Note on Walsh Distribution from Control card to DPCs

APRIL 17, 1994

Alka Dikshit

1 Introduction

The walsh demodulation for different channels of the system is done on DPC cards. The demodulation requires that the DPCs are provided with a pattern according to which the demodulation will be done. The control card will get walsh pattern from rack control card, which then has to be distributed on DSP bus. If we adopt a strategy, *(which was decided earlier)* that the walsh pattern supplied will be read by DSP and then written on its bus; to be written on various DPC cards, few problems may arise. Namely:

- The DPC cards which don't require any change in their walsh states will also be written and might produce glitches in the demodulation which would be disastrous for the spectrum.
- If we try to decide which cards are to be written with new walsh within the processor and then write; The skew of walsh pattern between the individual DPC cards can exceed more then micro second level, because each machine cycle of the DSP processor *(used on control card)* is 100 ns.
- The primary action of the DSP is to take care of new delays and new controls and, distribute them to relevant card as and when required. It also has to read and accumulate total power from DPC cards and then send this accumulated total power to the rack control. A single control card will take care of all the cards in a bin. Hence the processor may not have enough time to take care of walsh distribution itself.

Hence it is decided to make a State Machine on the control card which will take care of the walsh distribution from control card to DPC cards. By doing so we will be able to decouple the walsh distribution from processor. It can be asynchronously achieved as the processor used has facility for *bus grant and Bus request* which also is asynchronous. Although this will require a minor change on DPC card (i.e. a wire strap has to be put from dspsel to the clock pin of the latch which will latch the walsh pattern.)

2 Specification of the circuit

The requirements which are to be met by this circuit are following.

- The circuit should be able to detect the change in walsh pattern at any point of time and indicate the same to the processor by asserting *bus request.*
- The circuit should be able to figure out which DPC cards are to be written i.e. to check which card's pattern changed.
- It should be able to generate all the required signals for writing on to DPCs.
- It should be able to write to those cards once the bus is granted by the processor.
- It should de-asserts the *bus request* after latching the new walsh pattern and reset itself once all the changed walsh patterns are downloaded to respective DPCs.
- It should be able to generate read, write and control signal for each DPC card which are required for walsh loading.
- It should not create bus contention on the control card while in possession of the DSP bus. It should not create bus contention on back plane when it is driving the buses there.

3 Design Strategy

The circuit is designed after it is divided in to three parts. Namely

a. The input section which will receive new walsh pattern compare it with the existing pattern. If it detects a change in pattern from the previous pattern it provide this information to the state machine, then it provides information about for which channel the walsh pattern has changed one after the other sequentially with each clock cycle.

b. The section which will generate different signals and control lines required for walsh distribution can easily be designed using state machine approach. The state machine which will traverse through various of its states generating different signals required, with the help of the signals provided by the input section and a three bit sequencer which keeps track of present and next state.

c. The registered Multiplexors which will let the right signal pass for the DSP bus so that the bus contention is avoided. The registered Muxes are required to attain a better eye pattern for the DPC latches.

4 Input Section

The walsh pattern will be tested for a change in value *(with respect to the previous value)* with the help of XOR gates. And then asserts a *Bus Request* for the processor to release its bus. As XORs will tell if the change is there ORing all the changes will easily produce a high going pulse which can be used as *bus request* after inverting it. Simultaneously the XORed data is

loaded in to a shift register. The shift clock to this register is ORed with the *Bus Grant* from the processor. Hence the register starts shifting only after *Bus Grant* is asserted.

5 State Machine for Walsh Distribution

The State Machine which generates all required signals and manages DSP bus during bus grant is a Mealy Machine. Its has six well defined states.*(see figure.)* It takes as input the Walsh pattern serially shifted in, and a clock which is 32 MHz. A sequencer will keep track of the present state and next state. The transition from one state to other will take place depending upon the change in walsh pattern from the previous value.

State Machine enters its first state with a low going *Bus Grant* in fact the state machine is functional only during the time when ***bus grant*** is low i.e. asserted. During its starting state it resets all outputs, resets the sequencer and waits for next clock (inverted of the one given to shift register). The new walsh pattern is loaded and shifted out of a serial register one by one. At each clock cycle a 3 bit sequencer gets updated and a output of serial shift register is inputed to the Machine. This together with the sequencer output, will define in which state the Machine will go in. In a maximum of 6 clock cycles *besides the time it takes for the processor to release its bus* . (after it detects a change in walsh pattern) it will be able to set the new walsh patterns on relevant DPC cards . It will leave the DPC cards which don't undergo a walsh change untouched which will avoid any possibility of glitches on those DPC's.

The State Diagram is as shown. This require a altera EM 5032 EPLD a 22V10 and some more discrete logic *(this is optional)*. The 22V10 will be programmed as a Multiplexor to allow State Machine generated signals on DSP bus when DSP asserts a bus grant. The EM5032 EPLD will hold the State Machine, sequencer and the shift register (hopefully, if not that has to be divided in to two devices.). and discrete logic will take care of the XOR gates and other logic gates (this also may go in a 16L8 if needed.

6 Conclusion

This implies that the maximum skew which will now occur between the two DPC cards can not exceed more then 400 ns as compared to the earlier well within microsec range. This also implies that even if walsh has the priority to reduce the skew there is sufficient time left for the processor to download delays and read total power. It will also reduce the possibility of glitches on the DPC cards which don't undergo a walsh change. The possibility of glitches is also reduced in other control signals on the DPC cards which are downloaded with new walsh pattern because the two functions are now totally decoupled.

7 References

1. C.R. Subrahmanya, *Data Preparation Card for GMRT FX Systems.*
2. Analog Devices *ADSP2101 data Sheets*
3. Analog Devices *ADSP2101 User's Manual*
4. Alka Dikshit, *Data Preparation Card (ver0.1).*
5. Alka Dikshit, *Control Card (ver1)*
6. Venkat S., *Walsh and related issues*
7. Altera Inc. *Altera Data Book and Altera Application Book*

STATE DIAGRAM FOR WALSH DISTRIBUTION