AN210253

CY294xx High-Performance Clock: Getting Started and Best Design Practices

Authors: Amitava Banerjee;Pawan Kumar

Associated Part Families: CY29411, CY29412, CY29421, CY29422, CY29430

Product Portfolio: High Performance Programmable Oscillator

To get the latest version of this application note, please visit http://www.cypress.com/go/AN210253.

This application note describes the architecture, operation, and performance of CY294xx clock devices which are suited for application in systems such as OTN, SONET/SDH, xDSL, GbE, and wireless infrastructure. It also includes best practices and design guidelines for configuring the devices with details on reference selection including crystal, internal memory structure, and hardware-software platforms.

Contents

1 Introduction

CY294xx devices are a family of one-PLL, high-performance clock generators mainly targeted at high-end networking systems such as OTN, SONET/SDH, xDSL, GbE, and wireless infrastructure. These devices are available in QFN and LCC packages. CY294xx devices generate one differential (or one single-ended) output up to 2.1 GHz with RMS jitter as low as 110 fs. They are factory- or field-programmable; the configuration stored in volatile memory can be controlled over the I²C interface. The Sigma-Delta PLL–based clock synthesis technology provides excellent supply noise rejection, simplifying the task of generating low-jitter clocks in noisy environments typically found in networking and communication systems.

Cypress recommends that you read this application note to understand device features, programmability options, internal configurations, and hardware design guidelines to meet system design requirements. The memory map and JEDEC description of the device given in this application note will help you configure on-the-fly changes of frequency through the I²C master controller. You should evaluate the software configuration in the evaluation kit (EVK) and check the performance using different test equipment such as an oscilloscope, a signal source analyzer. The design guidelines described in this document will help you design your application PCBs to achieve a quality clock in the system.

When starting a system design using CY294xx, see the following associated documents in addition to this application note:

- Datasheets of CY29411, CY29412, CY29421, CY29422, and CY29430 devices from the Cypress High Performance Programmable Oscillator webpage
- CY3676 EVK (Hardware Evaluation Kit of CY29412) design documents from the CY3676 Evaluation Kit web page
- CY3677 EVK (Hardware Evaluation Kit of CY29430) design documents from the CY3677 Evaluation Kit web page
- ClockWizard 2.1 software

CY294xx devices are available with field- and factory-programmable features. A brief product portfolio of the device family is given in Table 1.

Table 1. CY294xx Device Family Product Portfolio

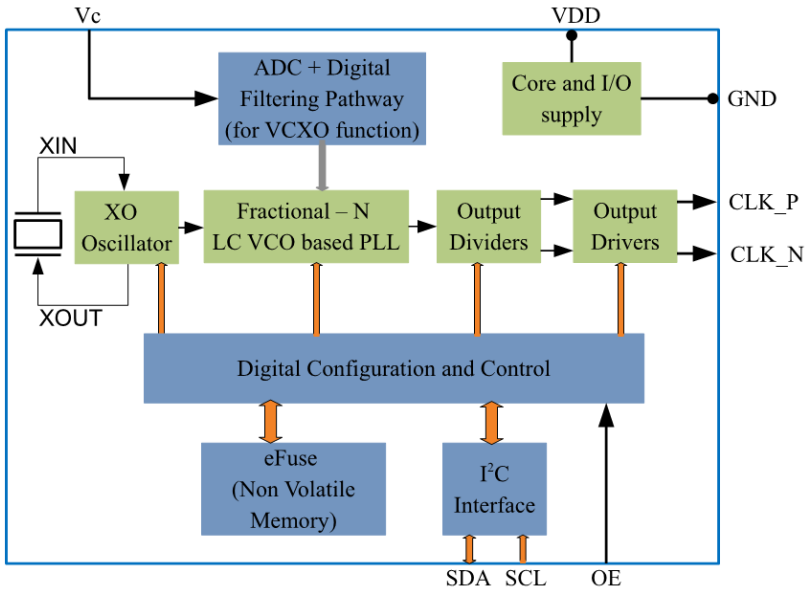
Device MPN	Package Type	VCXO Feature [1]	Input Type	Input Frequency Range	Output Type	Number of Frequency Profiles (FS)
CY29411	8-pin LCC (7 mm x 5 mm)	No	OT3 crystal inside	NA	One differential output	1
CY29412	8-pin LCC (7 mm x 5 mm)	Yes	OT3 crystal inside	NA	One differential output	1
CY29421	8-pin LCC (5 mm x 3.2 mm)	No	OT3 crystal inside	NA	One differential output	1
CY29422	8-pin LCC (5 mm x 3.2 mm)	Yes	OT3 crystal inside	NA	One differential output	1
CY29430	16-pin QFN	Yes	HFF crystal external	100 to 130 MHz	One differential and one single ended (only one works at a time based on programming)	4
			OT3 crystal external	100 to 130 MHz		
			LFF crystal external	50 to 60 MHz		
			TCXO input external	50 to 60 MHz		

Internal block diagrams of the CY29412/CY29422 and CY29430 devices are shown in Figure 1. The following are the key features of CY294xx devices:

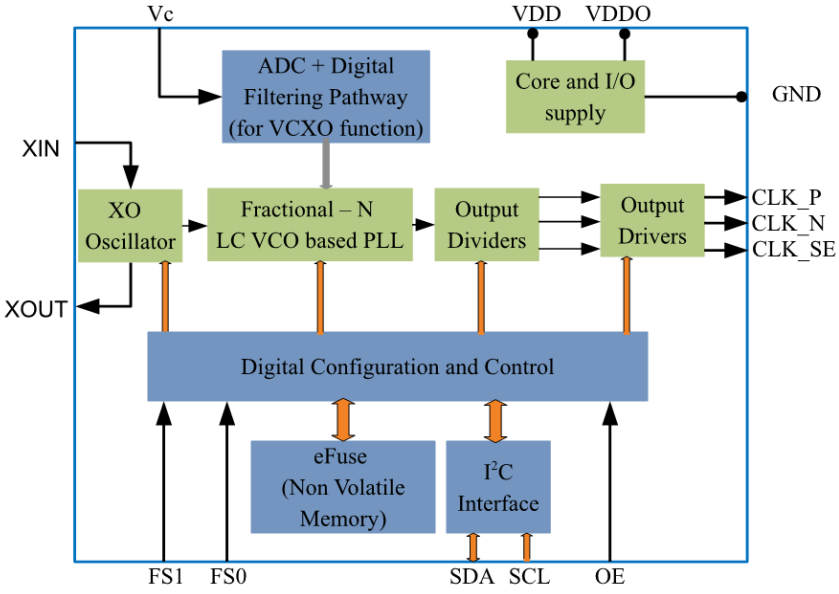
- Programmable fractional-N low-noise PLL with fully integrated VCO
- Programmable output frequency from 15 MHz to 2.1 GHz (differential), 15 MHz to 250 MHz (single-ended)
- Works on third overtone (OT3) of an integrated fixed-frequency crystal or high-frequency fundamental (HFF) mode crystal.
- LVPECL, LVPECL2, CML, HCSL, and LVDS programmable output formats
- Compatible with 3.3-V, 2.5-V, and 1.8-V supplies
- 110-fs typical integrated jitter performance (12 kHz to 20 MHz frequency offsets)
- VCXO functionality provided with tunable total pull range from ±50 ppm to ±275 ppm

[1] VCXO enabled part can be made non-VCXO through I^2C programming. However, a non-VCXO part cannot be made VCXO through I^2C programming. The details are explained in the section Large Change and Small Change Trigger.

Figure 1. Block Diagram of (a) CY29412/CY29422 and (b) CY29430



(a) CY29412, CY29422 architecture block diagram



(b) CY29430 architecture block diagram

Note: The blocks shown in green color in Figure 1 are configurable through ClockWizard 2.1 software.

2 Input and Output Settings and Programmable Features

Table 2 lists the programmable features of CY294xx devices.

Table 2. Programmable Features of CY294xx Devices

Programmable Feature	Description of the Feature
Input Reference	Input frequency
	Input reference type: OT3 / LFF / HFF / TCXO
Function	OE Polarity
Power Supply	V_{DD} (1.8, 2.5 or 3.3 V)
VCXO	Enable/Disable VCXO
	Kv Polarity
	Total Pull Range (TPR)
	Modulation Bandwidth
Output	Output frequency
	Output Standard (LVPECL, LVDS, HCSL, CML or LVCMOS)
Function	I2C address
	Four Frequency profile for CY29430, and one for CY2941x and CY2942x
Reference	Crystal (HFF, OT3, LFF) or TCXO input (This is for CY29430 device only)

2.1 Input Settings

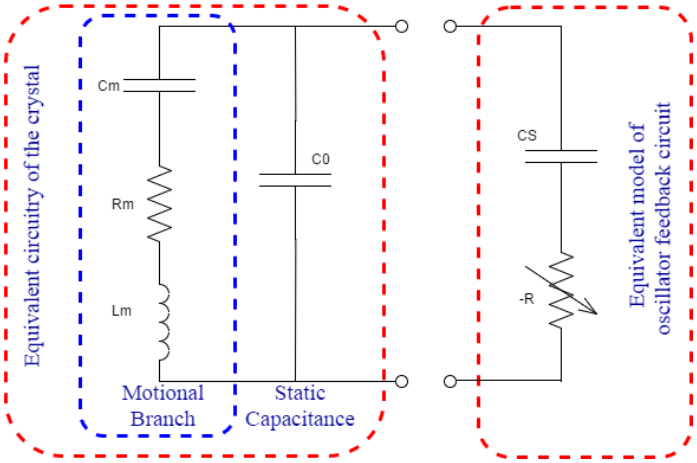
CY2941x and CY2942x devices have a crystal embedded inside the LCC packages.

