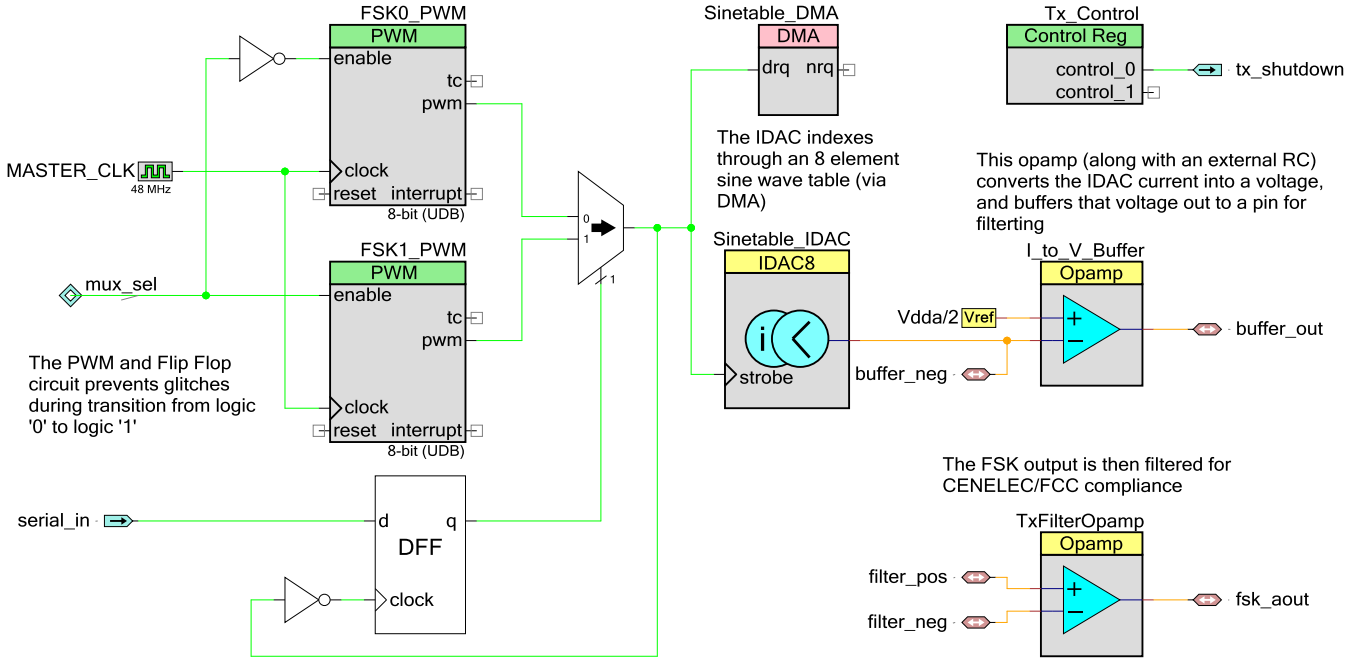
The architecture of the FSK Modulator is dependent on the value of the Wave_Type parameter (Sine Wave or Square Wave) and the Transmit_Filter (Enabled or Disabled) parameter.

Sine Wave Option

The Sine Wave option of the FSK Modulator consists of the following parts (PSoC Creator components in parentheses):

- Square wave modulator with glitch rejection (two PWMs, a multiplexer, and a flip-flop)
- Sine wave converter (IDAC)
- Current-to-voltage buffer (opamp)
- Filter (opamp)
- Transmit Amplifier Shutdown (control register)

Figure 6. FSK Modulator (Sine Wave) Internal Schematics



The transmitter performs the following steps to generate a sinusoidal FSK signal (Refer to Figure 10. FSK Modulation Timing Diagram):

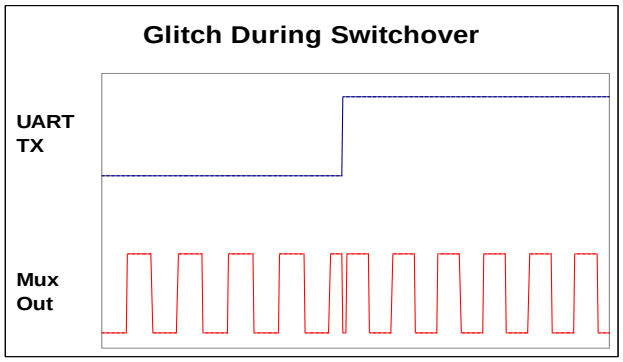
Square Wave Modulator with Glitch Rejection

The serial_in controls the switch input of a multiplexer. The multiplexer selects from the two square waves generated by the PWMs. The PWMs each generate a scaled up version of the two intended FSK frequencies. The IDAC generates eight samples for every period of the FSK signal, so the IDAC strobe (PWM frequency) needs to be eight times faster than the FSK

frequency. For example, to generate FSK frequencies of 130.434 kHz and 133.333 kHz, the PWM frequencies are 1043.478 kHz and 1066.666 kHz.

