Draw Block Report

Andrew Kubal

March 20, 2016

Background

As a segment of the vector display processor, draw block takes host commands from a host processor, deciphers these commands, and provide pixel write commands, or pixel operations, to the ram control block. Draw block consists of three segments: Draw block, draw any octant, and draw octant.

Draw Block

Draw Block is the top level file for the draw block segment and it supports the host and RCB interfaces. Draw block can be decomposed into smaller portions to represent how it operates. There is a process for each portion.

Process – Octant CMB

The Octant CMB combinational process figures out which octant to draw in by comparing (x,y) inputs to the (x,y) pen coordinates. As seen in figure 1, negx, negy, and swapxy are initially set to zero, and each will change only if each fulfils a certain requirement. This simplifies the logic. The process also handles the draw, reset and disable commands.

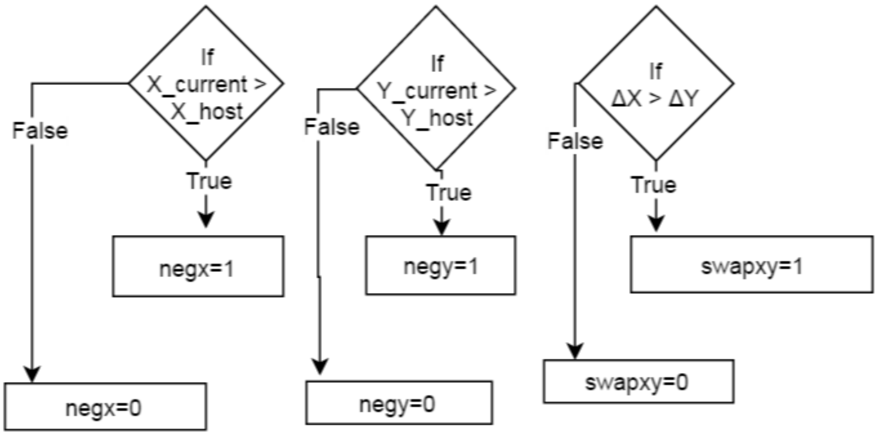

Figure 1.

Process – DB FSM

This registered process represents the finite state machine of draw block as can be seen in figure 2. The FSM assigns the proceeding state to the current state on the rising edge of the clock. In three instances, this does not occur: When the delay command is high, when reset is high, or when the Draw Any Octant state is not at pixel_state for clearscreen. The RCB can send the signal delaycmd=1 which stops the FSM from updating the current state for the single_pixel or clearscreen operation. The reset signal sends the state back to idle.

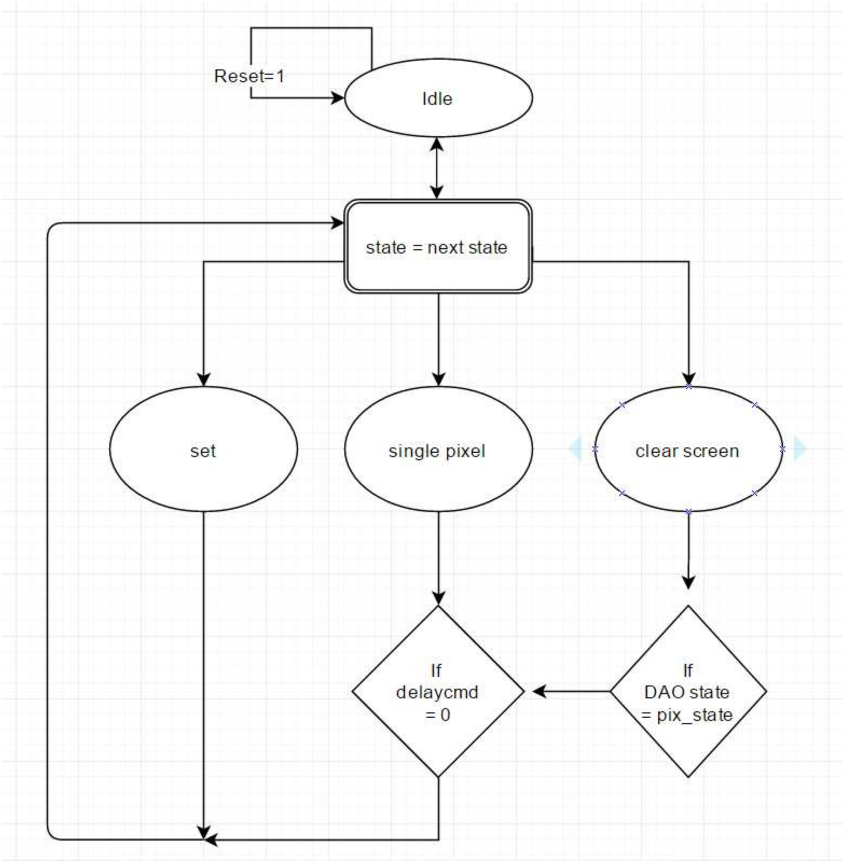

Figure 2.

Process – XY register

This combinational process updates the x and y registers

Process – hdb_busy

This combinational process defines how the hdb_busy output behaves in terms of DB FSM states. Hdb_busy is initially set high and goes low when moving or when idling. In the other cases, the delaycmd signal is passed on to hdb_busy. Draw line is the exception, and requires Draw Any Octant to be done before passing the delaycmd signal. Figure 3 displays the combinational logic.

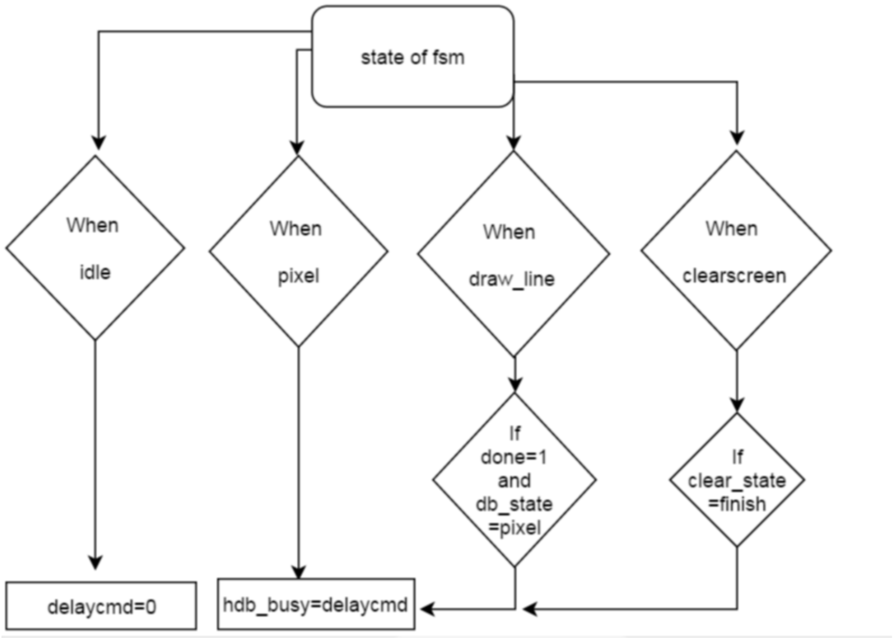

Figure 3.

Draw Any octant

This exercise will correctly draw lines in any of the eight octants, that is NNW, WNW, WSW, SSE, SSW, ENE, NNE, SSW. The octant CMB process converts the inputs and outputs to figure out where to draw the next pixel, and sends commands to draw Any Octant to do the drawing.