Developing Image Processing Platforms

ADAM TAYLOR – ADIUVO ENGINEERING
ADAM@ADIUVOENGINEERING.COM
How do we create this?

- MiniZed based IR Application
- Base image processing platform
- Expandable
  - WIFI image transmission
  - Advanced image processing using reVISION & SDSoc
With the correct sensor selection we can image a wide range of the EM spectrum.

Sensor Technology Types:

- **Charge Coupled Device** – X-Ray to Visible and stretching to Near Infra Red
- **CMOS Imager Sensor** – X-Ray to Visible and stretching to Near Infra Red
- **Uncooled IR** – Micro Bolometer – typically operate in MID-IR Range
- **Cooled IR** - HgCdTe or InSb based solutions require cooling

As the wavelength decreases, electron energy decreases, requiring more exotic semi conductors than silicon.
Need to be able to content with a range of inputs and outputs

**RGB** Each pixel contains Red / Green and Blue elements – Commonly 8 bit for each element = 24 bit per pixel

**YUV** Each pixel contains Luminance (Y) two Chrominance elements U (Blue-Y) V (Red-Y)

YUV 4:4:4 8 bits per element for a 24 bit pixel

YUV 4:2:2 8 bits per element – but U and V elements are shared between pixels

RAW Format 8, 10 or 12 bits raw from the sensor
Role of image processing chain

- Provides the ability to capture an image sequence from a camera or sensor, process the image and format for onwards transmission or display.

Input

Colour Space Conversion

Processing Algorithms

Video Direct Memory Access

Output

Generation

Timing Detection
What does this look like in Vivado
What does this looks like in Vivado
## Getting Started

<table>
<thead>
<tr>
<th>Video &amp; Image Processing</th>
<th>AXI4-Stream</th>
<th>Production</th>
<th>Included</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>Video In to AXI4-Stream</td>
<td>AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_vld_in_axi4s:4.0</td>
</tr>
<tr>
<td>AXI4-Stream to Video Out</td>
<td>AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_axi4s_vld_out:4.0</td>
</tr>
<tr>
<td>YCrCb to RGB Color-Space Converter</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_ycrcb2rgb:7.1</td>
</tr>
<tr>
<td>Video Test Pattern Generator</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_tpg:7.0</td>
</tr>
<tr>
<td>RGB to YCrCb Color-Space Converter</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_rgb2ycrcb:7.1</td>
</tr>
<tr>
<td>AXI Video Direct Memory Access</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:axi_vdma:6.3</td>
</tr>
<tr>
<td>Video Timing Controller</td>
<td>AXI4</td>
<td>Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_tc:6.1</td>
</tr>
<tr>
<td>Video Frame Buffer Write</td>
<td>AXI4, AXI4-Stream</td>
<td>Pre-Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_frambuf_wr:1.0</td>
</tr>
<tr>
<td>Video Frame Buffer Read</td>
<td>AXI4, AXI4-Stream</td>
<td>Pre-Production</td>
<td>Included</td>
<td>xilinx.com/ip:v_frambuf_rd:1.0</td>
</tr>
<tr>
<td>Color Correction Matrix</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_ccm:6.0</td>
</tr>
<tr>
<td>Chroma Resampler</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_chromares:4.0</td>
</tr>
<tr>
<td>Video Mixer</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_mix:1.0</td>
</tr>
<tr>
<td>Gamma Correction</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_gama:7.0</td>
</tr>
<tr>
<td>Image Enhancement</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_enhanc:8.0</td>
</tr>
<tr>
<td>Color Filter Array Interpolation</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_cfa:7.0</td>
</tr>
<tr>
<td>Video Processing Subsystem</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_proccs:2.0</td>
</tr>
<tr>
<td>Video Deinterlacer</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_deinterl:4.0</td>
</tr>
<tr>
<td>Video On Screen Display</td>
<td>AXI4, AXI4-Stream</td>
<td>Production</td>
<td>Purchase</td>
<td>xilinx.com/ip:v_osd:6.0</td>
</tr>
</tbody>
</table>
Interfacing to the Sensor / Camera

- Beauty of programmable logic is with the right PHY we can interface to any standard
- For logic level signals often no PHY is required
- 7 Series and Up IO is in two varieties, High Range and High Performance
- LVDS – Commonly used by many Image Sensors & Camera Link available on HR and HP
- TMDS – Used with HDMI commonly used by cameras and displays available on HR
- MIPI – Used with Cameras and Displays - UltraScale+ MPSoC - IBUFDS_DPHY & OBUFDS_DPHY IO primitives
Camera Link example
Some interfaces e.g. Xilinx MIPI and HDMI interface directly with AXI Stream.

