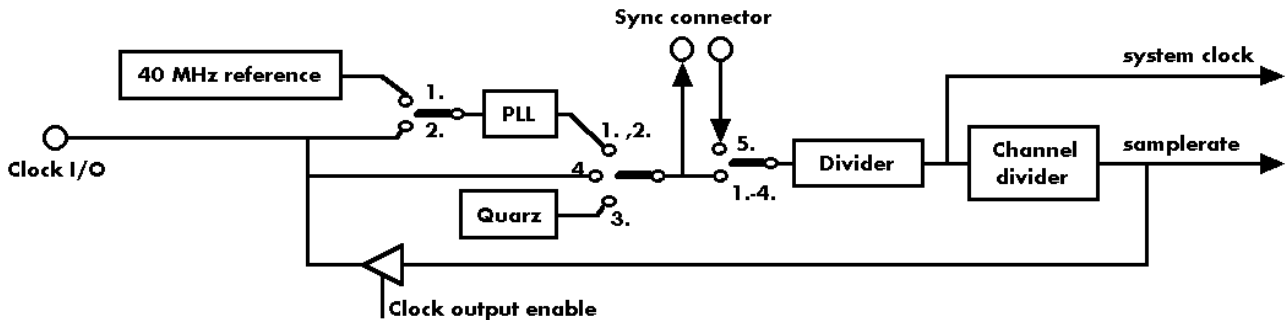




Samplerate generation on Ml.xxxx boards

Overview



The objective of this paper is to explain the different methods that exist to generate the A/D, D/A or Digital I/O sample rate. The discussion is limited to the Ml.xxxx boards, as these are of the newer design and share a common PCI motherboard onto which the various analog and digital interface modules are connected.

The sample rate can be generated onboard in five different ways:

1. PLL with internal 40 MHz reference. This is the easiest way to generate a samplerate with no need for additional external clock signals. The samplerate has a fine resolution.
2. PLL with external 2 MHz to 125 MHz reference clock. This provides a very good clock accuracy if a stable external reference clock is used.
3. Internal quartz clock with divider. For applications that need a lower clock jitter than the PLL produces. The possible samplerates are restricted to the values of the divider list as seen in the table on the right.
4. External clock. Any clock can be fed in that matches the specification of the board. The external clock signal can be used to synchronise the board on a system clock or to feed in an exact matching samplerate.
5. Using the clock synchronisation (only if the Cascaded Synchronization **or** Star Hub Synchronization option is installed). One board generates clock for all boards. The slave board(s) will get the clock from the sync connector, the master board could generate clock in any of the above four ways.

PLL

A PLL (Phase Locked Loop) is an electronic device that produces an extremely wide variety of output clocks by programmed multiplying and dividing of a fixed input clock.

possible divider values			
1	16	100	1000
2	20	200	2000
4	40	400	
8	50	500	
10	80	800	

Channel divider

The channel divider divides the system clock by the number of activated channels. The analogue and digital channels are always running with the divided clock. The memory is running with the system clock.

How to program the different modes

A detailed description of the software registers and the different modes can be found in the hardware manual. In particular the different external ranges are explained there.

mode	software register				
	SAMPLERATE	PLLENABLE	EXTERNALCLOCK	REFERENCECLOCK	EXTERNRANGE
1. PLL / internal ref.	needed samplerate	1	0	0	not used
2. PLL / external ref.	needed samplerate	1	0	external reference clock	not used
3. Quartz + Divider	needed samplerate	0	0	not used	not used
4. External clock	not used	not used	1	not used	range of external clock
5. Sync Master	same setup as mode 1. to 4.				
6. Sync Slave	same setup as sync master because driver needs to calculate programming of divider				

What the driver does

The driver performs the samplerate calculation and checks for any borders that may occur. It finds the nearest samplerate that is a match to the samplerate that the board hardware is capable of producing. After calculation, the samplerate that will be actually be generated by the board can be read out from the driver. Finally, the driver instructs the board of the sampling rate to be generated. To perform the calculation correctly the driver needs to know all of the above mentioned software registers and also the channel enable register (SPC_CHENABLE).

Number of channels for calculating samplerate

The number of channels that are used in the calculation of the samplerate depends on the number of interface modules on the board and the mode of operation. In standard mode both hardware modules are running synchronously with the same system clock. Therefore the channel divider is only programmed to the number of channels on one module. In FIFO mode both modules are running with half the system clock. The channel divider is programmed to the complete number of activated channels. Detailed information on the channels and hardware modules can be found in manual.

How the driver calculates possible samplerates with PLL use

The PLL generates any system clock between 1 MS/s and 125 MS/s. To obtain slower system clocks the additional internal divider is used. See the table above for possible divider values. The actual samplerate is then the result of the system clock being divided by the channel divider.

It is not necessary to do this calculation by yourself, the driver performs this internally every time a samplerate is programmed to it. The description above is only an explanation of what is happening inside the driver.

The PLL generates the system clock by multiplying (factor F) and dividing (factor R) the reference clock:

$$[\text{system clock}] = [\text{reference clock}] * (F+2) / (R+2)$$

with the limitation of: $[\text{reference clock}] / (R+2) \geq 300 \text{ kHz}$

where F and R could be in the range from 0 to 127. The reference clock is either the internal 40 MHz reference or the external fed in reference clock.

Example (internal reference of 40 MHz)

4 channels, needed samplerate 22.6 MS/s

system clock = 22.6 MS/s * 4 channels = 94.4 MS/s

The PLL is programmed to 94.4 MS/s (F = 57 and R = 23)

To program a larger range of samplerates it is also possible to use the PLL to generate a multiple of the samplerate and to divide it afterwards.

Example (internal reference of 40 MHz)

2 channels, needed samplerate 22.25 MS/s

system clock = 22.25 MS/s * 2 channels = 44.5 MS/s

44.5 MS/s could not be generated by the PLL, but 89 MS/s could !

So the PLL is programmed to 89 MS/s (F = 87 and R = 38) and the divider is programmed to 2.

Maximum stepsize of samplerate with internal reference

Due to of the internal structure of the PLL, the stepsize of the samplerate is depending on the samplerate itself. For higher samplerates the stepsize is larger and for lower samplerates it is smaller. To get an idea what samplerates are possible here is a table showing the maximum stepsize in some areas of samplerates. This table is only valid for internal reference clock.

This table only shows a very simple overview of stepsizes for different samplerate areas. It is only to examine what samplerates are available without calculating the formulas. To examine whether a samplerate is possible: first look in the table. If it's in the stepsize of that area → samplerate is possible. If it's not in the stepsize → you need to calculate it by the above formula.

system clock	max. stepsize
125 MS/s – 64 MS/s	1 MS/s
64 MS/s – 26 MS/s	500 kS/s
26 MS/s – 13 MS/s	200 kS/s
13 MS/s – 6.4 MS/s	100 kS/s
6.4 MS/s – 2.6 MS/s	50 kS/s
2.6 MS/s – 1.3 MS/s	20 kS/s
1.3 MS/s – 640 kS/s	10 kS/s
640 kS/s – 260 kS/s	5 kS/s
260 kS/s – 130 kS/s	2 kS/s
< 130 kS/s	1 kS/s