Chip ID should be only used to confirm if the I2C slave is active when the board first power on , You’d better to reset the chip after you got the correct chip ID value .

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0xEE	0x01	write	Enable I2C control
0x86	0xff	0xE1	write	write bank to 0xE1
0x86	0x00	0x22	read	chip ID register high byte
0x86	0x01	0x04	read	chip ID register low byte
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0xEE	0x00	write	Disable I2C control

The below registers are for the signal information with INT notification , the firmware must be flashed in advance .(These register don’t need to enable I2C control using 0xE0EE)

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0x80	/	read	chip name register
0x86	0x81	/	read	chip name register
0x86	0x82	/	read	Version High Byte
0x86	0x83	/	read	Version Low Byte
0x86	0x84	/	read	video interrupt type : Bit0:video ready Bit1:audio ready Bit2:HDRStreamEn

Resolution & Clock (Valid at 0xE084 bit0=1&bit2=0)

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0x85	/	read	PixelClock[23:16] KHZ
0x86	0x86	/	read	PixelClock[15:8] KHZ
0x86	0x87	/	read	PixelClock[7:0] KHZ
0x86	0x88	/	read	Htotal[15:8]
0x86	0x89	/	read	Htotal[7:0]
0x86	0x8A	/	read	Vtotal[15:8]
0x86	0x8B	/	read	Vtotal[7:0]
0x86	0x8C	/	read	Hactive[15:8]
0x86	0x8D	/	read	Hactive[7:0]
0x86	0x8E	/	read	Vactive[15:8]
0x86	0x8F	/	read	Vactive[7:0]
0x86	0x92	/	read	ByteClock[23:16] KHZ
0x86	0x93	/	read	ByteClock [15:8] KHZ
0x86	0x94	/	read	ByteClock [7:0] KHZ
0x86	0xA0	/	read	port number : 1,2,4,
0x86	0xA1	/	read	mipi format: 0x00:YUV422 8bit 0x01:YUV422 10bit

				0x02:RGB 8bit 0x03:Legacy YUV420
0x86	0x95	/	read	DP format: 0x00:RGB 0x01:YUV422 0x02:YUV444 0x03:YUV420
0x86	0x96	/	read	DP Color Depth: 0x00:6bit 0x01:8bit 0x02:10bit 0x03:12bit

Audio Sample Rate (Valid at 0xE084 bit1=1&bit2=0)

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0x90	/	read	Audio_FS_Value[16:8]
0x86	0x91	/	read	Audio_FS_Value [7:0]
0x86	0xA2	/	read	Audio Channel:8ch,6ch,4ch,2ch

MIPI stream enable (Valid when firmware supports this feature)

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0xB0	0x00	write	Disable MIPI Stream
		0x01	write	Enable MIPI Stream

DP HDR Information Frame(Valid at 0xE084 bit2=1)

Chip Address	Register Address	Value	R/W	Comment
0x86	0xff	0xE0	write	write bank to 0xE0
0x86	0x86	/	read	HDR_HB0
0x86	0x87	/	read	HDR_HB1
0x86	0x88	/	read	HDR_HB2
0x86	0x89	/	read	HDR_HB3
0x86	0x8A	/	read	HDR_DE0[7:0]
0x86	0x8B	/	read	HDR_DE0[15:8]
0x86	0x8C	/	read	HDR_DE0[23:16]
0x86	0x8D	/	read	HDR_DE0[31:24]
0x86	0x8E	/	read	HDR_DE1[7:0]
0x86	0x8F	/	read	HDR_DE1[15:8]
0x86	0x90	/	read	HDR_DE1[23:16]
0x86	0x91	/	read	HDR_DE1[31:24]
0x86	0x92	/	read	HDR_DE2[7:0]
0x86	0x93	/	read	HDR_DE2[15:8]
0x86	0x94	/	read	HDR_DE2[23:16]

0x86	0x95	/	read	HDR_DE2[31:24]
0x86	0x96	/	read	HDR_DE3[7:0]
0x86	0x97	/	read	HDR_DE3[15:8]
0x86	0x98	/	read	HDR_DE3[23:16]
0x86	0x99	/	read	HDR_DE3[31:24]
0x86	0x9A	/	read	HDR_DE4[7:0]
0x86	0x9B	/	read	HDR_DE4[15:8]
0x86	0x9C	/	read	HDR_DE4[23:16]
0x86	0x9D	/	read	HDR_DE4[31:24]
0x86	0x9E	/	read	HDR_DE5[7:0]
0x86	0x9F	/	read	HDR_DE5[15:8]
0x86	0xA0	/	read	HDR_DE5[23:16]
0x86	0xA1	/	read	HDR_DE5[31:24]
0x86	0xA2	/	read	HDR_DE6[7:0]
0x86	0xA3	/	read	HDR_DE6[15:8]
0x86	0xA4	/	read	HDR_DE6[23:16]
0x86	0xA5	/	read	HDR_DE6[31:24]
0x86	0xA6	/	read	HDR_DE7[7:0]
0x86	0xA7	/	read	HDR_DE7[15:8]
0x86	0xA8	/	read	HDR_DE7[23:16]
0x86	0xA9	/	read	HDR_DE7[31:24]

Note:

1. INT is used as the interrupt signal(50ms low level) to inform SOC start reading registers from LT7911UXE;
2. Support 100k/400k I2C;
3. Fps = PixelClock/(Htotal*Vtotal);
4. Audio_FS_Value: audio samling frequency;
5. Mipi clock rate = ByteClock*4;   Mipi data rate = ByteClock*8;