For others Parallel to AXI Convertor exists
AXI Stream Detail

- Efficient movement of data in the Image processing chain
- No need for address channel
- Flow control
  - TReady – Asserted by the downstream peripheral when ready to receive data
  - TValid – Asserted by transmitting peripheral when output data is valid
- TUSER issued for start of Frame
- TLAST Issued for end of line
Clock Domains

- Two Main Clock Domains – in the simplest case, plus AXI Lite
- Pixel Clock & AXI Stream
- To minimise buffering AXI Stream Clock > Pixel Clock
Creating Flexible Clocks

- To provide maximum flexibility we need to be able to adjust the pixel clock
- If necessary can be used for AXI Stream Clock
- 20MHz to 108 MHz + commonly required
- Use fabric clocks on Zynq / Zynq MPSoC
- Clock Wizard provides AXI Lite IF
- Can be set depending upon the mode detected
Many processing cores have the ability to process 1, 2, 4 or 8 pixels per clock. (careful though not all blocks can process multiple pixels per clock)

More Increasing the number of pixels decreases latency of the image

But! Increases resources requirements

Careful trade off between the latency and resources
We may need to adapt to different format inputs – Resolution, Frame Rate etc

Being able to detect this allows the Image Processing Chain to be re-configured for current mode

Video Timing Controller in Detection Mode

Provide Hsync, Vsync & Active Video

```c
#define XVT_VMODE_720P 1 /**< Video mode 720P */
#define XVT_VMODE_1080P 2 /**< Video mode 1080P */
#define XVT_VMODE_480P 3 /**< Video mode 480P */
#define XVT_VMODE_576P 4 /**< Video mode 576P */
#define XVT_VMODE_VGA 5 /**< Video mode VGA */
#define XVT_VMODE_SVGA 6 /**< Video mode SVGA */
#define XVT_VMODE_XGA 7 /**< Video mode XGA */
#define XVT_VMODE_SXGA 8 /**< Video mode SXGA */
#define XVT_VMODE_WXGAPLUS 9 /**< Video mode WXGAPlus */
#define XVT_VMODE_WSXGAPLUS 10 /**< Video mode WSXGAPlus */
#define XVT_VMODE_1080I 100 /**< Video mode 1080I */
#define XVT_VMODE_NTSC 101 /**< Video mode NTSC */
#define XVT_VMODE_PAL 102 /**< Video mode PAL */
```
What does this look like?

SXGA video outputting 1280 pixels by 1024 lines @ 60Hz

720P video outputting 1280 pixels by 720 Lines @ 60 Hz
Colour sensors use Bayer patterns – ensure a pixel receives only one wave length of light (R,G,B)

Four pixels, organised 2 by 2 consisting of One Red, One Blue, Two Green are used to recreate image colour

Post process to reconstruct the colour of a pixel

Colour sensor reduces image / sensor resolution by 20 %
One of most common colour scheme is RGB 888 this uses 8 bits per colour, providing over 16 Million colours. (24 bits total)

YUV colour space uses Luminance and two Chrominance elements for each pixel YUV can be represented with 8 bits per element called YUV4:4:4 (24 bits total)

However YUV can share UV elements between pixels this is called YUV4:2:2

Providing 16 bits per pixel – Easing routing – more efficient memory usage
VDMA

- Moves the image to and from a frame buffer – Most often the PS memory
- Enables access to the frame to the PS
- Useful for Frame Rate changes & resolution changes
- Can be synchronised
- Up to 680 HD Frames per second
- Needs Simple SW to configure over AXI
Use the right formatting – it helps
Verifying Data in Memory