The CY29430 device needs an external crystal (HFF crystal or OT3) or a TCXO input source on the PCB. Depending on the choice of crystal (HFF or OT3), the parameters of the internal oscillator circuitry change in the following way:

- With an HFF crystal, the oscillator circuit makes the crystal oscillate at the fundamental frequency by providing negative resistance at the fundamental frequency.
- With an OT3 crystal, the oscillator circuit suppresses oscillation at the fundamental frequency (by providing an effective positive resistance at the fundamental frequency) and makes the crystal oscillate at the third overtone by providing negative resistance at the third overtone frequency.

Figure 2 shows the equivalent RLC circuit of the crystal and the negative resistance of the oscillator. The internal oscillator circuit in the feedback path shows negative resistance (-R) at a particular frequency and starts the oscillation.

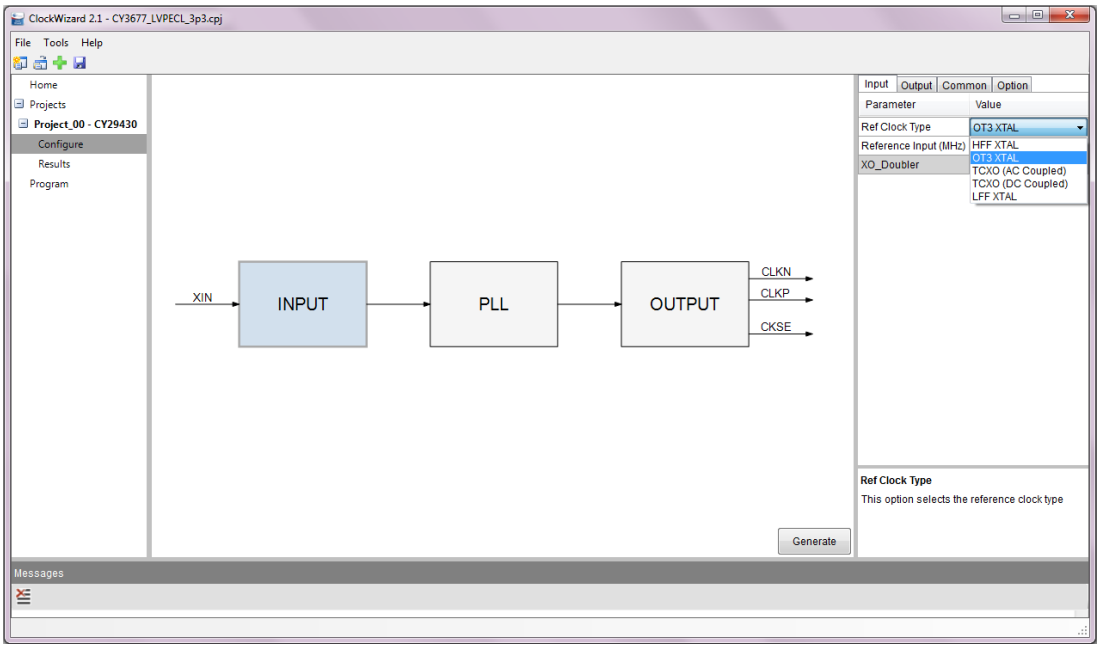
Figure 2. Block Diagram of Crystal Circuitry



For detailed crystal specifications, see the CY29430 datasheet. You can configure the crystal-related parameters with the ClockWizard 2.1 software.

Figure 3 shows the options to set configurable crystal parameters in the software.

Figure 3. CY29430 Input Crystal Settings in ClockWizard 2.1



If a TCXO or external clock is fed into the XIN input, a stable input must be present before the start of the V_{DD} ramp-up to the specified level. This is because the on-chip frequency calibration process starts at power ON and requires a stable reference input at the start of the calibration process. For detailed TCXO specifications, see the CY29430 datasheet.

2.2 Output Settings

CY2941x and CY2942x devices support the output differential standards LVPECL, LVPECL2, LVDS, CML, and HCSL. CY29430 has an additional LVCMOS output. LVPECL2 I/O standard is similar to the LVPECL standard with no common-mode output current. LVPECL2 output standard is the typical application of the LVPECL output driver when the V_{DD} is set to 1.8 V. However, LVPECL2 output standard is applicable for 2.5 V and 3.3 V supply as well. Figure 6 and Table 3 show the implementation difference and termination settings of the LVPECL and LVPECL2 output standard.

Any one of the differential or single-ended outputs in CY29430 can be enabled at a time. The I/O supplies can be 1.8 V, 2.5 V, or 3.3 V. You can program the power supply (V_{DD}), output frequency, and output standard of the device in the ClockWizard 2.1 software, as shown in Figure 4. The external termination settings of the outputs (on the PCB) should match the software profile of the device. The typical external (onboard) termination settings for the differential outputs are shown in Figure 5. The CMOS output does not need external termination. It is expected to drive capacitive load only.

LVDS I/O standard shown in Figure 5 is applicable for V_{DD} = 2.5 V, and 3.3 V only. For V_{DD} = 1.8 V, the output common mode voltage (V_{OCM}) is required to be supplied externally. Also, there should be AC coupling between the clock device (driver circuit) and the receiver circuit. The circuit representation is shown in Figure 7.

Figure 4. Termination Settings of Different Output Differential Standards in ClockWizard 2.1

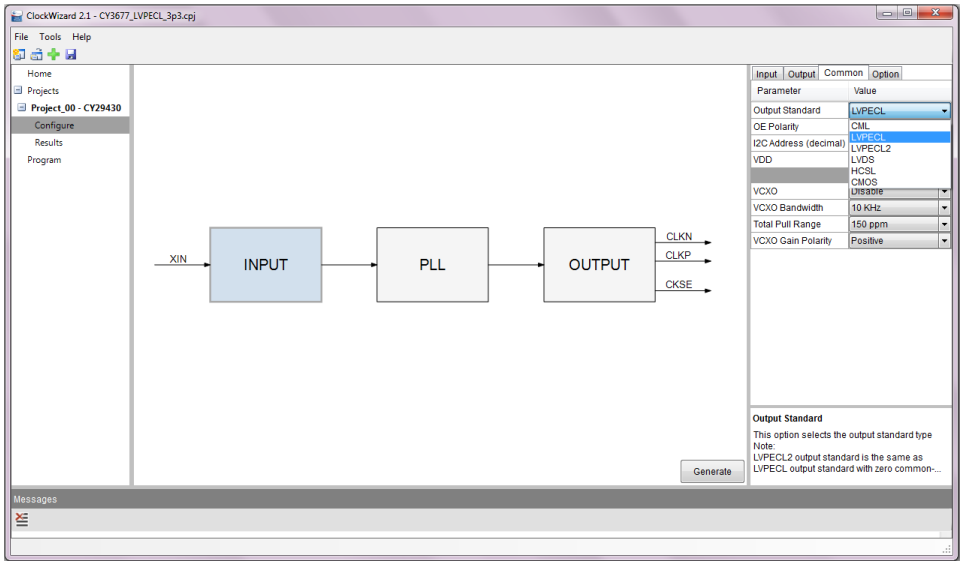
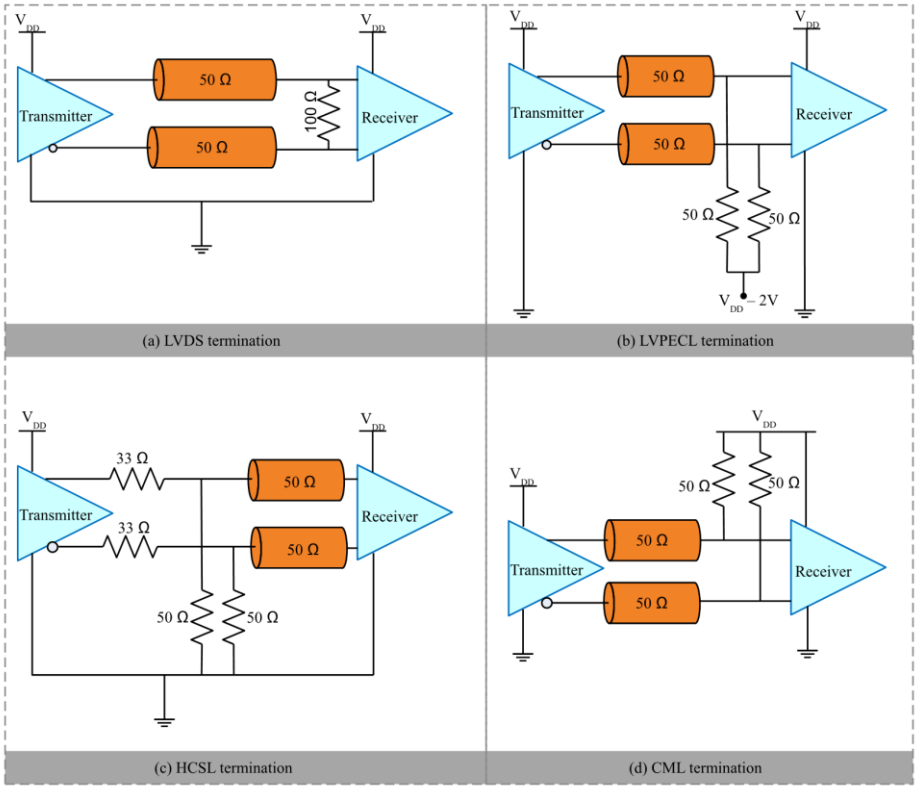


Figure 5. Termination Settings of Different I/O Standards External to Device



Note: The termination settings of the LVDS, and LVPECL I/O standards are applicable for V_{DD} = 2.5 V and 3.3 V. The termination settings of CML and HCSL I/O standards will work for all the V_{DD} ranges (1.8 V, 2.5 V, 3.3 V).

