It is important to note that there is a required PLL lock time for the SerDes of 1uS + 200 Refclk cycles. Additionally, every time the CFGTX and CFGRX RATE and BUSWIDTH fields change, there is a time period equal the 400/PLLMPY that is needed for the byteclocks to stabilize. This means that you must insert a wait period equal or greater than the total lock time period before writing other MMR.

Here are the suggest registers to be programmed for initialization and the sequence. You'll have to setup the other registers too, for example interrupts and messaging, but they aren't needed for the initialization to work. I'd suggest trying to use these register values as is, and maybe just leave the BASE_ID and DEVICEID_REG1 to the default values, until you get everything running without errors.

- Peripheral is configured for four serial ports, 1X mode.
- SerDes reference clock is 125Mhz
- Serial RapidIO link rate is 2.5Gbps
- SerDes transmitter set for AC coupling common mode, 750mVdiffpp swing, ENFTP=1 (low skew mode), 10b interface, full-rate mode, enabled
- SerDes receiver set for full adaptive EQ, Comma alignment, AC coupling termination, 10b interface, half-rate mode, enabled
- 333Mhz VBUS for timer prescalars
- 34b address support
- Packet forwarding ID ranges are essentially disabled by setting the forwarding range to 0x00 or 0x0000. This value is reserved for the system host and will never be used for an endpoint BASE_ID.
- Read a physical layer register, such as the SP_IP_MODE, before writing the PER_SET_CNTL register with the boot_complete. See note below.
- Read to RIO_SPn_ERR_STAT to check port OK bit and output/input error stopped states before final write to RIO_PCR. Performing the register writes below that are highlighted in blue, can be done after the port_ok bit is set regardless of whether or not it is currently in input/output error stopped state.

Offset	Register	Value
0x120	Rio_serdes_cfg0_cntl	0x0000000B
0x100	Rio_serdes_cfgrx0_cntl	0x00081101
0x104	Rio_serdes_cfgrx1_cntl	0x00081101
0x108	Rio_serdes_cfgrx2_cntl	0x00081101
0x10C	Rio_serdes_cfgrx3_cntl	0x00081101
0x110	Rio_serdes_cfgtx0_cntl	0x00010901
0x114	Rio_serdes_cfgtx1_cntl	0x00010901
0x118	Rio_serdes_cfgtx2_cntl	0x00010901
0x11C	Rio_serdes_cfgtx3_cntl	0x00010901
INSERT	PLL LOCK TIME	WAIT LOOP

0x1010	**Rio_pe_feat**	**0x20000019**
0x1018	**Rio_src_op**	**0x0000fdf4**
0x101C	**Rio_dest_op**	**0x0000fc04**
0x1100	**Rio_sp_mb_head**	**0x10000002**
0x115C	**Rio_sp0_ctl**	**0x00600001**
0x117C	**Rio_sp1_ctl**	**0x00600001**
0x119C	**Rio_sp2_ctl**	**0x00600001**
0x11BC	**Rio_sp3_ctl**	**0x00600001**
0x1060	Rio_base_id	Application specific
0x12004	**Rio_sp_ip_mode**	**0x4C000000**
0x80	Rio_deviceid_reg1	Application specific
0x90	Rio_pf_16b_cntl0	0x00000000
0x94	Rio_pf8b_cntl0	0x00000000
0x98	Rio_pf_16b_cntl1	0x00000000
0x9C	Rio_pf_8b_cntl1	0x00000000
0xA0	Rio_pf_16b_cntl2	0x00000000
0xA4	Rio_pf_8b_cntl2	0x00000000
0xA8	Rio_pf_16b_cntl3	0x00000000
0xAC	Rio_pf_8b_cntl3	0x00000000
0x12008	Rio_ip_prescal	0x00000021
0x113C	Rio_sp_gen_ctl	0x00000000
0x20	Rio_per_set_cntl	0x05053750
Wait for Port_ok	Can use the next 4 lines	To exit input/output error stopped
0x14014	Rio_sp0_cs_tx	0x40FC8000
0x14114	Rio_sp1_cs_tx	0x40FC8000
0x14214	Rio_sp2_cs_tx	0x40FC8000
0x14314	Rio_sp3_cs_tx	0x40FC8000
0x4	Rio_pcr	0x00000005

Note – Bold registers must be set before boot_complete is asserted in the RIO_PER_SET_CNTL MMR. After boot_complete is asserted these are read only registers. In addition, these registers sit on a different clock domain than the RIO_PER_SET_CNTL, which means a race condition can occur. This can easily be solved by performing the writes early in the boot process, while writing the RIO_PER_SET_CNTL at the end. Alternatively, a read to MAC layer registers can be done prior to the write to the RIO_PER_SET_CNTL.