Phase-Locked Loop (PLL) 6

PLL is the heart of practically all electronic components and or modules where different clock frequencies are required to synchronize the data transmitting and receiving to and from externals respectively. The input clock to the PLL is much lower than the DSP maximum clock frequency. PLL is typically being used as a frequency synthesizer to generate the clock for the DSP core. For example, the input clock to the 1.2GHz DSP [1] is 66MHz.

PLL is an analog circuit that is very sensitive power supply noise. Noise causes jitter and excessive jitter causes timing violations which lead to system failures. The two main PLL architectures are analog PLL (APLL) and digital PLL (DPLL). Understanding the differences help to make the design tradeoffs often required for minimizing noise and jitter caused by external circuitries, such as the power supply and other noisy switching devices.

6.1 ANALOG PLL (APLL)

As stated, PLL generally functions as a frequency synthesizer, multiplying the input clock by an integer. This integer is a ratio of the feedback counter M divided by the input counter N as shown in Figures 6.1.

T.T. Tran, *High-Speed DSP and Analog System Design*, DOI 10.1007/978-1-4419-6309-3_6,

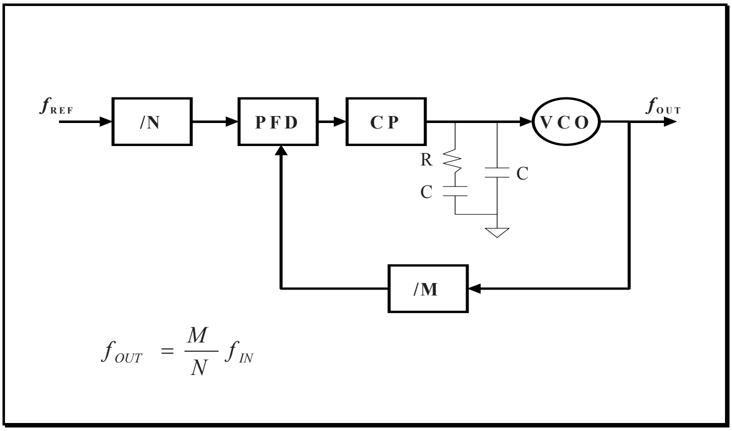


Figure 6.1. Analog PLL

The following table provides a brief description of each block shown Figure 6.1 for the APLL.

Table 6.1. Analog PLL Description

Name:	*Description:*	*Function:*
/N	Divide-by-N	Divide-by-N counter scales down the input frequency
PFD	Phase-Frequency Detector	PFD compares the frequency and the phase of the input and the feedback clock signals and generates an error signal.
CP	Charge Pump	This is typically a constant current source controlled by the error signal output from the PFD block.
VCO	Voltage Controlled Oscillator	This VCO oscillates at a frequency controlled by the DC input voltage derived from integrating the error signal.
/M	Divide-by-M	Divide-by-M counter scales down the output frequency.

The following provides an overview of how the PLL functions as a frequency synthesizer:

1. The reference clock is connected to the PDF input. The Divide-by N counter reduces the input frequency.
2. The PDF compares the output of the Divide-by-M counter with the reference clock and generates an error signal.
3. Base on the error signal, the CP charges or discharges the current store on the loop filter, an RC filter shown in Figure 6.1. This increases or decreases the VCO control voltage. For some PLL architectures, increasing the VCO control voltage increases the clock frequency and decreasing the voltage lowers the clock output frequency.
4. The phase correction continues until both the feedback signal from the Divide-by-M counter and the reference clock are synchronized. At this point, the error voltage should be zero.
5. The output clock frequency is equal to the ratio of the Divide-by-M counter and the Divide-by-N counter multiply by the input clock frequency. As a rule of thumb, a higher multiplier ratio yields higher jitter, so keep the M and N ratio as low as possible when designing with PLLs. The PLL output frequency, f_{out}, for a given input frequency, f_{in}, is

$$f_{out} = \frac{M}{N} f_{in} \text{ , where M is the PLL} \qquad (6.1)$$

feedback counter and N is the input counter.