Figure 6. Internal Circuitry and the Termination Settings of LVPECL and LVPECL2 I/O Standard

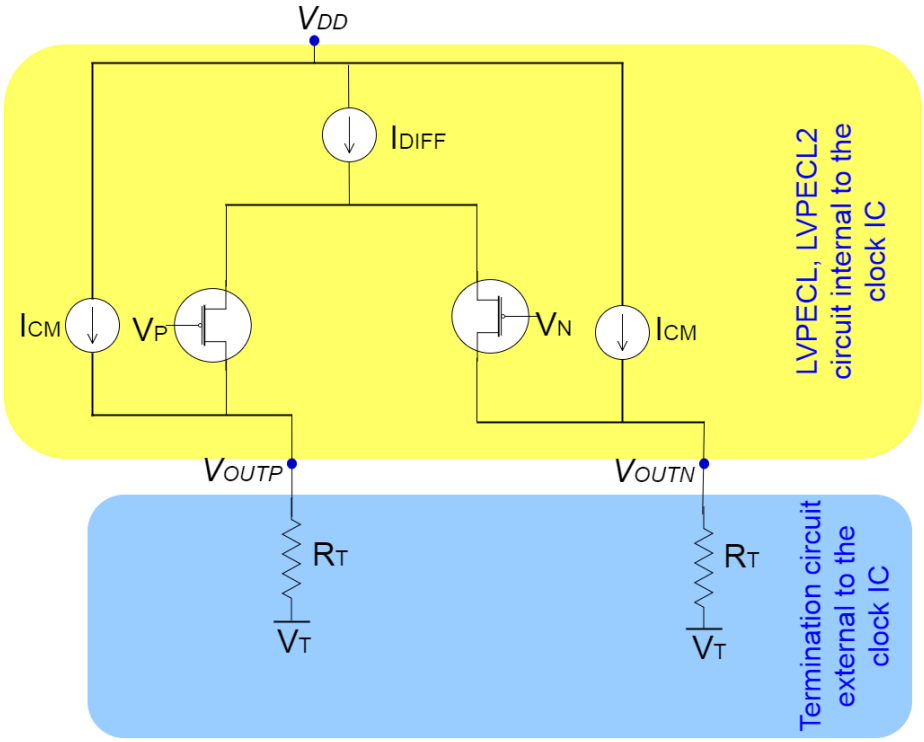
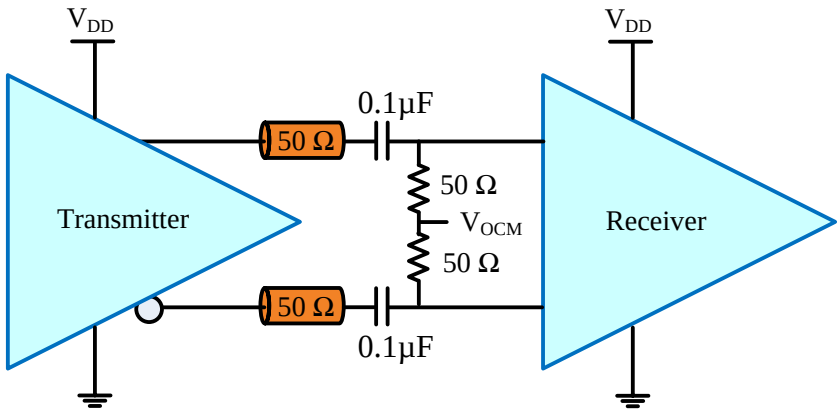


Table 3. Termination Settings of LVPECL and LVPECL2 I/O Standards

Sl. No.	Circuit parameters	LVPECL I/O standard	LVPECL2 I/O standard
1	V_{DD}	2.5 V, 3.3 V	1.8 V, 2.5 V, 3.3 V
2	I_{DIFF}	16 mA (typ)	16 mA (typ)
3	I_{DD} (total current consumption of the device)	93 mA (typ)	81 mA (typ)
		106 mA (max)	94 mA (max)
4	I_{CM}	6 mA	0
5	V_T	V_{DD} - 2 V	0 V
6	R_T	50 Ω	50 Ω

Note: V_P and V_N are the complementary logic signals controlling the ON/OFF operation of the LVPECL/LVPECL2 current sources. V_{OUTP} and V_{OUTN} signifies the complementary differential clock output pins.

Figure 7. Termination Settings of the LVDS I/O Standard for V_{DD} = 1.8 V



2.3 Other Configurable Parameters and Design Guidelines

In addition to the input reference and output settings, CY294xx devices offer the option to program other internal circuit parameters. You configure these features in ClockWizard 2.1 for field-programmable parts. Table 4 lists the configurable options of different control and power pins, which should be followed in hardware design also.

Table 4. Configurable Parameters of Control and Power Pins

IC pin	Functionality	Parameter Value	Guideline
V_{DD}	Configurable supply voltage	1.8 V / 2.5 V / 3.3 V	Use the onboard supply per the software configuration. Mismatch between these two may cause performance issues or reliability problems.
OE	Configurable polarity of OE	200 kΩ pulled down (if OE is programmed active LOW)	On the PCB, apply VDD to the OE pin to disable the output. Apply Gnd or leave the pin floating to enable the output.
		200 kΩ pulled up (if OE is programmed active HIGH)	Apply Gnd to the OE pin to disable the output. Apply VDD or leave the pin floating for output enabling.
V_C	Set external voltage for VCXO control [2]	Range of the external voltage should be 0.1 × V_{DD} to 0.9 × V_{DD}. VCXO feature cannot be enabled when PLL is configured in integer mode. VCXO functionality is provided with tunable total pull range from ±50 ppm to ±275 ppm.	For VCXO settings, add an LC filter circuit to ensure that a noise-free DC signal is applied at this input. An example of the filter circuit is given in Layout Guidelines. Do not apply any external voltage V_C = V_{DD}/2 when the device is configured as VCXO mode. Setting the V_C at V_{DD}/2 may cause spur at the output phase noise. Apply a voltage that is outside the range of V_{DD}/2 – 10 mV to V_{DD}/2 + 10 mV.

In addition to these pins, other internal parameters are programmable as follows:

- Choice of Kv polarity (for VCXO capable parts). If Kv is programmed negative, the output frequency ppm decreases with increasing Vc. For positive Kv, the output frequency ppm increases with increasing Vc.
- Choice of default frequency outputs (based on FS availability)
- Choice of total pull range (for VCXO capable parts)
- Choice of modulation bandwidth (for VCXO capable parts)

[2] VCXO enabled part can be made non-VCXO through I^2C programming. However, a non-VCXO part cannot be made VCXO through I^2C programming. The details are explained in the section Large Change and Small Change Trigger.

It is important to know the functionalities of the FS[1:0], SDA, and SCL pins for system design. Features and design guidelines are given in Table 5.

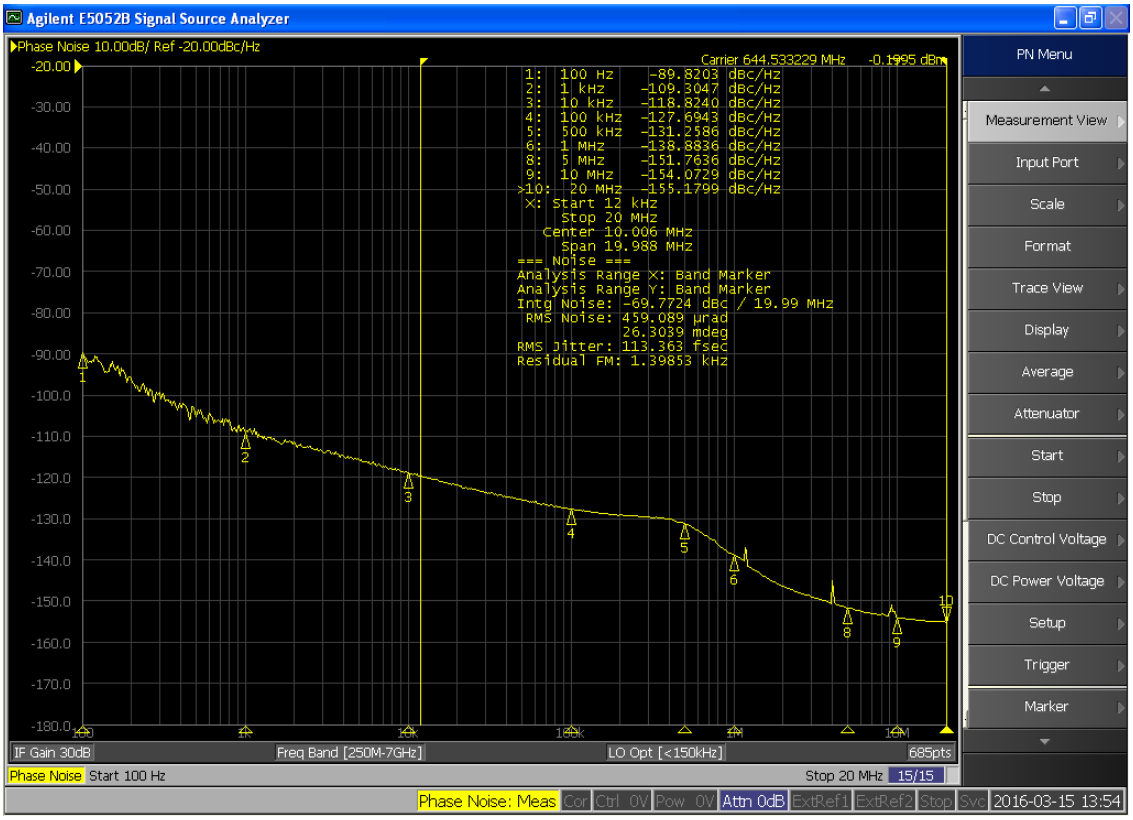
Table 5. Features of Frequency Select and I²C Pins

Pin	Internal Resistor	Guideline
FS[1:0] [3]	100 kΩ pull down	If no external signal is applied, default 00 profile will be selected. Apply logic '1' externally to change to other frequency profiles.
SDA	NA	No internal resistor. Connect external pull-up resistor close to the device pin. While PCB routing, it is recommended to have length matching between the SDA and SCL lines.
SCL	NA	No internal resistor. Connect external pull-up resistor close to the device pin. While PCB routing, it is recommended to have length matching between the SDA and SCL lines.

