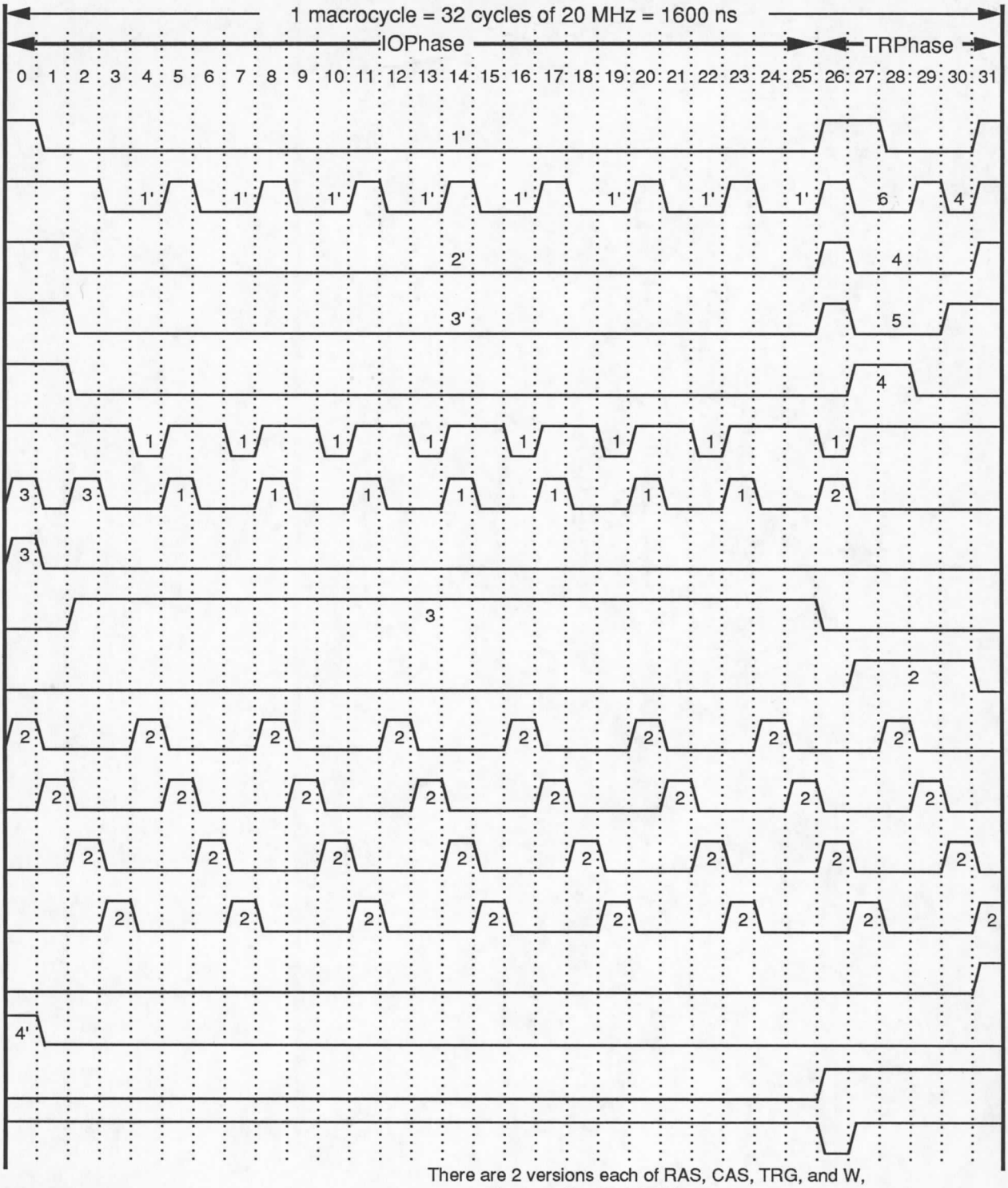
Macro Generator Outputs



Timing notes:

1 - pulse here only if TxMacro or WrMacro
1' - and correct bank enabled

2 - pulse here only if TxMacro macro (reading memory)
2' - and correct bank enabled

3 - pulse here only if WrMacro macro (writing memory)
3' - and correct bank enabled

4 - pulse here only if TfMacro
4 '- pulse here only if previous macro was TfMacro

5 - pulse here only if WMC or STM Transfer macro

6 - pulse here only if Refresh macro

There are 2 versions each of RAS, CAS, TRG, and W, each controlling 32 VRAMs.
They are identical during the refresh/transfer phase, but only one is potentially active during the I/O phase.

