

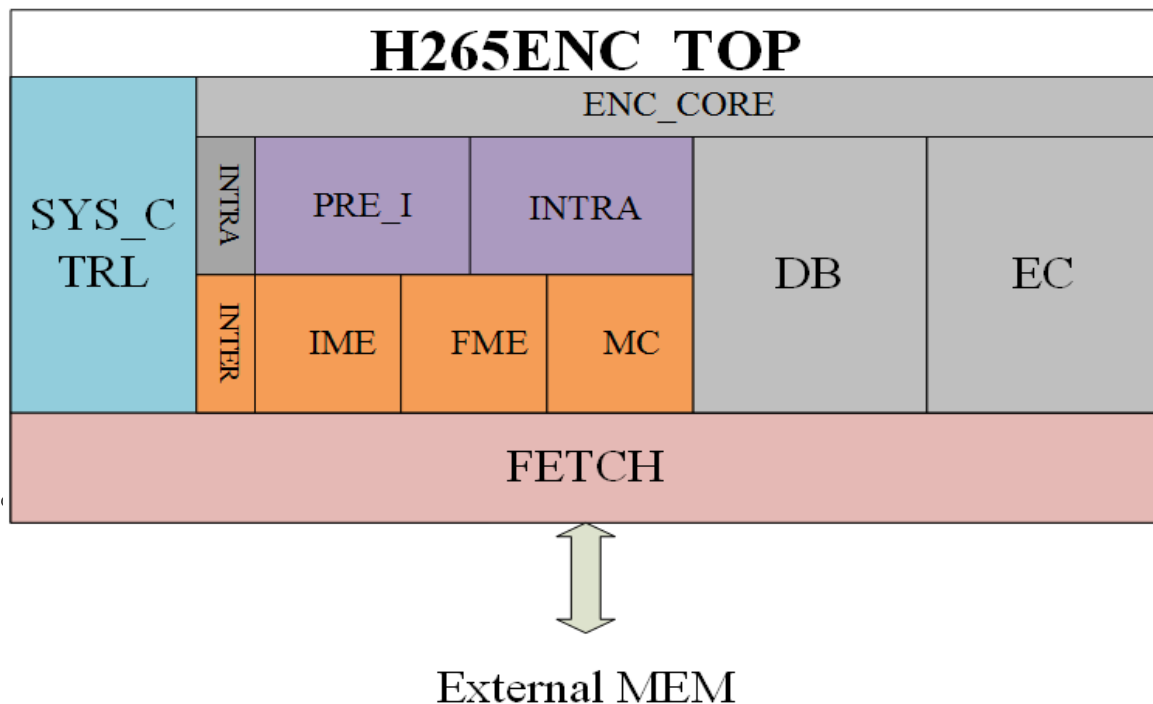
3.1 系统架构与模块仿真文件

作者：江亲炜

日期：2017/1/8

H265主要的模块：

1. 顶层模块H265ENC_top包括sys_ctrl, enc_core及fetch三个模块。
2. sys_ctrl就是一个状态机，控制fetch和enc_core中各子模块的工作。
3. enc_core 编码器的核心。
4. 存取cur_pixel和ref_pixel。