6.1.1 PLL Jitter

Jitter in PLL design is defined as the signal timing displacement from a reference clock. The three main sources of DSP PLL jitter are jitter generated by the reference clock itself, power supply noise and noise coupling from external and internal circuitries. The following lists important techniques for designers to minimize the DSP PLL jitter:

- Select a reference clock oscillator with the lowest jitter specification possible.
- Heavily filter the clock circuit to reduce the effect of noise on output jitter. See the following section on PLL isolation.
- Use a series termination resistor at the output of the reference clock to control the edge rate.
- Distribute the clock differentially if possible. Differential signals reject common mode noise and crosstalk.
- Set the multiplier as low as possible to achieve maximum DSP operating frequency. Keep in mind that a higher multiply ratio yields higher output jitter.

In all cases, jitter can be minimized but cannot be eliminated. The three types of deterministic jitter [2] important for frequency synthesizers and DSP performance are long term jitter, cycle-to-cycle jitter and period jitter.

Long Term Jitter: See Figure 6.2 where long-term jitter is defined as a time displacement from the ideal reference clock input over a large number of transitions. Long term jitter measures the deviation of a rising edge over a large number of cycles (N) after the first rising edge.

Peak-to-Peak Jitter = Max Period (N cycles) – Min Period (N cycles),

where Max Period is the maximum period equal to 1 divided by the operating frequency measured at N number of cycles and Min Period is the minimum period equal to 1 divided by the operating frequency measured at N number of cycles.

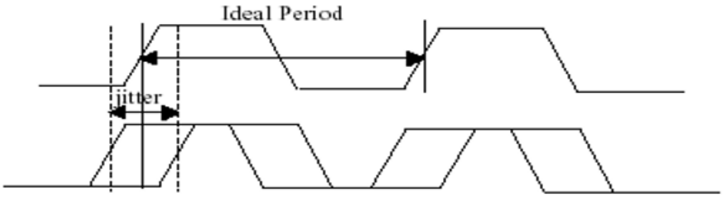


Figure 6.2. Long Term Jitter

The long jitter can be measured using an automatic jitter measurement equipment [3] or using a high speed digital sampling scope. Here are the steps to measure long term jitter using a scope:

- Use a high-speed 10GHz sampling oscilloscope.
- Use the input clock to trigger the scope and set the scope in the Infinite Persistence mode.
- The deviation is measured from the first rising edge to the Nth cycle. The "fuzz" shown on the scope in Figure 6.3 is the long term jitter.

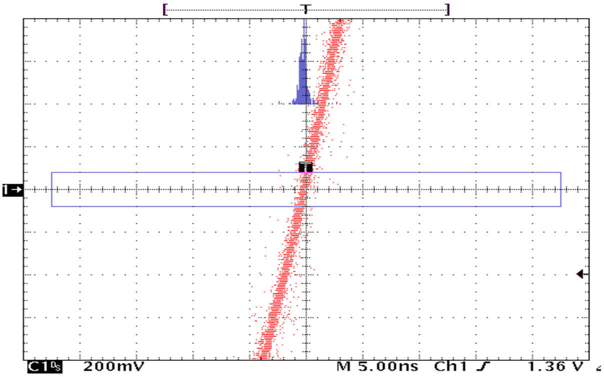


Figure 6.3. Jitter Measured by Digital Scope

Cycle-To-Cycle Jitter: See Figure 6.4 where cycle-to-cycle is defined as the deviation of the clock period between two consecutive clock cycles.

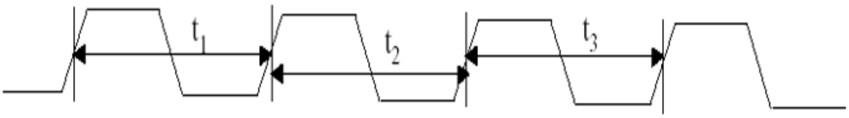


Figure 6.4. Cycle-To-Cycle Jitter

In Figure 6.4, the cycle-to-cycle is measured by subtracting t_2 from t_1, t_3 from t_2, and so on.