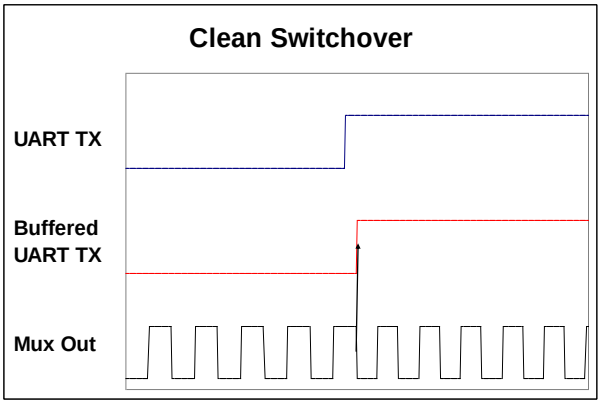
If the serial_in input transitions and the square wave has not completed its cycle, then the multiplexed wave may be cut short when it switches to the other frequency. This will generate a high frequency burst that the filters may not attenuate sufficiently and could cause compliance failures. Figure 7 shows an example when the multiplexer switches between 1000 kHz and 1100 kHz. There is a glitch pulse when it switches between these frequencies, which causes the generation of a higher frequency.

Figure 7. Runt Pulse During Switchover



To prevent the pulse, the transmitter only performs the transition when the multiplexed output has completed its cycle. Figure 8 shows the cleaner waveform. The multiplexer does not switch over to the other frequency until it has completed its cycle. The drawback of this implementation is that the delayed switchover introduces jitter due to the difference between the original UART output and the buffered UART output. However, in this case, the FSK frequency is significantly faster than the baud rate (45 - 55x) and combined with the oversampling of the IDAC (8x), a 1 cycle delay causes <0.2% of jitter, which is acceptable.

Figure 8. Clean Switchover



A flip-flop triggered by the negative edge of the output of the multiplexer implements the switchover circuit. The flip-flop then passes the UART's buffered output to switch the multiplexer. To make sure that each PWM starts from zero at every switchover, the UART's buffered output disables the PWM at the end of its cycle, just before switchover to the other PWM.

Sine Wave Converter

The multiplexer's output then connects to the strobe input of a current DAC (IDAC). The IDAC converts the square wave into a more sinusoidal wave. For each strobe, the IDAC reads the next element of a sine wave table via DMA. The strobe frequency is eight times the IDAC's sine wave output, and the sine table has eight elements.

The amplitude of the signal is set by the values in the sine wave table. The FSK_Modulator_SetAmplitude function changes these values.

The transmitter uses an IDAC instead of a VDAC, because the sample rate (strobe frequency) is higher than the maximum VDAC sample rate. The output resistance of the VDAC (4 kΩ) is high and would also require an opamp to buffer the output.

Current-Voltage Buffer

The IDAC output is not a pure sine wave and requires additional filtering in order to meet FCC Part 15 and CENELEC EN50065-1 conducted emissions requirments. First, the IDAC output is buffered to have a lower output impedance, which is required by the subsequent filter stage. This current-voltage buffer is implemented with an internal opamp and an external parallel RC filter (see Figure 9). The RC filter converts the current to a voltage, and it partially reduces the harmonics.

Filter

To reduce the harmonics further, the transmitter uses a low pass notch filter. This filter uses an internal opamp along with external resistors and capacitors (see Figure 9). When the

Transmit_Filter parameter is Disabled, this filter is bypassed, so that the FSK Modulator uses one less opamp.