- Bare Metal application to configure the VMDA writing to memory
  - Confirm expected pixels values against a test or calibration pattern.
  - Examine the Image Histogram, enabling histogram equalisation to be implemented if necessary.
  - Ensure the integration time of the imager is set correctly for the scene type
  - Examine the quality of the image sensor identifying defective pixels which are dead, stuck at a value.
  - Determination of noise present in the image, due to both inherent imager noise source for example fixed pattern noise, device noise and dark current but also system noise as coupled in via power supplies and other sources of electrical noise
- Simple SW function can be created to write out a BMP
Verifying Data in Memory

```c
while(1){
    for (y=0;y<480;y++){
        for (i = 0; i < 600;i++){
            red[y][i]= X11_In8(addr);
            blue[y][i]= X11_In8(addr+1);
            green[y][i]= X11_In8(addr+2);
            addr = addr +3;
        }
        addr = addr + 600;
    }

    y =0;
    i = 0;

    printf("%04x%04x%04x%04x%04x%04x%04x%04x%04x%04x%04x%04x",
            Header.Type,Header.Size,Header.Reserve1,Header.Reserve2,
            Header.Offset,Header.biSize,Header.biWidth,Header.biHeight,Header.biPlanes,Header.biBitCount,
            Header.biCompression,Header.biSizeImage,Header.biXPelsPerMeter,Header.biYPelsPerMeter,
            Header.biClrUsed,Header.biClrImportant);

    for (y=0;y<480;y++){
        for (i = 0; i < 600;i++){
            printf("%02x%02x%02x",blue[y][i],green[y][i],red[y][i]);
        }
    }
    //printf("\n");
    break;
}
```
Mixing Video Channels

- Up to 7 sources
- AXI Stream or Memory Map
- Additional Logo layer
- Multiple Sensors
- Overlay Telemetry
- Picture in Picture
Mixing Video Channels

- Alpha defines transparency
Mixing Video Channels
Output Timing

- For many outputs we need to generate the Video Timing of the required output.

- Use Video Timing Controller configured as Generator.

- Can be configured over AXI Lite to adapt to different output timings on the fly.
Synchronisation with an external frame sync

Video Timing will help you do this

However Need to control the Gen CLK pin

Failure to do this results in free running of Timing Generator

Use FSync OP to disable
AXIS To Video Out

- Converts from AXI Stream to Parallel
- Parallel output Hsyncs, Vsync & pixels
- Can be used to drive VGA Directly
- Alternatively can interface to external
Ok it is not working?

- **Check Reset and Clocks** – Is the Pixel Clock correct for the video timing
- **Check the Clock Enables** – Many blocks have enables are these tied off correctly
- **Check that the VTC** – Is the VTC source registers correctly set along with configuration
- **Check the AXIS-to-Video-Out** – Is configured for master mode if using VDMA
- **Check the AXIS-to-Video-Out** - VTC_ce signal is not connected to the VTC gen clock enable
- **Ensure that HSize and Stride are correct** - Remember these are defined in Bytes not pixels
- **Insert ILA’s & JTAG to AXI Bridge** – Monitor the AXI Streaming using ILA, configure using Bridge
- **Check the connections** – Ensure Hblank, Vblank, Active Video are connected if used
Wrapping it Up

- Creating the Image Processing Chain looks daunting but is pretty straightforward.
- Additional filters & processing can be created using HLS and the xfopenCV libraries.
- Enables creation of a base platform for SDSoC and reVISION use.
Example Designs

- **MicroZed Chronicles examples**
  - [https://github.com/ATaylorCEngFIET/UltraZed_Part18](https://github.com/ATaylorCEngFIET/UltraZed_Part18) - UltraScale+ MPSoC PL to PS Example P218
  - [https://github.com/ATaylorCEngFIET/Nexys_Video_Part220](https://github.com/ATaylorCEngFIET/Nexys_Video_Part220) - Direct HDMI TMDS Decoding P220
  - [https://github.com/ATaylorCEngFIET/FLIR_LEPTON2](https://github.com/ATaylorCEngFIET/FLIR_LEPTON2) - Arty Z7 FLIR Lepton Example P203 P205

- Pynq and Object Tracking P1 and P2
- Camera Link
- Addressing VDMA issues
- HDMI Sink & Source Options
- Artix / Microblaze Video Processing P204 P206
- Video Mixing
Questions?

- Read More each Week the [www.MicroZedChronicles.com](http://www.MicroZedChronicles.com)
- Contact me at [Adam@adiuvoengineering.com](mailto:Adam@adiuvoengineering.com)