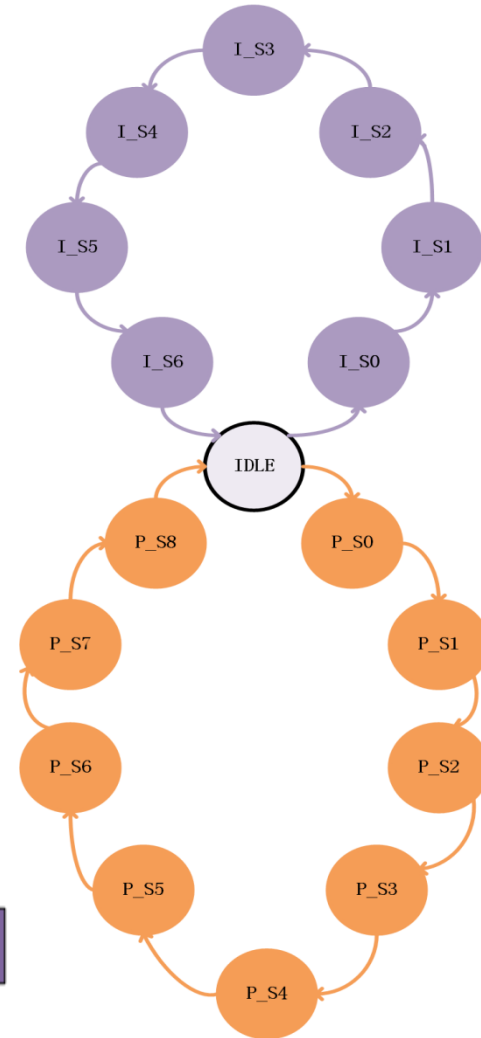


系统的状态机

Intra 编码时的状态：

	State	Description	State	Description
IDLE	IDLE	空闲状态		
fetch_ena	I_S0	Fetch load数据	P_S0	Fetch load数据
prei_ena	I_S1	Pre_intra模块on	P_S1	lme模块on
intra_ena	I_S2	Intra 模块 on	P_S2	Fme 模块 on
db_ena&ec_ena	I_S3	DB&EC 模块 on	P_S3	Mc模块 on
	I_S4	Fetch 模块off	P_S4	DB&EC 模块 on
fetch_ena	I_S5	Pre_intra模块off	P_S5	Fetch 模块off
ime_ena	I_S6	Intra 模块 off	P_S6	lme模块off
fme_ena			P_S7	Fme 模块 off
mc_ena			P_S8	Mc模块 off
db_ena&ec_ena				

Inter 编码时的



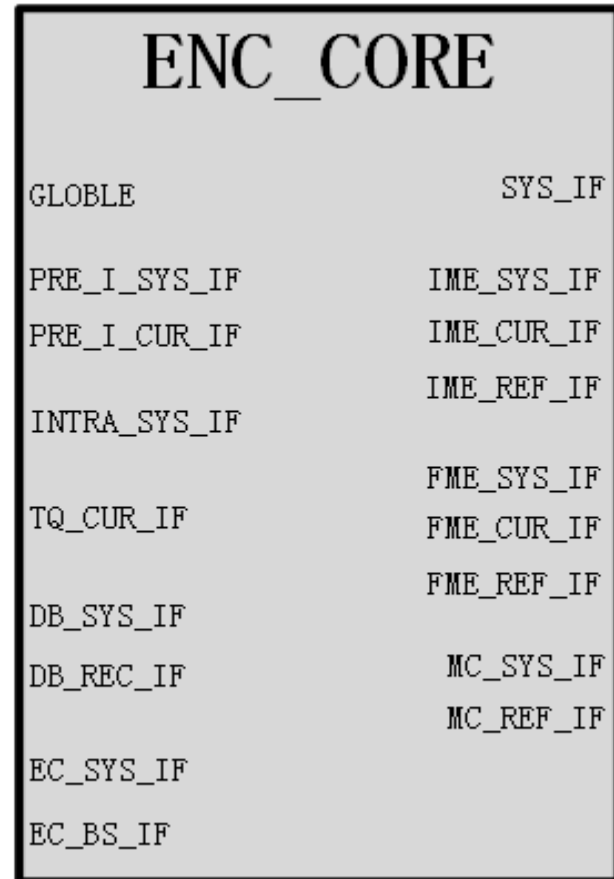
Enc_core 模块与其他模块的接口：

1. Sys_if接口为控制信号，与SYS_CTRL模块通信。以INTRA_SYS_IF为例，包括信号

Intra_start_i,
Intra_x_i,
Intra_y_i,
Intra_qp_i,
Intra_done_o

2. 其他接口为数据信号，与FETCH模块通信。以IME_CUR_IF为例,包括信号

lme_cur_4x4_x_o,
lme_cur_4x4_y_o,
lme_cur_idx_o,
lme_cur_sel_o,
lme_cur_size_o,
lme_cur_ren_o
lme_cur_data_i,



模块接口——ENC_CORE内部各子模块接口

Pre_intra顶层模块的接口

```
13 module hevc_md_top(  
14     clk,  
15     rstn,  
16     md_ren_o,  
17     md_sel_o,  
18     md_size_o,  
19     md_4x4_x_o,  
20     md_4x4_y_o,  
21     md_idx_o,  
22     md_data_i,  
23     md_we,  
24     md_waddr,  
25     md_wdata,  
26     enable,  
27     finish  
28 );
```

global

original pixel data
read

从fetch取
cur_pixel

mode ram
将pre_intra得到
的mode写入ram
由intra读取

sys_ctrl_if

Intra顶层模块的接口

```
52 module intra_top(  
53     clk  
54     rst_n  
55     pre_min_size_i  
56     uv_partition_i  
57     mb_x_total_i  
58     mb_x_i  
59     mb_y_i  
60     start_i  
61     done_o  
62     md_rden_o  
63     md_raddr_o  
64     md_rdata_i  
65     pre_en_o  
66     pre_sel_o  
67     pre_size_o  
68     pre_4x4_x_o  
69     pre_4x4_y_o  
70     pre_data_o  
71     pre_mode_o  
72     rec_val_i  
73     rec_idx_i  
74     rec_data_i  
75     cover_valid_i  
76     cover_value_i  
77 );
```