3 RMS Jitter Performance

The CY294xx device family is a high-performance, programmable PLL solution for crystal oscillators. It is mainly targeted at replacing complex SAW and inverted MESA oscillators with a cost-effective and more flexible solution. This part is designed to meet the frequency and jitter requirements of interface standards such as 10/40/100 GbE, SyncE, and IEEE 1588. The key specifications of the part are frequency support up to 2.1 GHz and a very low RMS phase jitter of 110 fs. The phase noise plots given in Figure 8 and Figure 9 show the excellent jitter performance of these devices. The profiles are created based on commonly used system-level applications of the CY294xx devices.

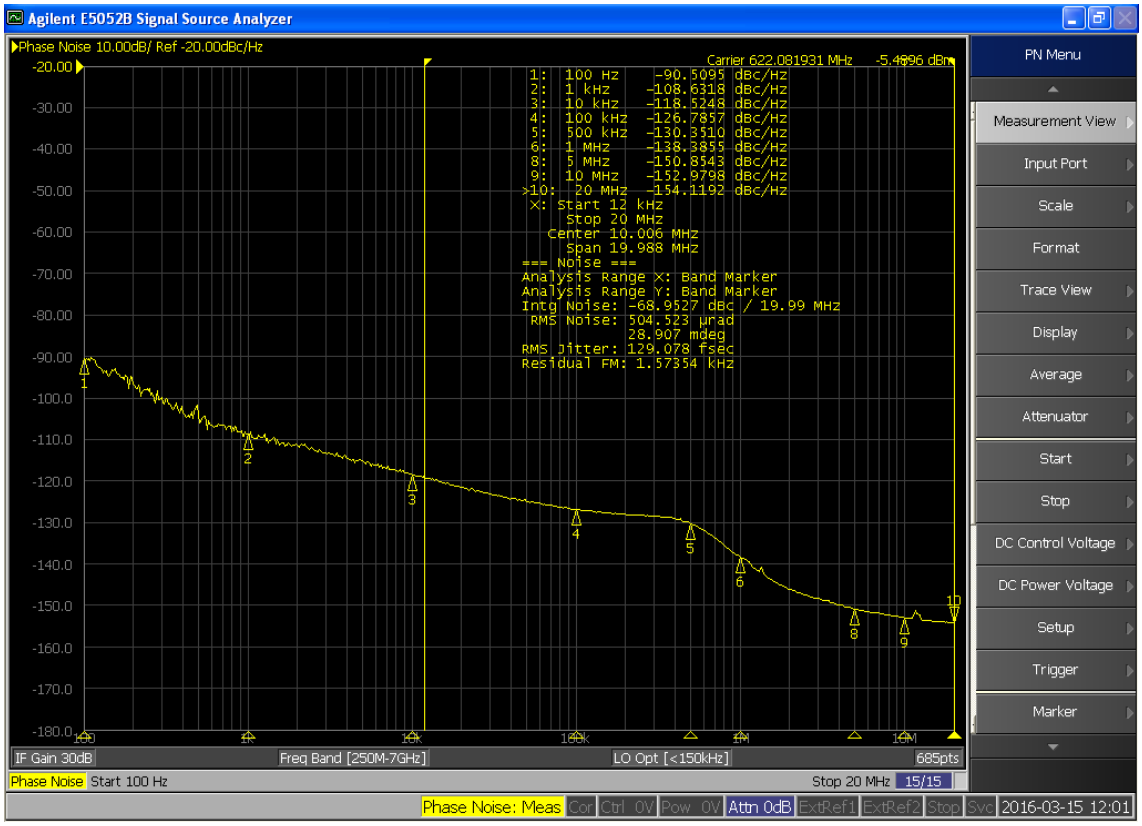
Figure 8. Phase Noise Plot of CY29430 (device programmed at V_{DD} = 3.3 V, output frequency 644.53 MHz, standard LVPECL, non-VCXO mode)



[3] This feature is only for CY29430. CY2941x and CY2942x can store only one profile and do not have FS pins. It is recommended that you refer to the schematic and layout files of CY3676 and CY3677 EVKs for designing the I²C interface on the PCB.

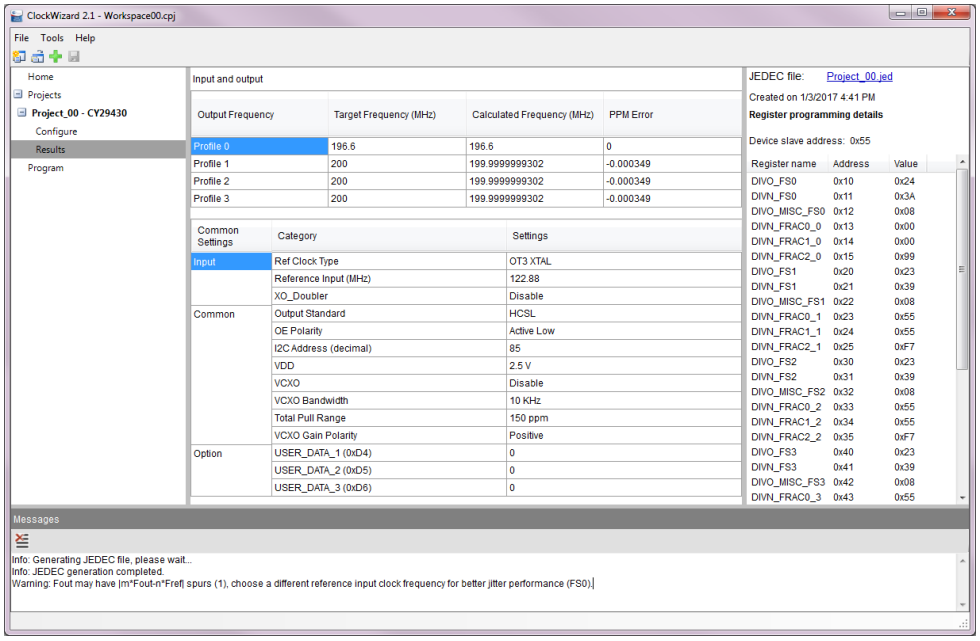
If the configuration of the device is such that, the output programmed frequency (f_{OUT}) and the input reference frequency (f_{REF}) holds a relationship of $0.01 \leq |m \times f_{REF} - n \times f_{OUT}| \leq 1.5$, the output phase noise plot will show spur within the integration bandwidth limit (12 kHz to 20 MHz). This will increase the RMS jitter value of the output. ClockWizard 2.1 will display this warning message while creating the configuration as shown in Figure 10.

Figure 9. Phase Noise Plot of CY29430 (programmed for V_{DD} = 2.5 V, output 622.08 MHz CML, non-VCXO mode)



The RMS jitter was calculated based on integrating the area under the phase noise curve over the 12 kHz to 20 MHz range. The experiment results shown here were achieved on the CY3676 and CY3677 EVKs.

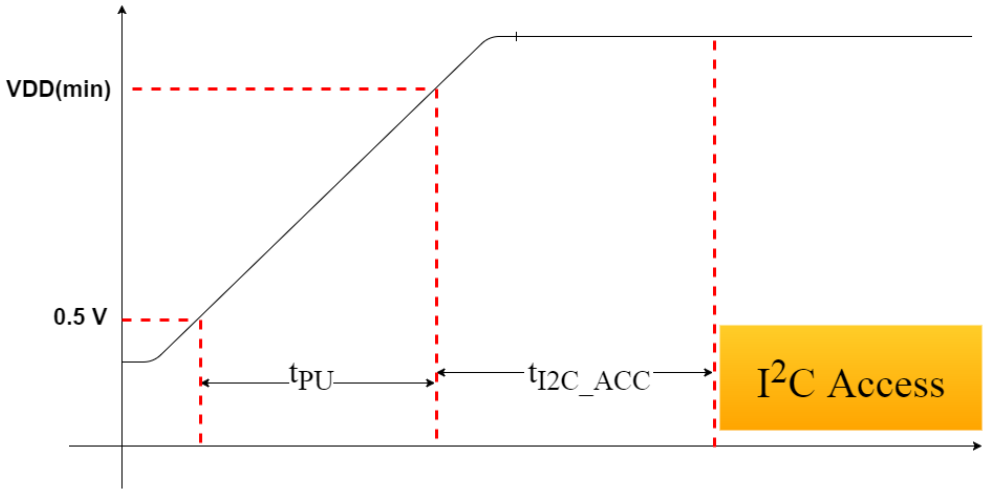
Figure 10. Example of a ClockWizard Configuration showing Possibility of Spur



4 Power Supply Specification

The system design should follow the power ramp guidelines of the CY294xx devices, as shown in Figure 11. Power ramp-up time (t_{PU}) is defined as the time taken to raise the supply voltage from 0.5 V to $V_{DD(min)}$. The t_{PU} of the device must be between 10 µs and 3 s. After the power reaches the minimum specified voltage, the internal state machine of the device loads the “eFuse” (one-time programmable nonvolatile memory) contents to the “NVMCopy” (volatile memory) location (See Section 5 for more details). It takes 5 ms time to complete this operation. Hence you need to wait a minimum of 5 ms for the first I^2C access (t_{I2C_ACC}). For programming a configuration to the nonvolatile memory section, the device power supply must be within the range 2.5 V ± 0.1 V.