An address mux, controlled by RowSelH and TRPhaseH, provides A<0:8>, which are identical for all VRAMs.

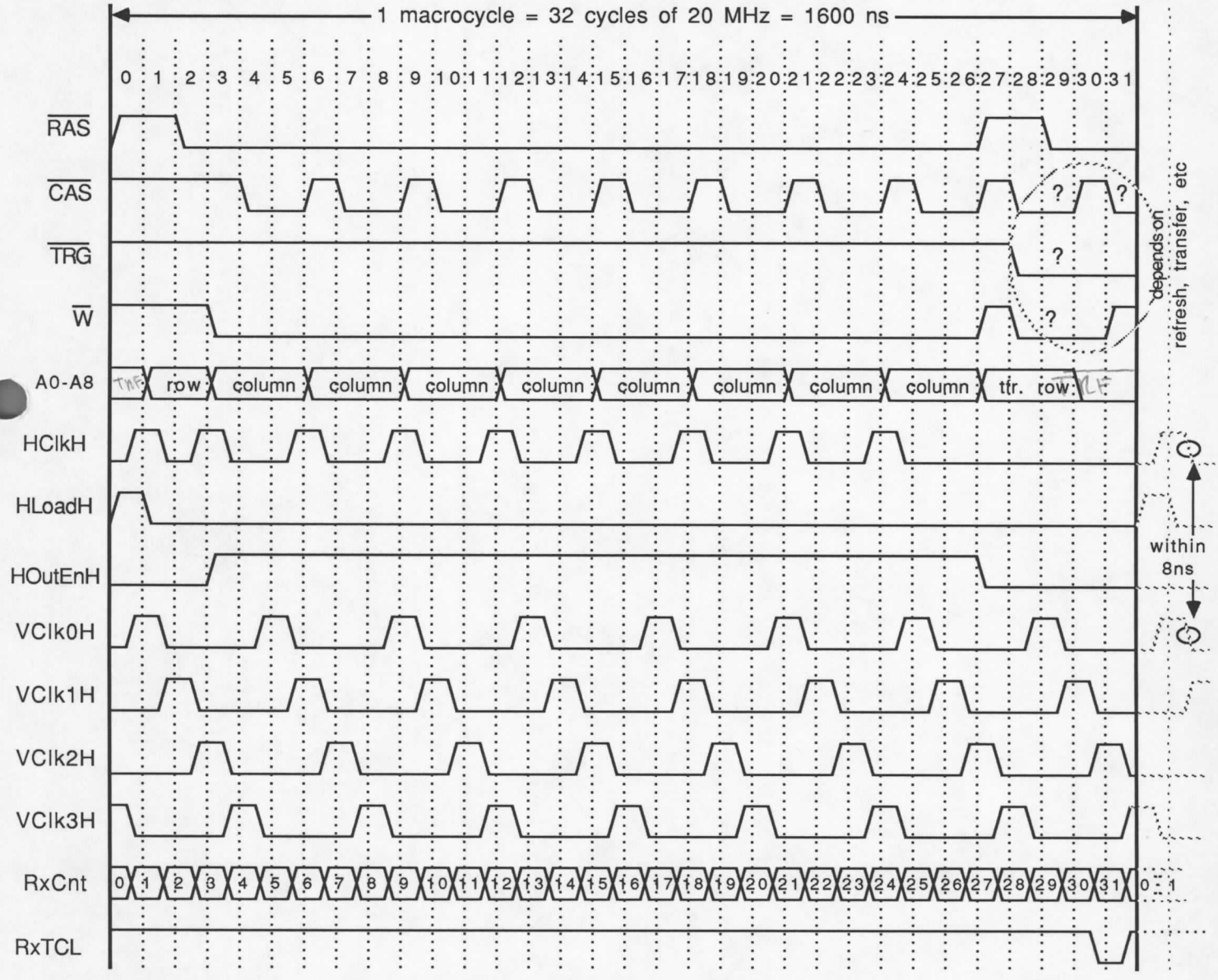
VRAM control signals SE and SC are generated by the Transfer Controller. VRAM control signal DSF is tied low.