Figure 9. FSK Modulator (Sine Wave) External Schematics

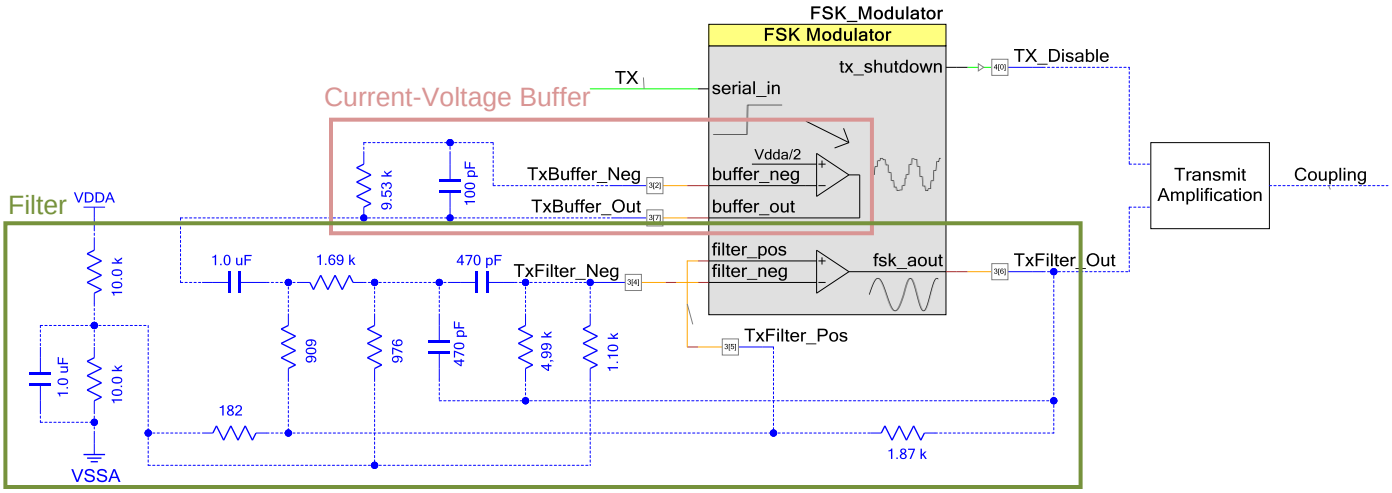
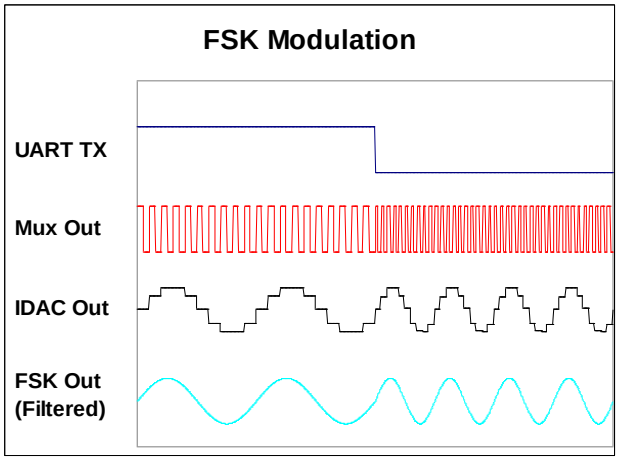


Figure 10 demonstrates this concept of FSK modulation and filtering. For illustrative purposes, the FSK sine wave frequencies are 100 kHz and 200 kHz and the IDAC is 10 times oversampled (10 element sine table). When the serial bit stream of the UART transmitter changes from logic ‘1’ to logic ‘0’, the square wave frequency changes from 1000 kHz to 2000 kHz. The IDAC then converts this square wave to a more sinusoidal 100 kHz or 200 kHz wave. The system then filters the signal to a 100 kHz or 200 kHz sine wave.

Figure 10. FSK Modulation Timing Diagram



Transmit Amplifier Shutdown

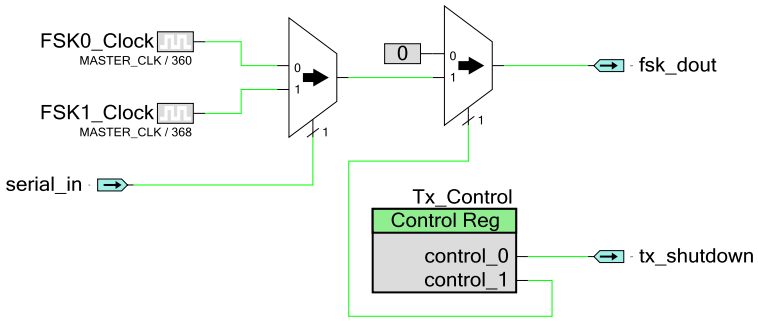
A 1-bit control register drives a digital signal that controls the tx_disable port. It is de-asserted when the the FSK Modulator is transmitting the FSK signal and is asserted when the FSK Modulator is not transmitting.

Square Wave Option

The Square Wave option of the FSK Modulator consists of the following parts (PSoC Creator components in parentheses):

- Square wave modulator (two clocks and a multiplexer)
- Transmit amplifier shutdown and FSK output enable (control register)

Figure 11. FSK Modulator (Square Wave) Internal Schematics



Square wave modulator

The serial_in signal controls the switch input of a multiplexer. The multiplexer selects from the two square waves generated by the digital clocks. The clock frequencies are the FSK frequencies.

Transmit amplifier shutdown and FSK output enable

A 2-bit control register performs the following functions:

- Bit 0 controls the tx_disable port. It is de-asserted when the the FSK Modulator is transmitting the FSK signal and is asserted when the FSK Modulator is not transmitting.
- Bit 1 controls a multiplexer that selects between a fixed output (logic '0') and the FSK modulated signal. This ensures that the there is no signal at the fsk_dout terminal when the FSK Modulator is turned off.

Performance

The performance of the FSK Modulator with the recommended coupling circuit is provided in the application note AN76458 "PSoC 5LP High Voltage (120 – 240 VAC) Powerline Communication".

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
1.0	First release.	First release.

© Cypress Semiconductor Corporation, 2013. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control, or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.