Cycle-To-Cycle Jitter Measurement:

This is a difficult parameter to accurately measure with the high-speed sampling scope. Because the sampling scopes on the market today are not capable of measuring jitter in a few picoseconds range. The best way is to use a Timing Interval Analyzer (TIA) which captures one cycle at a time and compares the timing differences between two consecutive cycles. Another method is to use a scope with a cycle-to-cycle jitter measurement option. This method is outlined as follows:

- Use a high-speed 10GHz sampling oscilloscope [4] with cycle-to-cycle jitter option.
- Trigger the PLL output clock and measure the cycle-to-cycle jitter. Use the windowing method to measure the changes from one cycle to another.

Period Jitter: See Figure 6.5 where period jitter is defined as the maximum deviation in the clock's transition from its ideal position. These periods are non-successive.

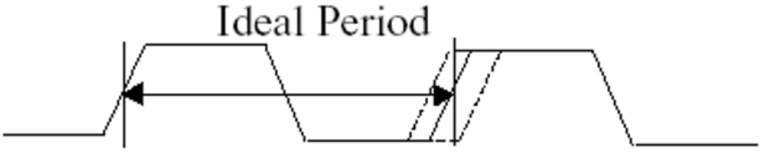


Figure 6.5. Period Jitter

Period Jitter Measurement:

- Use a high-speed 10GHz sampling oscilloscope [4].
- Set the scope in the Infinite Persistence mode and trigger the PLL clock output on the rising edge.
- Measure the "fuzz" shown on the screen at the next rising of the clock.

In summary, the jitter measurements can either be done using a high-speed digital sampling scope, Timing Interval Analyzer (TIA) or an automatic jitter measurement system [3].

6.2 DIGITAL PLL (DPLL)

The main differences between the APLL and DPLL are that the DPLL replaces the analog filter with a digital controller block that filters the phase error in the digital domain and replaces the VCO with a Digital Controller Oscillator (DCO). The advantages of the DPLL are:

- The DPLL supports a wide range of input frequency from 30KHz to 65MHz or higher.
- The DPLL design requires a smaller silicon area to implement and consumes less power than the APLL.
- The DPLL does not have analog filter components such as capacitors which can cause leakage current. This leads to lower power consumption.
- The DPLL block is scalable and portable. The same design can be implemented on different process technology nodes.
- The DPLL design can be optimized for low jitter. But it may not be acceptable for jitter sensitive designs such as USB, audio and video clocks.

The disadvantages of the DPLL are:

- It is very sensitive to external and internal power supply noise. Use of a linear regulator plus a Pi filter to isolate the power supply from the DPLL is recommended.
- Low power supply rejection ratio.
- In addition to power supply sensitivity, quantization noise and phase detector dead zone are the major sources of output jitter.
- Requiring a DAC block to control the oscillator. This makes the DPLL more sensitive to noise.

Figure 6.6 shows a typical DPLL architecture [6] and Table 6.2 describes the function of each block in the architecture.