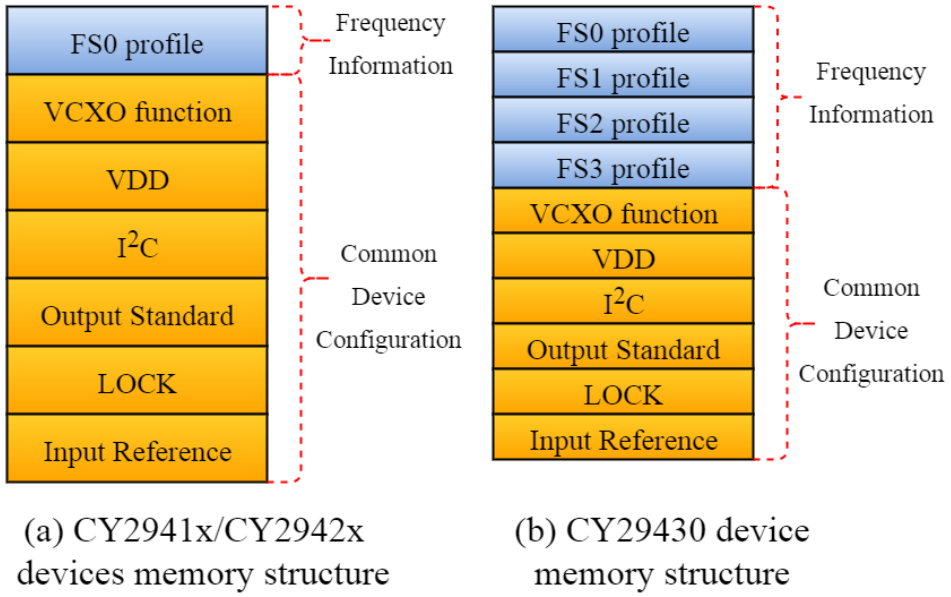
Figure 11. Power Ramp and I^2C Bus Access



5 Programming Interface

CY2941x, CY2942x, and CY29430 devices are either factory- or field-programmable. They support communication with the host over the I²C interface. The memory structure and accessibility sections are shown in Figure 12. The nonvolatile memory of CY294xx is a one-time programmable (OTP) eFuse. The eFuse can be partitioned into common device configurations and output frequency–related information. The common device configurations do not change with the output frequency and consist of the chip power supply, OE polarity, I²C device address, input reference, output standard, and VCXO functionality.

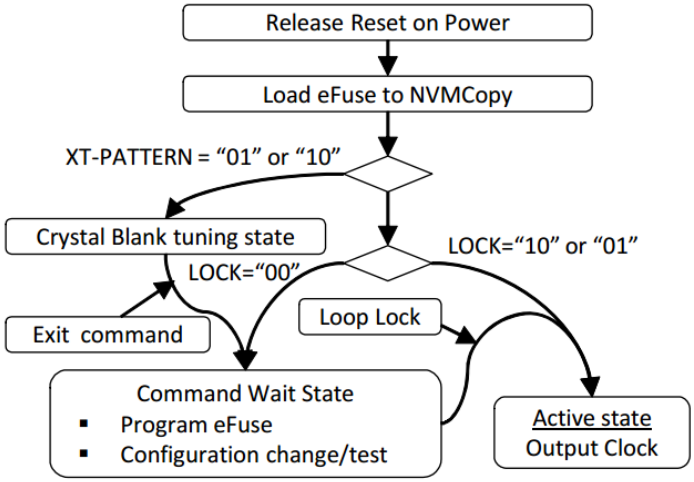
Figure 12. Memory Structure and Accessibility of CY294xx Devices



The devices have an internal state machine that controls the device behavior. The state machine loads the “eFuse” contents to “NVMCopy” (volatile memory) after reset, as indicated in Figure 13. The state machine enters one of the states—Command Wait state or Active state—according to the value of LOCK. In the Command Wait state, you can access all the registers and read/write the “NVMCopy” contents. The following features can be used in the Command Wait state:

- Program eFuse
- Copy eFuse to NVMCopy
- Loop Lock

Figure 13. State Diagram of CY294xx Devices Controlling Device Behavior



CY294xx devices also contain volatile memory (shown as “NVMCopy” in Figure 13) that stores an exact copy of the eFuse contents while power is ON. The chip settings depend on the contents of the volatile memory, and the output frequency depends on the configurations stored in it, as depicted in Figure 14. The volatile memory can be accessed and modified through the I^2C bus.

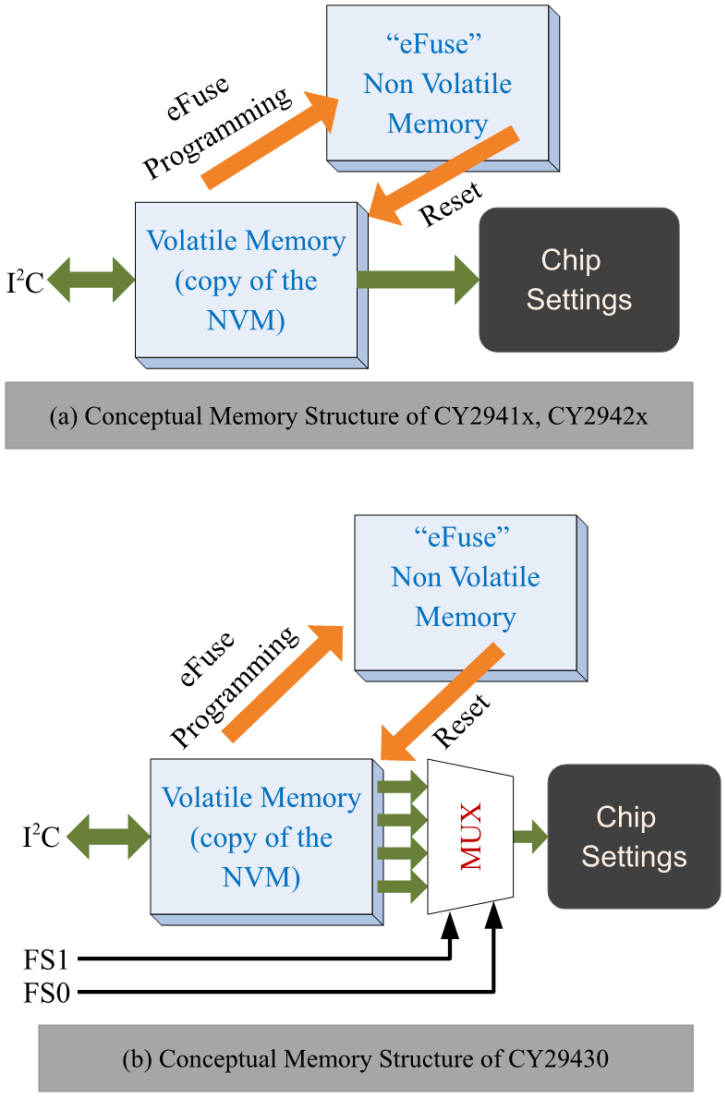
You can test the device functionality by issuing a Loop Lock command to enter the Active state without programming the LOCK. The device will function according to the settings.

When the LOCK is programmed to ‘10’, the device goes into the Active state, and the output clock is available after the completion of the power ON cycle. In the Active state, you can change the output frequency by applying Small Change or Large Change commands.

In the Command Wait state, you can configure the device with or without writing to the eFuse. This use case scenario validates the output frequency in the following conditions:

- You should write the JEDEC file to the eFuse only after proper validation of the configuration.
- Proceed to Loop Lock (optional) for testing purposes.
- The eFuse cannot be reprogrammed if it is moved to LOCK state. However, the output frequency can still be controlled using the Large Change or Small Change commands over the I^2C interface.

Figure 14. Memory Structure Showing Volatile and Nonvolatile Memory Structure (conceptual)



In the eFuse locked state, the output frequency can be changed using the Large or Small Change commands over the I^2C interface. You should write the JEDEC file to the eFuse only after proper validation of the configuration. The electrical parameters for eFuse programming are given in Table 6.

Table 6. Summary of eFuse Programming and I^2C Timing Specifications

Parameter	Description	Min	Max	Unit
V_{DD}	Device power supply for eFuse programming	2.4	2.6	V
t_{PU}	Power supply time from 0 to minimum specified V_{DD}	0.01	3000	ms
T_{PROG}	eFuse programming temperature	25	125	°C
f_{I2C}	I^2C bus clock frequency	–	400	kHz
t_{I2C_ACC}	Time to first I^2C access after power ON	5		ms

5.1 Program Routines

This section provides the routines and steps used to program the device successfully. With this information, you can develop your own programmer and transfer data through an I²C bridge to the device. Note that you must use ClockWizard 2.1 to generate the configuration file (JEDEC) or to get the values of register settings, which will be fed as input for the programming tool that you are developing.

At a high-level, the programming flow requires the steps shown in Figure 15. These steps are self-explanatory. During File Validation, a JEDEC file is validated by verifying the checksum (JEDEC Fuse sum) listed at the bottom of a JEDEC file against the locally calculated value of the checksum. You can skip File Validation if you are transferring only the register settings and not intending to use the JEDEC file. The Acquire Chip routine, shown in Figure 16, ensures that the device is on the bus and ready to communicate. This routine has two steps: communicate with the slave use the I²C address 0x55 (default for a blank part) and verify that the device has not been previously programmed using the lock-bit information (programmer must exit if the device is locked `LOCK <> 00`).

