RFL Electronics, Inc.

DACS / ILS Technical Note 01. **Rev. 07.22.1996 rdw**

IMUX 2000

DACS Clock Switching Mechanism

The DACS/ILS Processor monitors the state of ports and relationship of clocks on the DACS board which it controls. Depending on the situations and user settings, various actions are taken which result in switching of source of clock supplying the module.

All clocks are referred to by the address from which they originate. External source, internal oscillator and ports 1 through 6 may be utilized. Which of these sources ultimately will supply the system clock is determined by the decision process described in this note.

The hardware and software involved in this decision process utilizes the concept of Actual, Reference and Requested clocks.

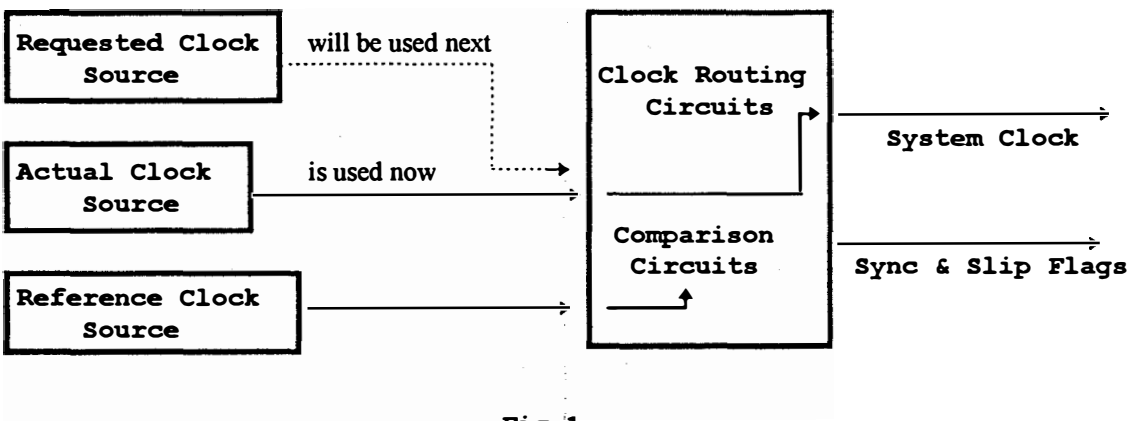

Fig.1.

Because RFL and Hubbell® have a policy of continuous product improvement, we reserve the right to change designs and specifications without notice.

- The **actual** clock address is the location of the source currently supplying the system clock.
- The **reference** clock address is the location of the source providing signal for clock switch-over synchronization and clock slip detection.
- The **requested** clock address is the location of the source which should become the system clock, if various other constraints are satisfied.

The software is using the requested clock to get the hardware to switch to a new clock source. In a nutshell, if the requested and reference clock addresses are the same, the hardware will proceed with setting the actual clock to requested. This implies, of course, that in order to switch clocks, both requested and reference clock addresses must be set.

The details of this procedure are shown in Figure 2. As mentioned above, actual clock address denotes the source of current system clock. This address is held in a latch.

Another clock is used as reference. This is typically the "alternate" clock specified in the DACS "set:clock" command.

The reference clock waveform is compared to the actual clock waveform in order to determine clock slip. If the waveforms slip by each other by more than 4 clocks, a flag will be set to signal this to the controller together with the direction of slip. The same waveforms are compared in order to determine if they are in phase. If they precisely coincide, a sync pulse is generated which triggers the switch-over process. As a result, a switch will take place without glitches due to clock disturbances.

When a switch is desired, software presents an address of the requested clock to the address latch. The same address must be also already present as the reference clock address. If these two addresses are indeed equal, the hardware will wait for the next instance of actual and reference clocks being in sync. At that moment, the requested address will be latched-in as the new actual address.

After requesting a switch, software is comparing the requested and the actual clock addresses to determine if it took place. If these addresses are identical, then it will be understood that the synchronization and other constraints were satisfied and that hardware accomplished the switch.