All signals clocked on rising edge of Clk20H, with Tpd = 2 to 12 ns.

VRAM controls are delayed by one 20MHz cycle, Corner-Turner control signals go directly to CTs.

All Corner-Turner control signals except VOutEnab<0:3> and VClk<0:3> are common to all 4 CT chips.

Macrocycle Timing for "Receive BS Data"

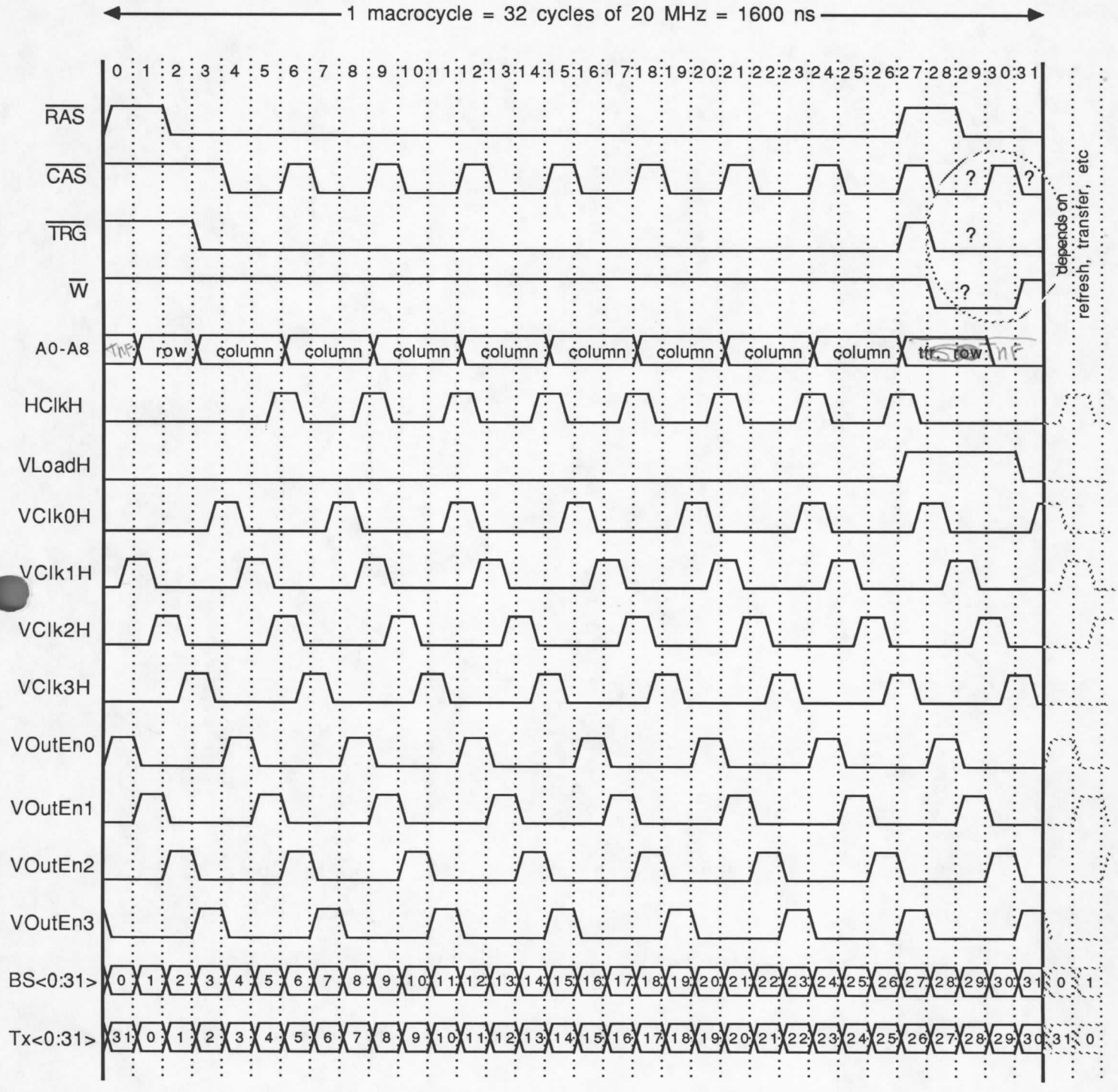


VLoadH and VOutEn<0:3>H are low for the entire macrocycle.