Figure 15. High-level Programming Flow

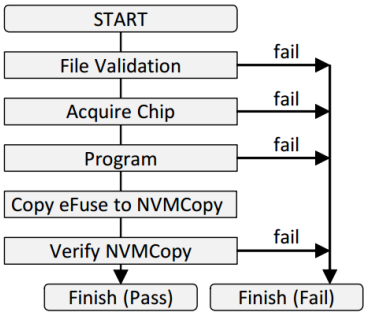
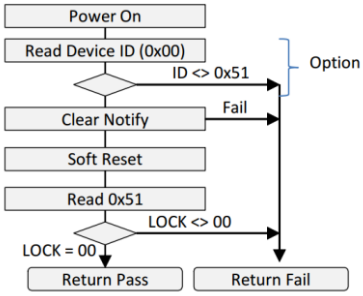


Figure 16. Acquire Chip, Check Device Status



As explained in the Programming Interface section, there are two types of programming:

1. eFuse programming, where you essentially blow the eFuse, and the configuration is set permanently into the non-volatile memory
2. Functional programming, where the configuration data is placed onto the volatile memory of the device

5.1.1 eFuse Programming

Figure 26 shows the flow for eFuse programming. It is recommended that eFuse programming be done at a supply of 2.5 V only. Figure 17 to Figure 21 show the routines involved during the eFuse.

The routines `Clear Notify` and `Soft Reset` prepare the device for data transfer. The configuration data is then written to the volatile memory (NVMCopy), and after verification the command `Program eFuse` is run. Further, you can verify the content written into the nonvolatile memory by having the content of the nonvolatile memory copied on to the volatile memory, NVMCopy, and comparing it with only the user-programmed data. You can ignore or mask the pre-programmed registers.

Figure 17. Clear Notify

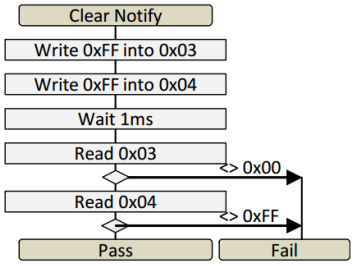


Figure 18. Soft Reset

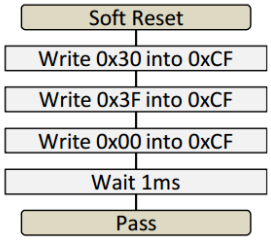


Figure 19. Exit Command

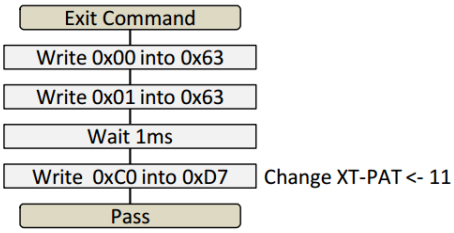


Figure 20. Program eFuse

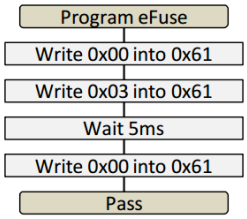
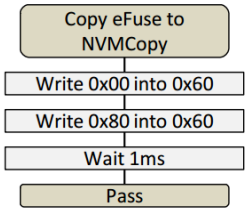


Figure 21. Copy eFuse to NVMCopy



5.1.2 Functional Programming

Figure 27 shows flow for the Functional programming. Functional programming is used when testing different clock configurations; you can write and verify your configuration multiple times without using the nonvolatile memory of the device. Note that the configuration is lost when the device is powered down. The programmer transfers user data to volatile memory, which is then copied on to the chip-registers (NVMRegisters). A `Loop Lock` command is then issued, which initiates the PLL start-sequence and output appears on the pin.

Figure 22. NVMCopy to NVMRegisters

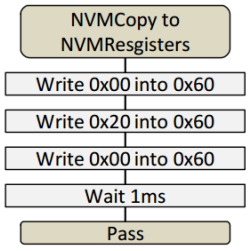
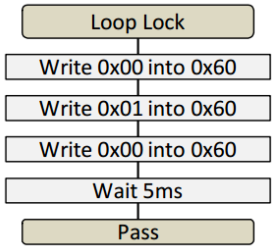


Figure 23. Loop Lock



5.1.3 Small Change Routine

The Small Change routine is used when the device is in Active state and you want to change the output frequency by ±500 ppm. This requires update to only the DIVN_FRAC registers values. Figure 24 shows the flow for Small Change.

Figure 24. Programming Flow for Small Change

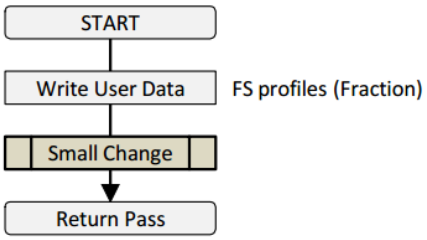
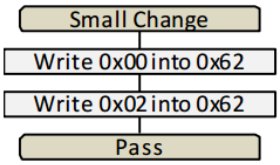


Figure 25. Small Change Routine



Once the new values of DIVN_FRAC are written to volatile memory, the Small Change command is issued as shown in Figure 25.

Calculating the new values for DIVN_FRAC requires reading the original values of DIVN_FRAC and the values of DIVO and DIVN_INT. For example,

Register value: 0x2F-39-0A-00-A0-CC (Addr: 0x10~0X15)
Find DIVO and DIVN_INT

```
0x10: 0x2F = 00101111b
0x11: 0x39 = 00111001b
0x12: 0x0A = 00001010b
0x13<FRAC 0>: 0x00 = 00000000b
0x14<FRAC 1>: 0xA0 = 10100000b
0x15<FRAC 2>: 0xCC = 11001100b
DIVO[8:0] = 0-00101111 = 47d
DIVN_INT[8:0] = 00-0111001 = 57d
```

The following are the steps to be followed leading up to finding the new value of DIVN_FRAC:

1. Calculating the current value of DIVN_total

   ```
   FRAC = 11001100-10100000-00000000;
   ```

   FRAC is 2's Complement expression, and the MSB is the sign bit. Therefore, the decimal value of FRAC is -3366912d, further it needs to be converted into decimal representation.

   ```
   Fraction: = FRAC / 2^24
   Fraction = -0.200683594
   DIVN_total = DIVN_INT + Fraction
   DIVN_total = 57 - 0.200683594 = 56.79931641
   ```

2. Consider the current output is 148.5 MHz. Thus, following variables get values:

   ```
   fOUT = 148.5 MHz
   DIVO = 47d ;(read from the device current state)
   fVCO = 148.5 * 47 = 6979.5 MHz
   ```

3. Consider a Small Change of 10 ppm is required.

   ```
   fOUT_new = 148.5 + 0.001485 = 148.501485-MH,
   ```

   The new values of DIVN_FRAC can be calculated from the following steps:

   Step 1:

   ```
   DIVN_total_new = fOUT_new * DIVO_original * DIVN_total_original / fVCO
   ```

   ```
   Calculation: DIVN_total_new = 148.501485 * 47 * 56.79931641 / 6979.5 = 56.7998844
   ```

   Step 2:

   ```
   Fraction = DIVN_total_new - DIVN_INT = 56.7998844 - 57
   ```

Step 3:

`FRAC_new = Fraction * 2^24 = -3357383d`

Step 4:

`The new DIVN_FRAC [23:0] = 0xCCC539` (MSB is the sign bit: 2's complement expression),

Figure 26. Flow for eFuse Programming

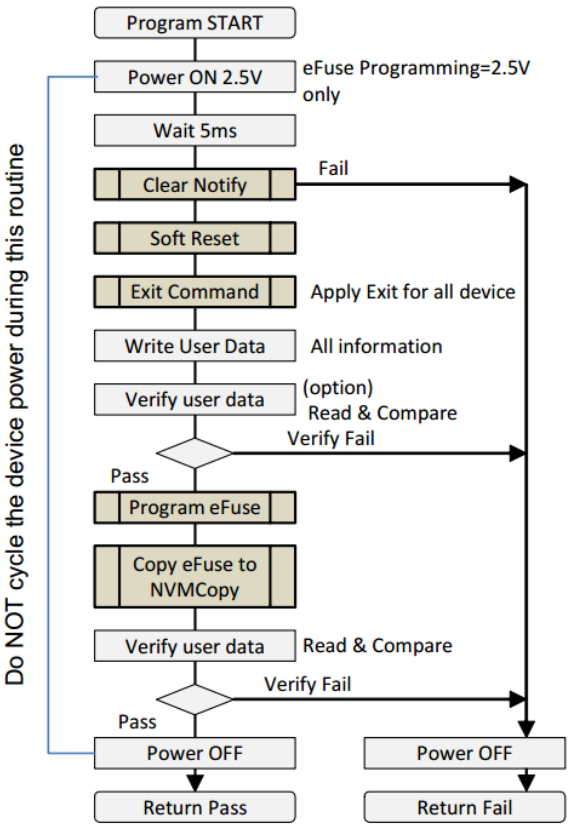
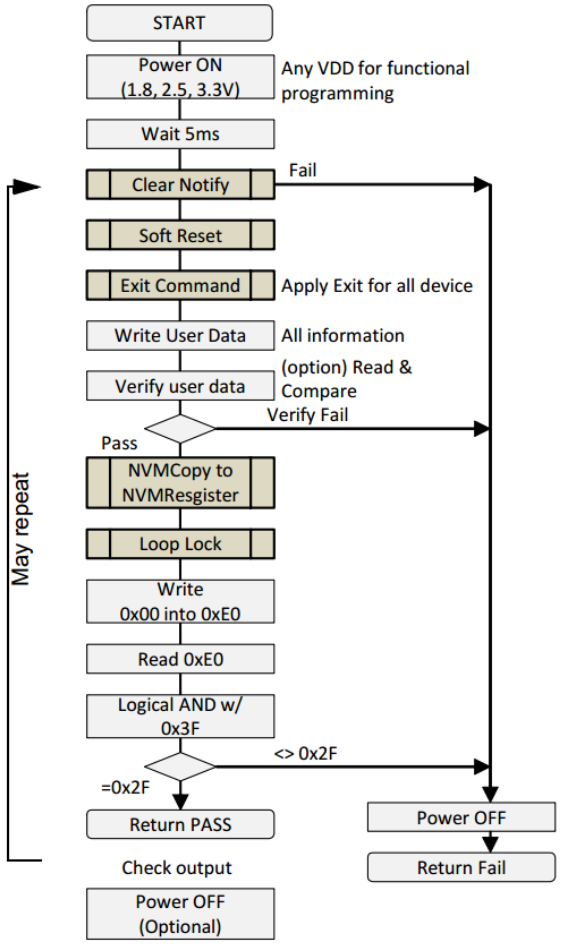


Figure 27. Flow for Functional Programming



Note: In Functional programming, as shown in Figure 27, compare against 0x2A (instead of 0x2F, see the second decision block), when OE is de-asserted and output is disabled.