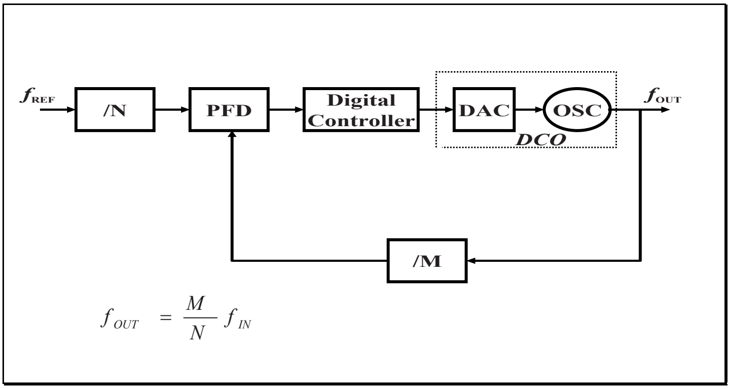


Figure 6.6. Digital PLL

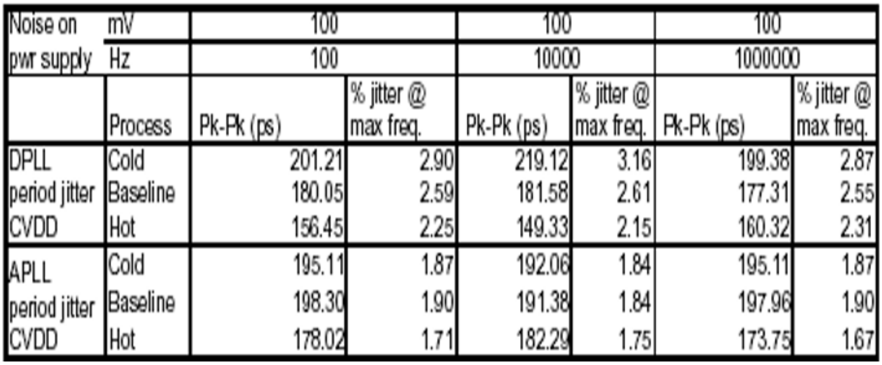
Table 6.2. Digital PLL Description

Name:	*Description:*	*Function:*
/N	Divide-by-N	Divide-by-N counter scales down the input frequency
PFD	Phase-Frequency Detector	PFD compares the frequency and the phase of the input and the feedback clock signals and generates an error signal.
Digital Controller	Digital Controller	This digital filter block detects the phase error information and digitally controls the oscillator.
DCO	Digital controlled oscillator	This DCO converts the control code to analog levels and generates a stable clock output.
/M	Divide-by-M	Divide-by-M counter scales down the output frequency.

APLL and DPLL Jitter Characterization

Table 6.3 shows a jitter comparison between an analog and a digital PLL that shows the effects of process variation where Hot is fast, Cold is slow and Baseline is typical. In this DSP design, the DPLL power supply is isolated by an internal low dropout regulator (LDO) while the APLL is connected directly to the common power supply plane. To test the noise sensitivity, 100mV of noise modulating from 100Hz to 1MHz is injected into the power supply rails. The results showed that the peak-to-peak period jitter is less than 3% for the DPLL and is less than 2% for the APLL. With the LDO, the DPLL jitter is less than 4% up to 50mV of noise on the power supply.

Table 6.3. APLL and DPLL Jitter Comparison



Noise on	mV	100		100		100	
pwr supply	Hz	100		10000		1000000	
	Process	Pk-Pk (ps)	% jitter @ max freq.	Pk-Pk (ps)	% jitter @ max freq.	Pk-Pk (ps)	% jitter @ max freq.
DPLL	Cold	201.21	2.90	219.12	3.16	199.38	2.87
period jitter	Baseline	180.05	2.59	181.58	2.61	177.31	2.55
CVDD	Hot	156.45	2.25	149.33	2.15	160.32	2.31
APLL	Cold	195.11	1.87	192.06	1.84	195.11	1.87
period jitter	Baseline	198.30	1.90	191.38	1.84	197.96	1.90
CVDD	Hot	178.02	1.71	182.29	1.75	173.75	1.67

Designers need to be careful when injecting a signal onto the power supply to do jitter measurements. The nature of the signal used for simulating a noisy power supply condition can have a major impact on the PLL jitter. A squarewave signal with a frequency less than the PLL bandwidth characterizes the worst case PLL jitter. As far as the amplitude of the noise, the peak-to-peak voltage has to be within the power supply limits. For example, for a 1.6V +/-3% Core, the maximum acceptable peak-to-peak noise is 96mV (-48mV min and +48mV max).

