

G1-Unit Prototypes - Stackbot

04/03/2013

Group Members:

Qi Zhiyuan

Sanjay Mario Dmello

Jovita Castelino

Anitha Ganesha

Software:

Bt878 video driver

Following plug in of the Bt878 card, the BSP configuration address space was verified. By doing a PciAllHeaderShow; and using the vendorId and deviceId, the Bt878 was found to reside on Bus No. 5. and Device No. 0. The interrupt pin was found to be 0x01 and interrupt line 0x0b.

Using PciFuncBarShow(BusNo, Device No); the MMIO address of 0xfca00000.

So likewise, in the code the Btvid MMIO address and interrupt line were defined.

pciDeviceShow(0) returned the list of devices on the corresponding bus. Using the PCI List Table lookup North Bridge with deviceId 0x2570 and South Bridge with deviceId 0x244e was established and defined in the header.

Single Video Frame acquisition:

The next change was implemented in the Activate function to start 320x240 RGB video decoding. Here the input of setmux was set to 3 to allow for composite S-Video.

So for the acquisition of the first frame; in the start_video function after initialization of the PCI devices and its configuration the btvid_drvr task is spawned following which the commands are activate(), reset_status and write_to_buffer upon which the first frame in .ppm format is created.

Vpipe Display Utility:

This is Portable Pixel Map(PPM) Stream Viewer built in python used for display of uncompressed video stream.

A TCP/IP Connection is established between the host and the target for transfer of video frames at a maximum of 30fps.

TCP Server application:

Vpipe_display python utility acts as a server which listens for a client connection. Once the connection is established with the target, it waits on data to be received. the target sends one frame at a time that contains 320 * 240 * 3 bytes video frame bytes + 2 information bytes.

After all the data is received, this information along with ppm header information is written to a ppm file. This is then displayed on python tk utility. The function which waits on connection

receive, is invoked once in every 30 milliseconds.

TCP Client application -VxWorks Functions:

Frametx.c and Frametx.h files are included with the main file, which provides functions for establishing connection with the host and transferring video frame at 30fps.

A VxWorks task is spawned which first calls init_frametx() function responsible for host connection establishment. Then it later waits on a semaphore that gets released by the Bt878 video driver ISR every 30 millisecond. once the semaphore is available, it transfers the data 2 bytes at a time to the host.

Challenges: The main challenge that we faced is to stream the data at 30fps. Initially we observed that the data was transferred at a very low speed that corrupted the video display. This is due to another high priority task that was running under the target. When this task was later removed, we were able to stream the data at 30fps. Following this observation, we now need to figure out how to schedule this task for uninterrupted video display along with the other tasks such as Image processing and centroid computation.

Hardware:

We designed, built and tested a speed tunable conveyor belt that is based on Pulse Width Modulator voltage control method. This homemade conveyor belt can convey objects at the speed from 1cm/s to 15cm/s. The speed is tuned by varying the voltage that is imposed on the driving motor. The conveying speed as a function of voltage is also calibrated. We also designed, fabricated and tested a Buck converter. The purpose of the Buck converter is to provide tunable voltage for the driving motor of the convey belt. We are going to use the ATOM board to generate a 20 kHz PWM signal, which is controlled by a feedback voltage sensing loop on the Buck converter to ensure the stability of the output voltage. Finally, we finished the control algorithm as well as the pseudo code for generating of the PWM from ATOM.

Challenges: The main challenge for building and testing the conveyor belt is the mechanical balance. We solved the problem via fine thread screwing on both ends of the conveyor cylinders. The challenge for the buck converter, on the other hand, is the spike noise from the PWM signal amplification. We solve the problem by introducing a isolation transformer. The transformer can not only keep the signal in good shape but also remove high frequency noise, which will make the rotational speed of our motor more stable.