sys_ctrl_if

mode ram

tq_pred_if

Intra模块预测的
数据交给tq模块
tq /itq 得到重建
像素

tq_rec_if

重建数据写回，
作为相邻块的
ref_data

模块接口 — ENC_CORE内部各子模块接口

Ime顶层模块的接口

```
56 module ime_top (  
57     clk  
58     rstn  
59     sysif_cmb_x_i  
60     sysif_cmb_y_i  
61     sysif_qp_i  
62     sysif_start_i  
63     sysif_done_o  
64     curif_en_o  
65     curif_num_o  
66     curif_data_i  
67     fetchif_ref_x_o  
68     fetchif_ref_y_o  
69     fetchif_load_o  
70     fetchif_data_i  
71     fmeif_partition_o  
72     fmeif_cu_num_o  
73     fmeif_mv_o  
74     fmeif_en_o  
75 );
```

sysif_* (lines 59-63) → **sys_ctrl_if**

curif_* (lines 64-66) → **cur_if**
原始像素

fetchif_* (lines 67-70) → **ref_if**
前一帧的参考像素

fmeif_* (lines 71-74) → **fme_if**
Partition和imv

Fme顶层模块的接口

```
24 module fme_top (  
25     clk  
26     rstn  
27     sysif_cmb_x_i  
28     sysif_cmb_y_i  
29     sysif_qp_i  
30     sysif_start_i  
31     sysif_done_o  
32     fimeif_partition_i  
33     fimeif_mv_rden_o  
34     fimeif_mv_rdaddr_o  
35     fimeif_mv_data_i  
36     cur_rden_o  
37     cur_4x4_idx_o  
38     cur_4x4_x_o  
39     cur_4x4_y_o  
40     cur_pel_i  
41     ref_rden_o  
42     ref_idx_x_o  
43     ref_idx_y_o  
44     ref_pel_i  
45     mcif_mv_rden_o  
46     mcif_mv_rdaddr_o  
47     mcif_mv_data_i  
48     mcif_mv_wren_o  
49     mcif_mv_wraddr_o  
50     mcif_mv_data_o  
51     mcif_pre_pixel_o  
52     mcif_pre_wren_o  
53     mcif_pre_addr_o  
54 );  
55
```

sysif_* (lines 27-31) → **sys_ctrl_if**

fimeif_* (lines 32-35) → **ime_if**

cur_* (lines 36-39) → **cur_if**

ref_* (lines 40-44) → **ref_if**

mcif_* (lines 45-53) → **mc_if**
最终的mv和luma分量的预测像素

模块接口 — ENC_CORE内部各子模块接口

Mc顶层模块的接口

```

31 module mc_top (
32     clk
33     rstn
34     mb_x_total_i
35     mb_y_total_i
36     sysif_cmb_x_i
37     sysif_cmb_y_i
38     sysif_qp_i
39     sysif_start_i
40     sysif_done_o
41     fetchif_rden_o
42     fetchif_idx_x_o
43     fetchif_idx_y_o
44     fetchif_sel_o
45     fetchif_pel_i
46     fmeif_partition_i
47     fmeif_mv_i
48     fmeif_mv_rden_o
49     fmeif_mv_rdaddr_o
50     pred_wrddata_o
51     pred_wren_o
52     pred_wraddr_o
53     pred_ren_o
54     pred_size_o
55     pred_4x4_x_o
56     pred_4x4_y_o
57     pred_4x4_idx_o
58     pred_rdata_i
59     mvd_wen_o
60     mvd_waddr_o
61     mvd_wdata_o
62     pre_start_o
63     pre_en_o
64     pre_sel_o
65     pre_size_o
66     pre_4x4_x_o
67     pre_4x4_y_o
68     pre_data_o
69     rec_val_i
70     rec_idx_i
71 );