6.3 PLL ISOLATION TECHNIQUES

As shown in previous sections, both an APLL and a DPLL are sensitive to noise, especially to noise frequency within the PLL bandwidth. PLL isolation is needed in order to prevent the high frequency PLL signal from propagating out of the PLL section and affecting other circuits. PLL isolation can also attenuate the external noise propagating to the PLL circuit which causes excessive jitter. In many cases, external power supply noise causes the PLL to go unstable and the DSP to lock-up randomly.

6.3.1 Pi and T Filters

The two important filter schemes discussed in this document to isolate the PLL are low frequency filtering and high frequency filtering. For high frequency filtering, a Pi or T network filter can be used as shown in Figures 6.7 and 6.8:

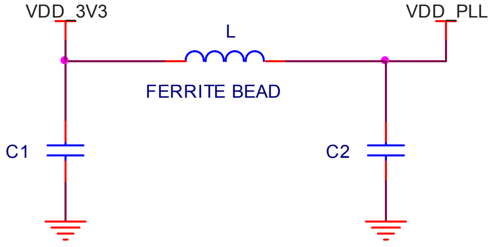


Figure 6.7. Pi Filter Circuit

The Pi filter circuit consists of one ferrite bead, L and two capacitors, C_1 and C_2. This circuit provides both input and output isolation where noise from the 3.3V supply is attenuated by the ferrite bead and the C_2 capacitor and noise generated by the PLL circuit is isolated by the ferrite bead and the C_1 capacitor. Refer to Chapter 5 for the filter design and simulation information.

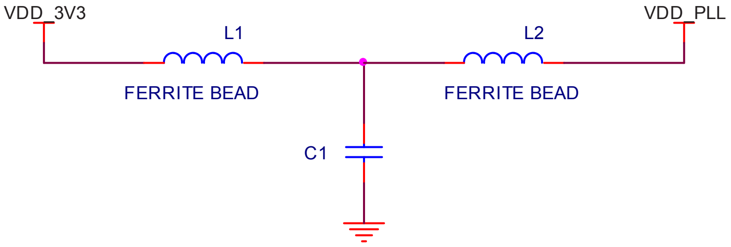


Figure 6.8. T Filter Network

A T filter consists of two ferrite beads and one capacitor as shown in Figure 6.8. Just like in a Pi filter, 3.3V supply noise is attenuated by the L_1 ferrite bead and the C_1 capacitor and PLL noise is isolated by the L_2 ferrite bead and C_1 capacitor. Refer to Chapter 5 for the filter design and simulation information.

Both Pi and T circuits are good for filtering high frequency noise but they are not as effective for low frequency filtering since ferrite beads have almost zero AC impedance at low frequency. The Pi circuit has an advantage over the T circuit. Because this topology makes it possible to place the capacitor closer to the PLL voltage pin that ensures low impedance to ground and also the smallest current loop area, which reduces noise and EMI.

For low frequency isolation, there are two common techniques, Pi filter with large bulk capacitor and linear voltage regulator.

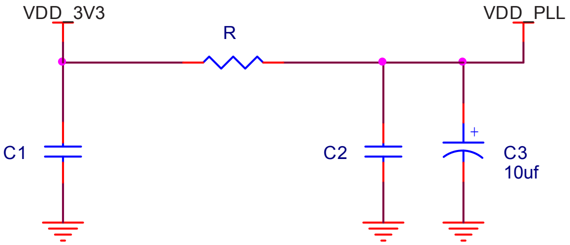


Figure 6.9. Low Frequency Pi Filter

One method for low frequency filtering is shown in Figure 6.9, where a resistor R replaces the ferrite bead and a bulk capacitor C_3 (10uF to 33uF) is added to the circuit. Low frequency noise is attenuated by the resistor R and the bulk capacitor C_3. The resistor needs to be selected such that the voltage drop across the resistor is negligible. The low frequency -3dB corner for this filter is approximated by Equation 6.2. Notice that C_1 and C_2 are negligible in this case, since its value is a lot lower than the bulk capacitor C_3.

$$f_{-3dB} = \frac{1}{2\pi RC_3} \tag{6.2}$$

Design Example 6.1:

Design a PLL power supply filtering circuit that provides a 20dB attenuation at 15KHz. The tolerance for the PLL power supply is +/-5% and the maximum current consumption is 10mA.