6 Device I²C Interface

CY294xx devices have a hardware implementation of an I²C slave. You need to interface an I²C master controller to establish a communication channel to program the devices.

CY294xx accepts 8-bit units per the I²C protocol to write to the internal register map if the address is selected. These devices support two-wire serial interface in Fast Mode (400 kbps) and 7-bit addressing. They support single-byte access only. The device I²C address is programmable. See Table 7 to see the register map for the programming communication to the device.

You should use the programmed device address for communication, such as for Large Change or Small Change, after the eFuse is locked. The I²C functionality can be disabled by setting the I²C enable bit to OFF, but the device remains active for the 0x55 I²C transaction. The device is shipped with a default, factory-programmed I²C address of 55h. You should maintain the slave addresses when multiple I²C devices are connected to the same bus and access to the 0x55 device occurs.

Table 7. Internal Memory Map (for Frequency Configuration) of CY294xx Device Family

Address	Name		bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0x10	DIVO_FS0		DIVO < 7 : 0 >							
0x11	DIVN_INT_FS0		DIVO<8>	DIVN_INT < 6 : 0 >						
0x12	DIVO_MISC_FS0		*Reserved*	ICP_OFFSET	DIVN_INT < 8 : 7 >		ICP_BIN < 2 : 0 >			PLL_MODE
0x13	DIVN_FRAC0_FS0		DIVN_FRAC < 7 : 0 >							
0x14	DIVN_FRAC1_FS0		DIVN_FRAC < 15 : 8 >							
0x15	DIVN_FRAC2_FS0		DIVN_FRAC < 23 : 16 >							
0x20	DIVO_FS1		DIVO < 7 : 0 >							
0x21	DIVN_INT_FS1		DIVO<8>	DIVN_INT < 6 : 0 >						
0x22	DIVO_MISC_FS1		*Reserved*	ICP_OFFSET	DIVN_INT < 8 : 7 >		ICP_BIN < 2 : 0 >			PLL_MODE
0x23	DIVN_FRAC0_FS1		DIVN_FRAC < 7 : 0 >							
0x24	DIVN_FRAC1_FS1		DIVN_FRAC < 15 : 8 >							
0x25	DIVN_FRAC2_FS1		DIVN_FRAC < 23 : 16 >							
0x30	DIVO_FS2		DIVO < 7 : 0 >							
0x31	DIVN_INT_FS2		DIVO<8>	DIVN_INT < 6 : 0 >						
0x32	DIVO_MISC_FS2		*Reserved*	ICP_OFFSET	DIVN_INT < 8 : 7 >		ICP_BIN < 2 : 0 >			PLL_MODE
0x33	DIVN_FRAC0_FS2		DIVN_FRAC < 7 : 0 >							
0x34	DIVN_FRAC1_FS2		DIVN_FRAC < 15 : 8 >							
0x35	DIVN_FRAC2_FS2		DIVN_FRAC < 23 : 16 >							
0x40	DIVO_FS3		DIVO < 7 : 0 >							
0x41	DIVN_INT_FS3		DIVO<8>	DIVN_INT < 6 : 0 >						
0x42	DIVO_MISC_FS3		*Reserved*	ICP_OFFSET	DIVN_INT < 8 : 7 >		ICP_BIN < 2 : 0 >			PLL_MODE
0x43	DIVN_FRAC0_FS3		DIVN_FRAC < 7 : 0 >							
0x44	DIVN_FRAC1_FS3		DIVN_FRAC < 15 : 8 >							
0x45	DIVN_FRAC2_FS3		DIVN_FRAC < 23 : 16 >							

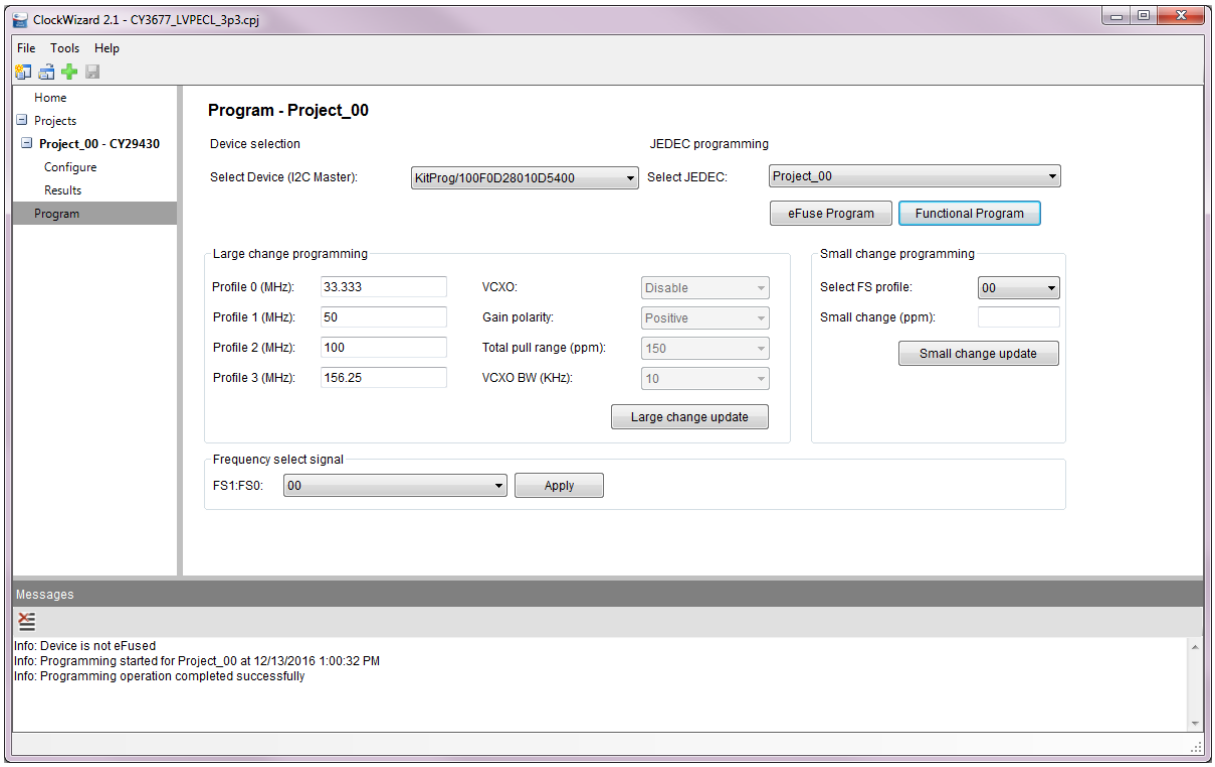
7 Large Change and Small Change Trigger

The change in the fractional part of the PLL division factor is referred to as a mall change trigger. In the large change trigger, the PLL completely shuts down and comes up with the new frequency. When the device is in the Active state, you can change the output frequency by applying the Small Change or Large Change commands.

- Small Change refers to the case in which the frequency changes within ±500 ppm. The frequency information will be loaded through I^2C, and the output frequency will change without any glitch from its original frequency to the new frequency. Note that the Small Change trigger functionality is not supported if the PLL is configured in integer mode.
- Large Change refers to the case in which the frequency changes more than ±500 ppm, and the change is done through an I^2C. The change of output frequency through the FS state change is similar to the Large Change operation. The device will recalibrate and reconfigure the PLL, and the output will be unstable until this process is completed.

The large change trigger and small change trigger functionality can be verified in ClockWizard 2.1 using the CY3676 or CY3677 EVK hardware platform. The selection of settings is shown in Figure 28.

Figure 28. Large Change Trigger and Small Change Trigger Selection



Note: VCXO enabled part can be made non-VCXO with large trigger command executed over I^2C interface. However, a non-VCXO part cannot be made VCXO through large trigger change command. The input impedance of the VC pin cannot be changed through programming (it is always high impedance > 5 MOhm).