```

sysif_cmb_x_i
 sysif_cmb_y_i
 sysif_qp_i
 sysif_start_i
 sysif_done_o
 sys_ctrl_if

fetchif_rden_o
 fetchif_idx_x_o
 fetchif_idx_y_o
 fetchif_sel_o
 fetchif_pel_i
 ref_if x/y position of
 ref lcu in SW
 (U/V)

fmeif_partition_i
 fmeif_mv_i
 fmeif_mv_rden_o
 fmeif_mv_rdaddr_o
 fme_if

pred_wrddata_o
 pred_wren_o
 pred_wraddr_o
 pred_ren_o
 pred_size_o
 pred_4x4_x_o
 pred_4x4_y_o
 pred_4x4_idx_o
 pred_rdata_i
 Pred_wr_if Chroma predicted pixel out
Pred_rd_if Luma/Chroma predicted pixel in

mvd_wen_o
 mvd_waddr_o
 mvd_wdata_o
 ec_if

pre_start_o
 pre_en_o
 pre_sel_o
 pre_size_o
 pre_4x4_x_o
 pre_4x4_y_o
 pre_data_o
 rec_val_i
 rec_idx_i
 Tq_pre_if
Tq_rec_if

pre_data_o [16*pixel_width-1 :0]
 4x4 tq_data

模块接口 — ENC_CORE内部各子模块接口

deblocking顶层模块的接口

19	module db_top(42	mb_db_en_o	,	
20	clk	43	mb_db_rw_o	,	
21	rst_n	44	mb_db_addr_o	,	
22	mb_x_total_i	45	mb_db_data_o	,	
23	mb_y_total_i	46	db_wen_o	,	
24	mb_x_i	47	db_w4x4_x_o	,	Write out pixel
25	mb_y_i	48	db_w4x4_y_o	,	
26	qp_i	49	db_wprevious_o	,	1:previous 0:current
27	start_i	50	db_wdone_o	,	
28	done_o	51	db_wsel_o	,	
29	mb_type_i	52	db_wdata_o	,	
30	mb_partition_i	53	mb_db_ren_o	,	Read the top pixel
31	mb_p_pu_mode_i	54	mb_db_r4x4_o	,	
32	mb_cbf_i	55	mb_db_ridx_o	,	
33	mb_cbf_u_i	56	mb_db_data_i	,	
34	mb_cbf_v_i	57			
35	mb_mv_ren_o	58)		
36	mb_mv_raddr_o);		
37	mb_mv_rdata_i				
38	tq_ren_o				
39	tq_raddr_o				
40	tq_rdata_i				
41	tq_ori_data_i				

Diagram annotations:

- Intra/inter if (lines 32-34)
- Mv ram if (lines 35-37)
- Tq if (lines 38-41)
- Write out pixel (lines 24-25)
- 1:previous, 0:current (lines 26-27)
- Read the top pixel (lines 30-31)

模块接口 — ENC_CORE内部各子模块接口

Cabac顶层模块的接口

```

20 module cabac_top(
21     clk
22     rst_n
23     mb_type_i
24     mb_x_total_i
25     mb_y_total_i
26     mb_x_i
27     mb_y_i
28     qp_i
29     param_qp_i
30     start_i
31     sao_i
32     luma_mode_i
33     chroma_mode_i
34     mb_p_pu_mode_i
35     merge_flag_i
36     merge_idx_i
37     mb_partition_i
38     cu_skip_flag_i
39     tq_cbf_luma_i
40     tq_cbf_cb_i
41     tq_cbf_cr_i
42     cu_luma_mode_ren_o
43     cu_luma_mode_raddr_o
44     cu_chroma_mode_ren_o
45     cu_chroma_mode_raddr_o
46     mb_mvd_rdata_i
47     tq_rdata_i
48     mb_mvd_ren_o
49     mb_mvd_raddr_o
50     coeff_type_o
51     tq_ren_o
52     tq_raddr_o
53     bs_val_o
54     bs_data_o
55     bs_wait_i
56     done_o
57     slice_done_o
58 );
59
60     Coeff data
61
62     bs buffer full
63     cabac done
64     slice done } Bs if

```

模块仿真文件

Tb文件格式：

```

`define DUMP_FSDB
`define XXX_AUTO_CHECK
parameter XXX_CHECK_FILE_I //仿真的输入文件
parameter XXX_CHECK_FILE_O //仿真的比对文件
Module xxx_tb
dut_xxx
initial begin
    控制仿真的过程块
End
Task load; //load 输入数据的任务块
Task run; //对设计文件请求和应答 ( start和done信号 )
`ifdef XXX_AUTO_CHECK
    .....// 设计输出与CHECK_FILE_O数据比对
`endif
//DISPLY
//DUMP_FSDB
endmodule
    
```