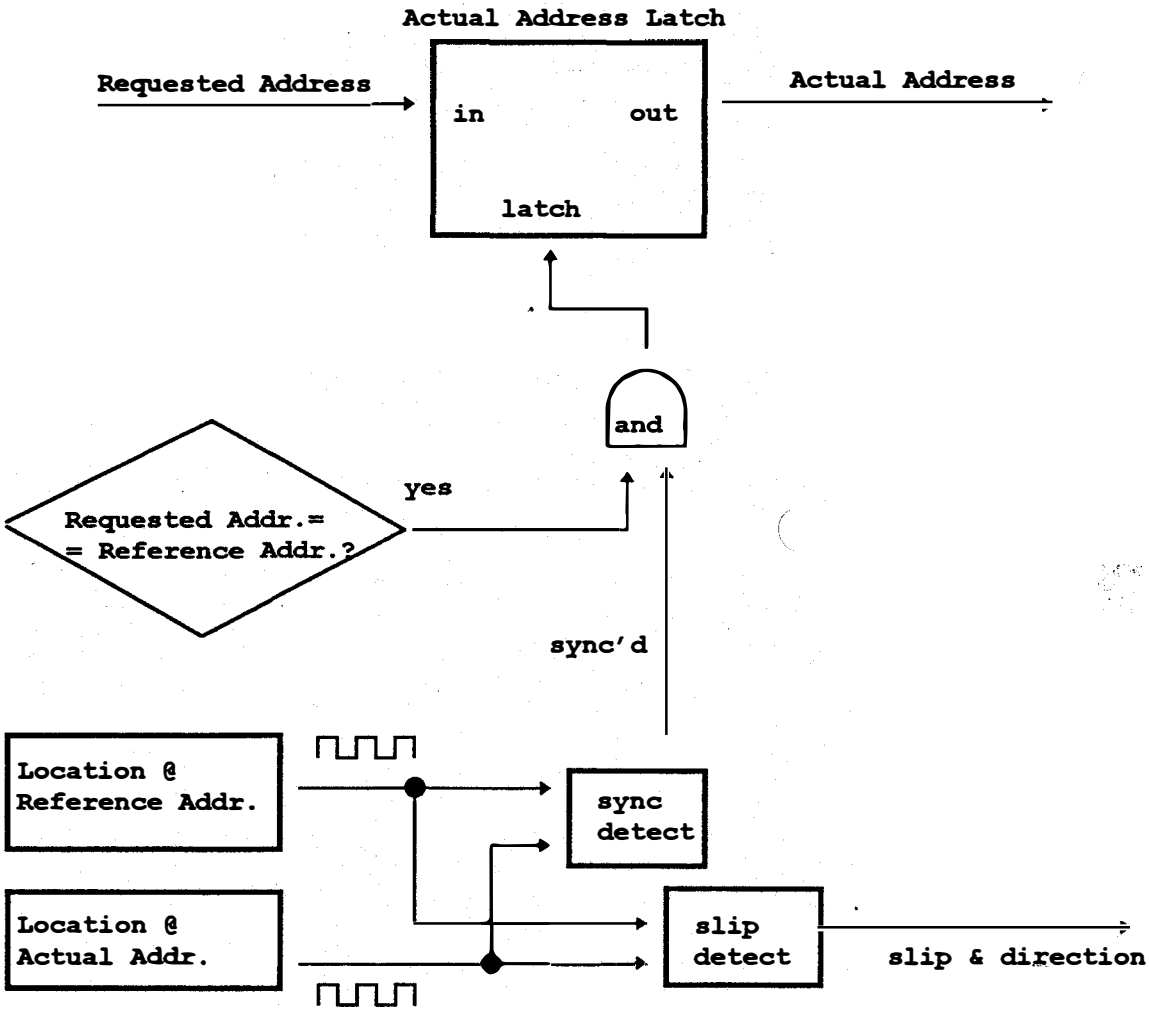

Fig.2.

The process of switching consists therefore of several discrete steps, which are outlined on following pages. These sequences of events pertain to a hypothetical situation where a DACS receives its clock on port 4 and is set up to use port 1 as an alternate via **::set:clock=4,1;** command.

Sequence of Events: DACS switches clocks due to port failure.

	Clock sources:	Actual	Refer.	Request.
Steady state - clock from 4, alternate from 1.		4	1	4
Port 4 fails. AIS or loss of data detected. Software requests port 1.		4	1	**1**
Hardware waits until reference is in sync with actual and switches to reference.		**1**	1	1
Software detects that actual clock became the same as requested. When port 4 restored, sets reference to 4.		1	**4**	1
Steady state - clock from 1, alternate from 4.				

Sequence of Events: DACS switches clocks due to clock slip.
(Node is not directly involved in break)

	Clock sources:	Actual	Refer.	Request.
Steady state - clock from 4, alternate from 1.		4	1	4
Two consecutive clock slips in the same direction detected. Software requests port 1.		4	1	**1**
Hardware waits until reference is in sync with actual and switches to reference.		**1**	1	1
Software detects that actual clock became the same as requested, sets reference to 4.		1	**4**	1
Steady state - clock from 1, alternate from 4.				

Sequence of Events: User switches clocks to external (or other port).

	Clock sources:	Actual	Refer.	Request.
Steady state - clock from 4, alternate from 1.		4	1	4
User requests external clock (E) or another specific port source. Software sets reference to E and requests E.		4	**E**	**E**
Hardware waits until reference is in sync with actual and switches to reference.		**E**	E	E
Software detects that actual clock became the same as requested, sets reference to 1 (assuming user selected E,1).		E	**1**	E
Steady state - external clock , alternate from 1.				

Sequence of Events: Alternate-Master DACS switches clocks @ port failure

Clock sources: Actual Refer. Request.

Steady state - clock from 4, alternate from 1.

4 1 4

Port 4 fails. AIS or loss of data detected.
Software sets reference to internal (I)
and requests internal clock.

4 I I

Hardware waits until reference is in sync
with actual and switches to reference.

I I I

Software detects that actual clock became the
same as requested, sets reference to 1.

I 1 I

Software checks now for clock slip
between reference (1) and actual (I).

No slip detected.
There is no other master
in the system. This node
must become master.

Slip detected.
There is another
master elsewhere.

Slip detected: request clock 1.

I 1 1

Hardware waits until reference is in sync
with actual and switches to reference.

1 1 1

Software detects that actual clock became the
same as requested.
When port 4 restored, sets reference to 4.

1 4 1

Steady state - system clock from 1, alternate from 4.

No slip: software stays with internal timing
as long as port 4 is failed.

I 1 I

When port 4 recovers, software sets reference
and requested clocks to 4.

I 4 4

Hardware waits until reference is in sync
with actual and switches to reference.

4 4 4

Software detects that actual clock became the
same as requested, sets reference to 1.

4 1 4

Steady state - system clock from 4, alternate from 1.