The I^2C configurable parameters through large change update are:

- Output frequency
- VCXO features:
  - VCXO gain polarity
  - Total pull range (ppm)
  - VCXO BW (kHz)

8 JEDEC File

```
# < Checksum: 20B8D58B      << CYPRESS CHECKSUM
# s 29430                   << Device Information
# f --- INPUT SECTION XIN ---
# f 114.285000 ;XIN (MHz)
# f OT3 ;XO mode
# f Disable ;XO doubler
# f Enable ;VCXO enable/Disable
# f 50 ;VCXO pull range
# f 10k ;VCXO BW
# f Positive ;VCXO polarity
# f --- DEVICE CONFIGURATION ---
# f 2.5V ;VDD Range
# f PECL ;OUTPUT standard
# f actH ;OE polarity
# f Enable ;I2C BUS
# f 55 ;I2C device address (HEX)
# f --- OUTPUT FREQUENCY ---
# f 156.250000 ;FS0
# f 155.520000 ;FS1
# f 688.812300 ;FS2
# f 322.562562 ;FS3
*
QP0016* QF2048* G0*
L00128                      << FS0 configuration
00101100001111000000101010100011000110000010100 0*
L00256                      << FS1 configuration
00101100001111000000101010100001001001011110000 0*
L00384                      << FS2 configuration
00001011010000100000011011101111011100000010011 00*
L00512                      << FS3 configuration
00010110001111100000100001111111011111011001011 1*
L00640                      << Common configuration
00000100101010000001110110110000110101011001010111000101010100000*
L01696                      << User option configuration
00000000000000100000010*
C0D3E*                      << FUSE sum = 0D3E
0000
```

The JEDEC file of CY294xx devices contain the entire user-created profile information such as PLL configuration, output standard, input reference, VCXO, OE polarity, and other user-configurable options. A typical JEDEC file example appears as follows:

The double-underlined fields contain LOCK (2-bit), R_CAL (4-bit), and RC_CAL (4-bit) information respectively. The R_CAL and RC_CAL fields are factory-programmed. The memory address and its content can be calculated from the JEDEC file in the following way (also see Table 7):

- Memory address = xxxx/8 (xxxx = 4-digit number of L0xxxx in the JEDEC). Binary data stored in the memory address (L00128, that is, 0x10) is 00101100-00111100-00001010-10100011-00011000-00101000*.
- Data in memory location 0x10 is 0x2C (00101100) sets required DIVO parameter
- Data in memory location 0x11 is 0x3C (00111100) sets required DIVO, DIVN_INT parameters.
- Data in memory location 0x12 is 0x0A (00001010) sets required ICP, DIVN_INT, PLL_MODE parameters
- Data in memory location 0x13 is 0xA3 (10100011) sets required DIVN_FRAC0 parameter.
- Data in memory location 0x14 is 0x18 (00011000) sets required DIVN_FRAC1 parameter.
- Data in memory location 0x15 is 0x28 (00101000) sets required DIVN_FRAC2 parameter.

9 Layout Guidelines

As CY294xx devices support very high-frequency (up to 2.1GHz for LVDS, LVPECL, LVPECL2, and CML standards) outputs and need a 114.285-MHz OT3 or 122.88-MHz HFF crystal as input, the layout should follow certain design guidelines.

- The input crystal should be placed very near to the IC input pins. Figure 29 shows a typical crystal routing scheme on the PCB to CY29430 input.
- Routing of output traces needs ground shielding (impedance matching and minimum cross-talk). The trace width, trace-to-ground spacing, and PCB stack should be such that the controlled impedance of 50-Ω is maintained everywhere. You should design the transmission lines on the PCB maintaining either individual 50-Ω lines or a 100-Ω differential line. It is recommended that you follow the design guidelines mentioned in the Fab Notes of the CY3676 and CY3677 EVKs.
- The VC pin of CY29430 needs an RLC filter on the PCB. The filter at the VCXO input reduces noise coupling and thereby RMS phase jitter. It is recommended that you place the VCXO filter very close to the VC pin of the device. Figure 30 shows the typical VCXO input filter design recommended on the PCB.

The schematic and layout guidelines are shown in Figure 29 and Figure 30. It is recommended that you refer to the schematic and layout files and follow the crystal and output transmission lines according to the CY3676 and CY3677 EVKs.

Figure 29. Schematic and Layout Example of Crystal Circuitry

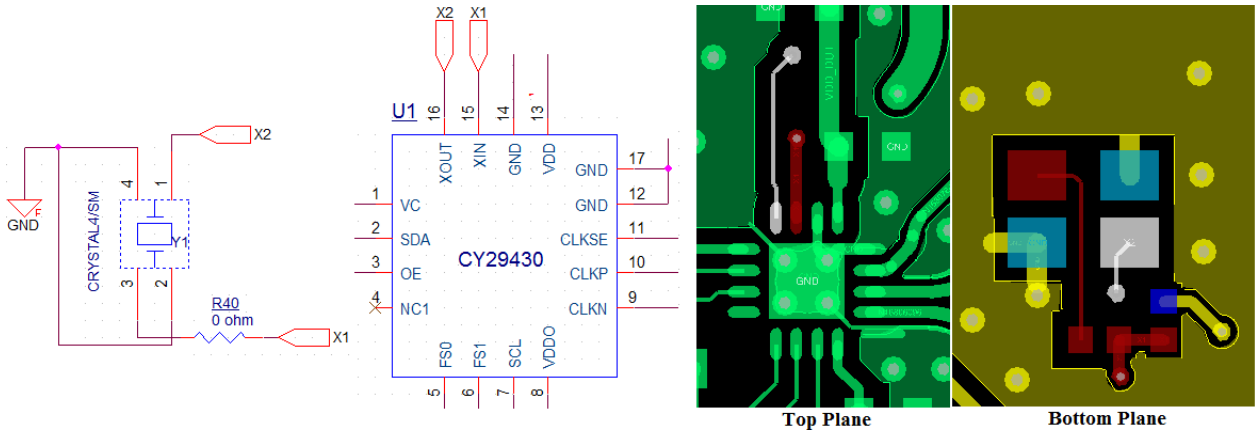
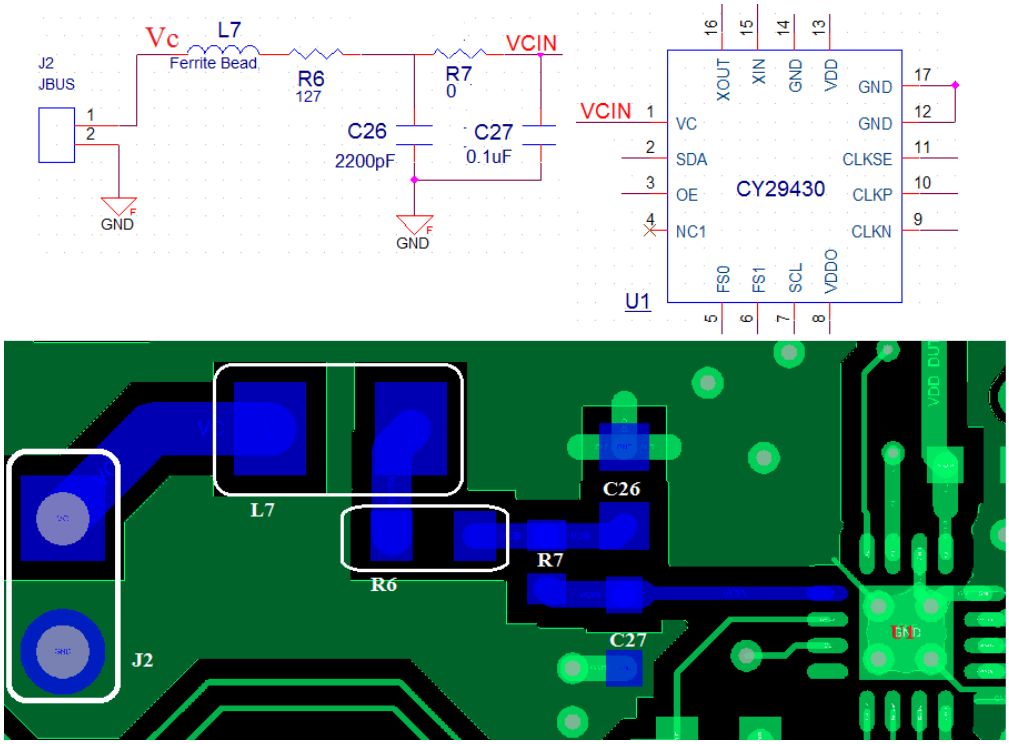


Figure 30. Filter Circuitry of VCXO Input



10 Summary

This application note described the CY294xx device features in detail and gave relevant references of hardware and software platforms to evaluate clock features.

11 Acronyms

Table 8. List of Acronyms Used in this Document

Acronym	Definition
CML	Current-Mode Logic
DSL	Digital Subscriber Line
FS	Frequency Select
GbE	Gigabit Ethernet
HCSL	High speed Current Steering Logic
HFF	High Frequency Fundamental
I^2C	Inter-Integrated Circuit
JEDEC	Joint Electron Device Engineering Council
LFF	Low Frequency Fundamental
LVCMOS	Low Voltage Complementary Metal Oxide Semiconductor
LVDS	Low Voltage Differential Signaling
LVPECL	Low Voltage Positive Emitter Coupled Logic
OTN	Optical Transport Network
OT3	Third Overtone
SDH	Synchronous Digital Hierarchy
SONET	Synchronous Optical Networking
TCXO	Temperature Compensated Crystal Oscillator
VCXO	Voltage Controlled Crystal Oscillator

Document History

Document Title: AN210253 - CY294xx High-Performance Clock: Getting Started and Best Design Practices

Document Number: 002-10253

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5268540	TAVA	06/09/2016	New application note.
*A	5666985	TAVA	03/21/2017	Updated Introduction: Updated Table 1. Added Footnote 1 and referred the same footnote in “VCXO Feature” in Table 1. Added a note below Figure 1. Updated Input and Output Settings and Programmable Features. Updated Table 2. Updated Input Settings. Updated Figure 3. Updated Output Settings. Updated description. Updated Figure 4. Added Figure 6. Added Table 3. Added Figure 7. Updated Other Configurable Parameters and Design Guidelines. Updated Table 4. Replaced “VIN” with “V_C” in “IC pin” column and updated details in all columns. Added Footnote 2 and referred the same footnote in “Functionality” column corresponding to V_C pin. Updated RMS Jitter Performance. Updated description. Updated Figure 8. Updated Figure 9 (Updated caption only). Added Figure 10. Updated Programming Interface. Updated Figure 13. Updated Device I2C Interface. Updated Table 7. Updated Large Change and Small Change Trigger. Updated description. Updated Figure 28 and added a Note below. Updated JEDEC File. Updated description. Updated to new template. Completing Sunset Review.
*B	6152124	PAWK	04/23/2018	Migrated to new template Added Section 5.1 Program Routines

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709