

main.c Code

NARLabs

```
while (!AXI_DMA_TxDone && !AXI_DMA_RxDone) {}
AXI_DMA_TxDone=0;
AXI_DMA_RxDone=0;

AXI_DMA_Transfer((UINTPTR)&Padding3, PADDSIZE, XAXIDMA_DMA_TO_DEVICE);
AXI_DMA_Transfer((UINTPTR)&Padding3_RX, PADDSIZE, XAXIDMA_DEVICE_TO_DMA);

//if 0
Xil_DCacheFlushRange((UINTPTR)&Padding3, PADDSIZE);
Xil_DCacheFlushRange((UINTPTR)&Padding3_RX, PADDSIZE);
#endif

//for verify data
for(int i=0; i<PADDSIZE; i++)
{
    if(Padding3_RX[i]!=Padding3[i])
    {
        xil_printf("Padding3 error \r\n");
    }
}

while (!AXI_DMA_TxDone && !AXI_DMA_RxDone) {}
AXI_DMA_TxDone=0;
AXI_DMA_RxDone=0;

//end Padding to TR CVTNEW add
yolo_demo(Padding1, Padding2, Padding3, FstFrame, DEMO_P_CNT);
if(DEMO_P_CNT<2)
{
    DEMO_P_CNT=DEMO_P_CNT+1;
}
else
{
    DEMO_P_CNT=0;
}
memcpy((uchar *)YOLO_BASE, FstFrame, TBSIZE);
```

Transfer data from DDR to IP ④

Transfer data from IP to DDR ⑤

Wait AXI DMA transfer complete

demo function for draw box with lable

102

Demo Function For Draw Box with Label

DEMO_P_CNT變數為座標測試樣本，定義在draw.c

```
yolo_demo(Padding1, Padding2, Padding3, FstFrame, DEMO_P_CNT);
if(DEMO_P_CNT<2)
{
    DEMO_P_CNT=DEMO_P_CNT+1;
}
else
{
    DEMO_P_CNT=0;
}
```

103

Draw.c

NARLabs

測試樣本以P2組為例，第一點左上座標（15,60），第二點右下座標（250,140）
辨識類別1：person

以此類推

```
int P1_x1[] = {10, 100, 200, 300, 400, 500, 600, 700, 800, 300, 500, 600, 100, 700, 600, 1500};
int P1_y1[] = {10, 100, 200, 300, 400, 300, 400, 700, 500, 600, 700, 80, 200, 300, 400, 500};
int P1_x2[] = {200, 700, 500, 700, 500, 900, 700, 1000, 900, 1000, 1100, 1200, 1300, 1400, 15};
int P1_y2[] = {30, 150, 250, 350, 450, 590, 690, 750, 890, 990, 1080, 157, 296, 395, 491, 59};
int P1_str[] = {0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3};

int P2_x1[] = {15, 50, 250, 350, 450, 550, 650, 750, 850, 350, 550, 650, 150, 750, 650, 1550, 1750, 1850};
int P2_y1[] = {60, 50, 250, 350, 450, 350, 450, 750, 550, 650, 750, 130, 250, 350, 450, 550, 110, 750, 650, 510};
int P2_x2[] = {250, 750, 550, 750, 550, 950, 750, 1050, 950, 1050, 1150, 1250, 1350, 1450, 1550, 1650, 1510, 1};
int P2_y2[] = {140, 200, 300, 400, 500, 640, 740, 800, 940, 1040, 1130, 207, 346, 445, 541, 644, 210, 840, 942};
int P2_str[] = {1, 2, 2, 3, 3, 0, 0, 1, 1, 2, 2, 3, 3, 0, 0, 1, 1, 2, 2};

int P3_x1[] = {50, 150, 250, 350, 450, 591, 650, 750, 850, 950, 1050, 1150, 1250, 1350, 1450, 1550, 1650, 1750};
int P3_y1[] = {50, 100, 150, 200, 250, 381, 350, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950};
int P3_x2[] = {200, 300, 500, 700, 500, 789, 700, 800, 900, 1000, 1100, 1200, 1300, 1400, 1500, 1600, 1700, 18};
int P3_y2[] = {400, 600, 800, 921, 300, 691, 400, 450, 500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000};
int P3_str[] = {3, 2, 1, 0, 3, 2, 1, 0, 0, 1, 2, 3, 0, 2, 1, 3, 0, 2, 1, 3};
```

104

Video Data in DDR4

NARLabs

SDK Log Memory

Monitors

0x10000000

VDMA write channel address

Address	0 - 3	4 - 7	8 - B	C - F
0000000010000000	C1FF96A7	30CAFF5C	FF59CAFF	CAFF9
0000000010000010	GBRG	BRGB	GBRG	F9BCF
0000000010000020	FF9CC3FF	CAFF42CA	A5CAFF5F	FFBFC
0000000010000030	CAFFC0CA	C0CAFFC0	FFC0CAFF	CAFFC
0000000010000040	C0CAFFC0	FFC0CAFF	CAFFC0CA	C0CAFI
0000000010000050	FFC0CAFF	CAFFC0CA	B9C6FFC0	FFADC
0000000010000060	BEFFB7BB	FFC3FFFF	FFFFBEFF	C3FFFI
0000000010000070	45CAFFA8	FF60CAFF	CAFFA8CA	C0CAFI
0000000010000080	FFC0CAFF	CAFFC0CA	C0CAFFC0	FFC0C
0000000010000090	CAFFC0CA	C0CAFFC0	FFC0CAFF	CAFFC
00000000100000A0	C0CAFFC0	FFC0CAFF	CAFFC0CA	C0CAFI
00000000100000B0	FFC0CAFF	CAFFC0CA	C0CAFFC0	FFC0C
00000000100000C0	CAFFC0CA	C0CAFFC0	FFC0CAFF	CAFFC
00000000100000D0	FFC0CAFF	CAFFC0CA	C0CAFFC0	FFC0C

105