模块仿真测试文件-top_module和各子模块

```

Top_module
`define TEST_I
`define TEST_P //I帧和P帧是分开单独仿真的
./tv/cur_mb_p32.dat
./tv/bs_i(p)_check_o.dat
其中cur_mb_p32.dat 数据格式是
QLCU0_luma //每行32pixel
QLCU1_luma //    32pixel
QLCU2_luma //    32pixel
QLCU3_luma //    32pixel
QLCU0_chroma//cbcrcbcr .....cbcrcbcr 32pixel
QLCU1_chroma//cbcrcbcr .....cbcrcbcr 32pixel
QLCU2_chroma//cbcrcbcr .....cbcrcbcr 32pixel
QLCU3_chroma//cbcrcbcr .....cbcrcbcr 32pixel
tv/bs_i(p)_check_o.dat 数据为编码码流
    
```

模块仿真测试文件-top_module和各子模块

Intra模块：

./tv/intra_mode.dat

./tv/intra_mode_uv.dat

./tv/intra_mb_position.dat

./tv/intra_md_decision.dat

./tv/intra_rec4x4(8x8,16x16,32x32)(_u ,_v).dat

./tv/intra_pred4x4(8x8,16x16,32x32)(_u ,_v).dat

Intra_mode : pre_intra输出的方向模式，

pre_min_size=='d0 num=85 in a LCU

pre_min_size=='d0 num=21 in a LCU

Intra_mode_uv: 用于UV预测的方向模式。

Intra_mb_position: mb_x mb_y mb_x_total, mb_x mb_y mb_x_total,.....

Intra_md_decision: rec_cover

Intra_rec: 经rec_loop重建的数据

Intra_pred:设计文件输出比对的数据。

模块仿真测试文件-top_module和各子模块

lme模块：

```

./tv/ime_check_i.dat          /*
./tv/ime_check_o.dat          Line 64
其中ime_check_i.dat 数据格式 Line 96          Lcu0
//lcu 0                       Line 64
LCU_luma // 每行 64pixel      Line 96          Lcu1
Ref_luma // 每行 (sw+LCU_size) pixel Line 64
lme_check_o.dat 数据格式     Line 96          Lcu2
//imv data                    ...
mv8x4_x,mv8x4_y //按zig_zag顺序 lcu中 ...
所有8x4块的imv(x,y)         */
//partition info
For loop                       Case -1: "11"
lme cu16_mode                  Case 0: "00"
For loop                       Case 1: "01"
lme cu32_mode                  Case 2: "10"
                               Case 15: "xx"
lme cu64_mode                  Default : "E!"
    
```

模块仿真测试文件-top_module和各子模块

Fme模块：

./tv/fme_input.dat

./tv/fme_check.dat

其中fme_input.dat 数据格式

类似ime的cur_mb , ref_mb

以及ime的输出 mv&partition info

fme_check.dat 数据为luma分量的pred_pixel

Mc模块：

./tv/mc_luma_chroma.dat //luma pred & chroma ref

./tv/mc_fmvp_part.dat

./tv/mc_check_uvpred.dat

Mc_fmvp_part.dat

Line1 partition 同ime_check

Line64 fmvp

mc_check_uvpred数据为cb/cr分量的pred_pixel

模块仿真测试文件-top_module和各子模块

Db模块：

./tv/db_input.dat

./tv/db_check.dat

./tv/db_left_check.dat

其中db_input.dat 数据格式:

```
//lcu 0
frame_mb_x_total -1
frame_mb_y_total-1
mb_x
mb_y
qp
slice_type
is_spilt(16x16,32x32,64x64)
mode(16x16,32x32,64x64, size2nx2n,2nxn,nx2n)
cbf \n cbf_u \n cbf_v
mv
mv_top
luma pixel (68x68) \n cb pixel \n cr pixel //4x4 block in a line
luma_top (4x68)\n cb_top pixel \n cr_top pixel
org pixel \n org_u \n org_v
```

db_check.dat :

y_cache[68][68]//16pixel 4x4 block in a line

u_cache[34][34]

v_cache[34][34]

db_left_check.dat

模块仿真测试文件-top_module和各子模块

Cabac模块：

./tv/cabac_input.dat

./tv/bs_check.dat

其中cabac_input.dat 数据格式:

```
/*
Ctrl_if
Sao_if (when sao_open)
Intra/inter if (split info ,mb_i_luma/chroma_mode,mb_p_luma/chroma_mode)
Tq_rdata_i (coeff info,cbf_info)
Mb_mvd_rdata_i (mvd&mvp_idx)
*/
bs_check.dat (bitstream info)
```


ASIC^o

专注开源硬件 IP Core

www.openasic.org