#include <stdio.h>

#include <xparameters.h>

#include "xdogain.h"

#include "xaxidma.h"

// IP config pointers and handlers

XDogain doGain;

XDogain\_Config \*doGain\_cfg;

XAxiDma axiDMA;

XAxiDma\_Config \*axiDMA\_cfg;

// DMA Addresses

#define MEM\_BASE\_ADDR 0x01000000

#define TX\_BUFFER\_BASE (MEM\_BASE\_ADDR + 0X00100000)

#define RX\_BUFFER\_BASE (MEM\_BASE\_ADDR + 0X00300000)

#define SIZE\_ARR 1000

int inStreamData[SIZE\_ARR];

void initPeripherals()

{

// Initialize dogain core

printf("Initializing doGain\n");

doGain\_cfg = XDogain\_LookupConfig(XPAR\_DOGAIN\_0\_DEVICE\_ID);

if (doGain\_cfg)

{

int status = XDogain\_CfgInitialize(&doGain,doGain\_cfg);

if (status != XST\_SUCCESS)

{

printf("Error initializing doGain core\n");

}

}

// Initialize AxiDMA core

printf("Initializing AxiDMA\n");

axiDMA\_cfg = XAxiDma\_LookupConfig(XPAR\_AXIDMA\_0\_DEVICE\_ID);

if (axiDMA\_cfg)

{

int status = XAxiDma\_CfgInitialize(&axiDMA,axiDMA\_cfg);

if (status !=XST\_SUCCESS)

{

printf("Error initializing AxiDMA core\n");

}

}

XAxiDma\_IntrDisable(&axiDMA, XAXIDMA\_IRQ\_ALL\_MASK, XAXIDMA\_DEVICE\_TO\_DMA);

XAxiDma\_IntrDisable(&axiDMA, XAXIDMA\_IRQ\_ALL\_MASK, XAXIDMA\_DMA\_TO\_DEVICE);

}

int main()

{

// Pointers to DMA TX/RX addresses

int \*m\_dma\_buffer\_TX = (int\*) TX\_BUFFER\_BASE;

int \*m\_dma\_buffer\_RX = (int\*) RX\_BUFFER\_BASE;

initPeripherals();

// Do the stream calculation

for (int idx = 0; idx < SIZE\_ARR; idx++)

{

inStreamData[idx] = idx;

}

// Set gain to 5 and start core

while (1)

{

int gain;

printf("Choose gain : ");

scanf("%d",&gain);

XDogain\_Set\_gain(&doGain, gain);

XDogain\_Start(&doGain);

// Flush the cache of the buffers

Xil\_DCacheFlushRange((u32)inStreamData,SIZE\_ARR\*sizeof(int));

Xil\_DCacheFlushRange((u32)m\_dma\_buffer\_RX,SIZE\_ARR\*sizeof(int));

printf("Sending data to IP core slave\n");

XAxiDma\_SimpleTransfer(&axiDMA,(u32)inStreamData,SIZE\_ARR\*sizeof(int),XAXIDMA\_DMA\_TO\_DEVICE);

printf("Get data\n");

XAxiDma\_SimpleTransfer(&axiDMA,(u32)m\_dma\_buffer\_RX,SIZE\_ARR\*sizeof(int),XAXIDMA\_DEVICE\_TO\_DMA);

while(XAxiDma\_Busy(&axiDMA,XAXIDMA\_DEVICE\_TO\_DMA));

//Invalidate the cache to avoid reading garbage

Xil\_DCacheInvalidateRange((u32)m\_dma\_buffer\_RX,SIZE\_ARR\*sizeof(int));

while(!XDogain\_IsDone(&doGain));

printf("Calculation complete\n");

// Display data

for (int idx = 0; idx < SIZE\_ARR; idx++)

{

printf("Recv[%d]=%d\n",idx,m\_dma\_buffer\_RX[idx]);

}

}

return 0;

}