Design steps are:

- The Pi filter circuit in Figure 6.9 is a single pole filter neglecting C_1 and C_2. For a single pole filter, the attenuation is -20dB/decade starting at the -3dB corner frequency as shown in Equation 6.2.
- f_{-20dB} = 10xf_{-3dB} , slope is 20dB/dec so the frequency at -20dB is equal to 10 times the frequency at -3dB. Therefore, f-3db = (15KHz)/10 = 1.5KHz.
- From Equation 6.2,

$$f_{-3dB} = \frac{1}{2\pi RC_3} = 1.5KHz$$

$RC_3 = 1.06\text{x}10^{-4}$,

Let R = 10 ohms,

C3 = 10.6uF or 10uF.

- The voltage drop across the resistor is
  - V = IR = 10x10^{-3} = 0.01V. This is very small and is way within the power supply limits of 3.3V±5%.
- The resistor power dissipation is
  - P=VI where V=IR -> P=I^{2}R =(10mA)^{2}x10 = 0.01mW. This small power dissipation allows designers to use a very small size resistor for this filter.
- Let C_1 and C_2 be a 0.01uF capacitor since this is a good high frequency decoupling capacitor as discussed in Chapter 5. The final circuit and simulation are shown in Figure 6.10 and Figure 6.11 respectively.

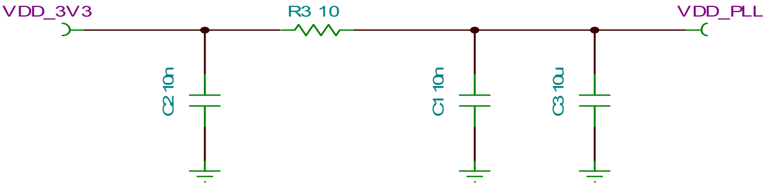


Figure 6.10. Final Pi Filter Circuit for PLL

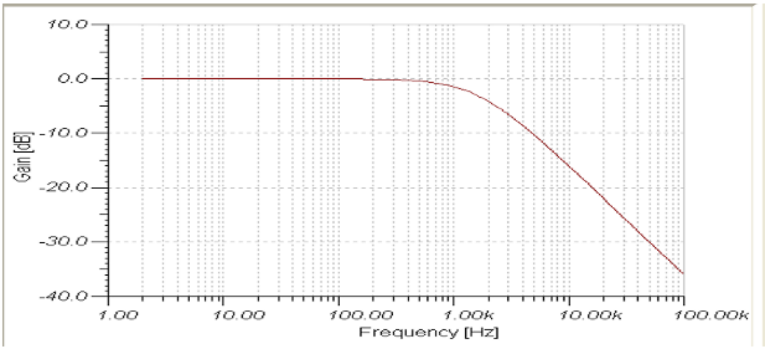


Figure 6.11. Final Pi Circuit Simulation

As shown in Figure 6.11 simulation, the -3dB corner frequency is at 1.5KHz and the -20dB attenuation is at 15KHz. These are the design specifications.

In this design example, there is an IR voltage drop across the resistor R so it is very important to select the resistance to guarantee that the PLL supply voltage range are within the specified limits for worst case PLL current consumption.

6.3.2 Linear Voltage Regulator

Another method of low frequency filtering is to use a linear voltage regulator. This method has the least effect on PLL performance. The linear regulator typically has good line regulation and power supply rejection characteristics which prevent low frequency transients and high frequency noise from entering the PLL circuit. The method shown in Figure 6.12 is more expensive to implement than other methods described previously. But it is extremely effective in keeping the PLL voltage as clean as possible to guarantee lowest PLL jitter. Refer to Chapter 4 for design considerations.

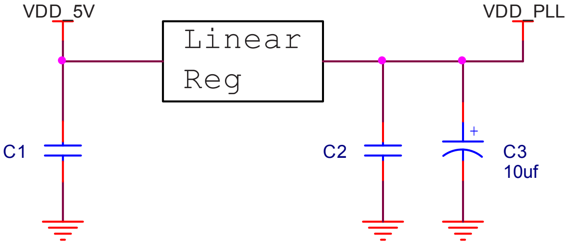


Figure 6.12. Noise Isolation with Voltage Regulator

One issue with using a linear regulator is that it does not reject high frequency very well. As shown in Figure 6.13, the ripple rejection is approaching 0dB (no rejection at all) for noise that is higher than 1MHz. This high frequency noise can cause more jitter in the PLL.

In summary, the best way to isolate PLL is using a combination of Pi filter and linear regulator. In this case, the Pi filter can be implemented with ferrite bead and capacitors so there is no IR drop across the resistor as shown in the previous example. The final circuit is shown in Figure 6.14.

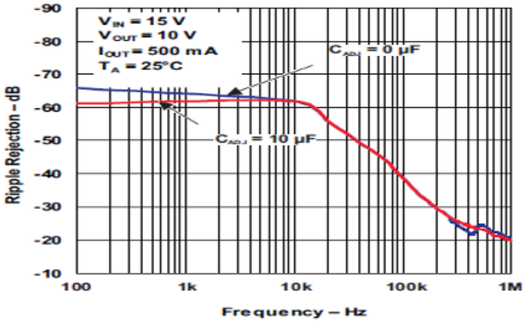


Figure 6.13. Linear Regulator Ripple Rejection

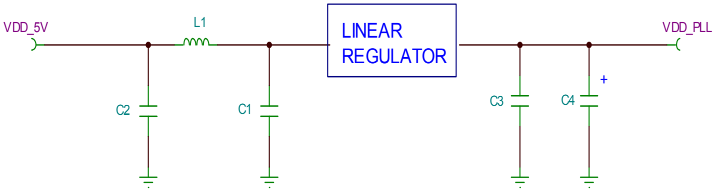


Figure 6.14. Linear Regulator Pi Filter Circuit

6.4 SUMMARY

Because of low power consumption and fast response time, most of the PLL designs integrated in the DSP today are based on digital PLL concepts. As discussed, DPLL is very sensitive to power supply and input noise, so proper design noise isolation filters are required to achieve the lowest jitter possible. The best approach is using a combination of Pi filter and linear regulator as shown in Section 6.3.2. This may not be possible due to PCB space limitation so designers have to make design compromises. If there is not enough room for the regulator circuit, then implementing a Pi filter using a resistor instead of a ferrite bead is the second best approach. This has low frequency and high frequency filtering characteristics as demonstrated in Section 6.3.1.

REFERENCES

[1] Texas Instruments Inc (2008) SM320C6455-EP Fixed-Point Digital Digital Signal Processor. SPRS462B. http://focus.it.com/lit/ds/symlink/sm320c6455-ep.pdf.

[2] Cypress Semiconductor Corporation (1997) Jitter in PLL-Based Systems: Causes, Effects and Solutions.

[3] Wavecrest (2002) Examining Clock Signals And Measuring Jitter with the WAVECREST SIA-300. Application Note No. 142.

[4] Agilent Technologies (2003) Jitter Generation and Jitter Measurements with the Agilent 81134A Pulse Pattern Generator & 54855A Infiniium Oscilloscope.

[5] Lin J, Haroun B, Foo T, Wang J, Helmick B, Mayhugh T, Barr C, Kirkpatrick J (2004) A PVT Tolerant 0.18MHz to 600MHz Self-Calibrated Digital PLL in 90nm CMOS Process. ISSCC.