Assume all signals propagate between 0 and 12 ns from beginning of cycle.

Write data must be valid 0 ns before until 30 ns after falling edge of CAS (assuming W falls before CAS falls).

Macrocycle Timing for "Send BS Data"



Assume all signals propagate between 0 and 12 ns from beginning of cycle.

HOutEnabH and HLoadH are low for the entire macrocycle.

TxPut is low for the entire 1st macrocycle of the command, and high for the entirety of the remaining macrocycles.

Data is valid starting 35ns after CAS falls, till 0ns after CAS or TRG rises.
(delay of CAS transition, ringing on data wires)

What about CToE

Same for CToE, except remains valid 1 more cycle

This FSM lives on two different PALS, each is clocked by Clk20H (rising edge).
Assumes message includes destination address which is just treated as a command opcode. Commands may not cross message boundaries.
If unexpected 'Head' is seen, will generate BS error, go to Idle, and continue parsing. However, if error occured during "receive BS data" command, the Head word is lost so next message is likely to be messed up too (unless its dest-addr was a no-op.
Undefined BS command aslo generates BS error.
Assumes IO commands do not complete until last macro is over, so OK to do StartIO immediately.
Whenever TxTrig or WrTrig is asserted, must begin checking for MacroEnd immediately.
Note that Cmd input to PAL, and counters loaded by StartIOL, are driven by TxLatch, so there is a cycle of delay. However, Head bit is not delayed (comes directly from RxLatch).

'Reset' input causes immediate jump to Idle, with all outputs de-asserted. All other transitions implicitly assume ~Reset.

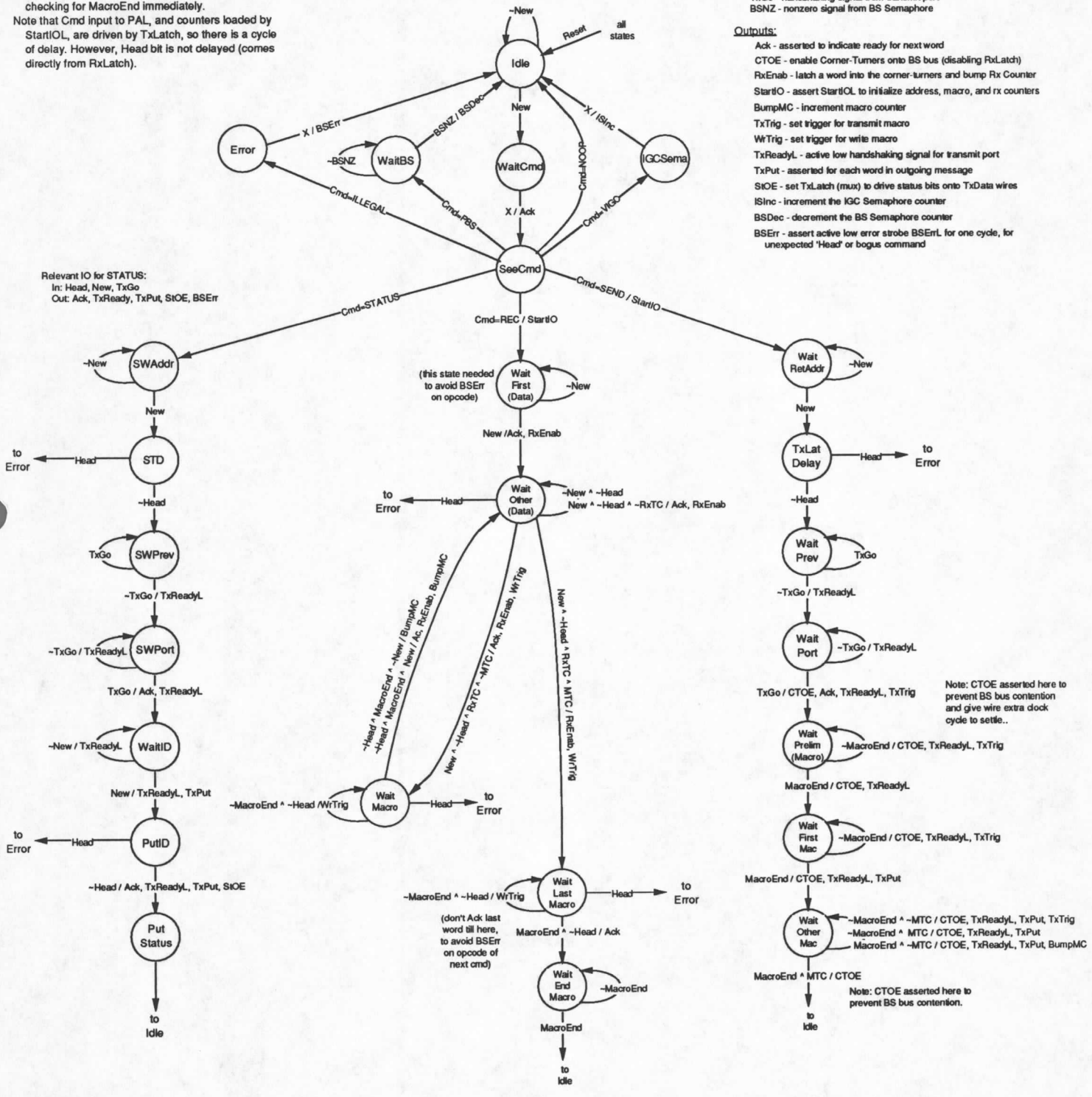
Inputs:

Reset - unconditonally resets to Idle state
Head - resets to Idle and generates BS error if occurs at other than opcode position
New - indicates next word available in RxLatch
Cmd - BS command op code, ILLEGAL for undefined command
- NOOP for no-operation
- VIGC for V op on IGC semaphore
- PBS for B op on BS semaphore
- STATUS for send-status command
- REC for receive BS data command
- SEND for send BS data command

MTC - Macro-counter terminal count MTCL asserted
MacroEnd - asserted during last cycle of a macrocycle
RxTC - terminal count RxTCL from 5-bit Rx Counter is asserted
TxGo - handshaking signal from transmit port
BSNZ - nonzero signal from BS Semaphore

Outputs:

Ack - asserted to indicate ready for next word
CTOE - enable Corner-Turners onto BS bus (disabling RxLatch)
RxEnab - latch a word into the corner-turners and bump Rx Counter
StartIO - assert StartIOL to initialize address, macro, and rx counters
BumpMC - increment macro counter
TxTrig - set trigger for transmit macro
WrTrig - set trigger for write macro
TxReadyL - active low handshaking signal for transmit port
TxPut - asserted for each word in outgoing message
StOE - set TxLatch (mux) to drive status bits onto TxData wires
ISInc - increment the IGC Semaphore counter
BSDec - decrement the BS Semaphore counter
BSErr - assert active low error strobe BSErrL for one cycle, for unexpected 'Head' or bogus command



State Diagram for Backing Store Port FSM (RDBSFSM and RDBSOTHER)

OLD VERSION

This FSM lives on two different PALS, each is clocked by Clk20H (rising edge).
Assumes message includes destination address which is just treated as a command opcode. Commands may not cross message boundaries.
If unexpected 'Head' is seen, will generate BS error, go to Idle, and continue parsing. However, error occured during "receive BS data" command, the Head word is lost so next message is likely to be messed up too (unless its dest-addr was a no-op.
Undefined BS command aslo generates BS error.
Assumes IO commands do not complete until last macro is over, so OK to do StartIO immediately.
Whenever TxTrig or WrTrig is asserted, must begin checking for MacroEnd immediately.
Note that Cmd input to PAL, and counters loaded by StartIOL, are driven by TxLatch, so there is a cycle of delay. However, Head bit is not delayed (comes directly from RxLatch).

'Reset' input causes immediate jump to Idle state, with all outputs de-asserted. All other transitions implicitly assume ~Reset.

lost first word of BS_TRANSMIT because of dynamic nature of CT outputs

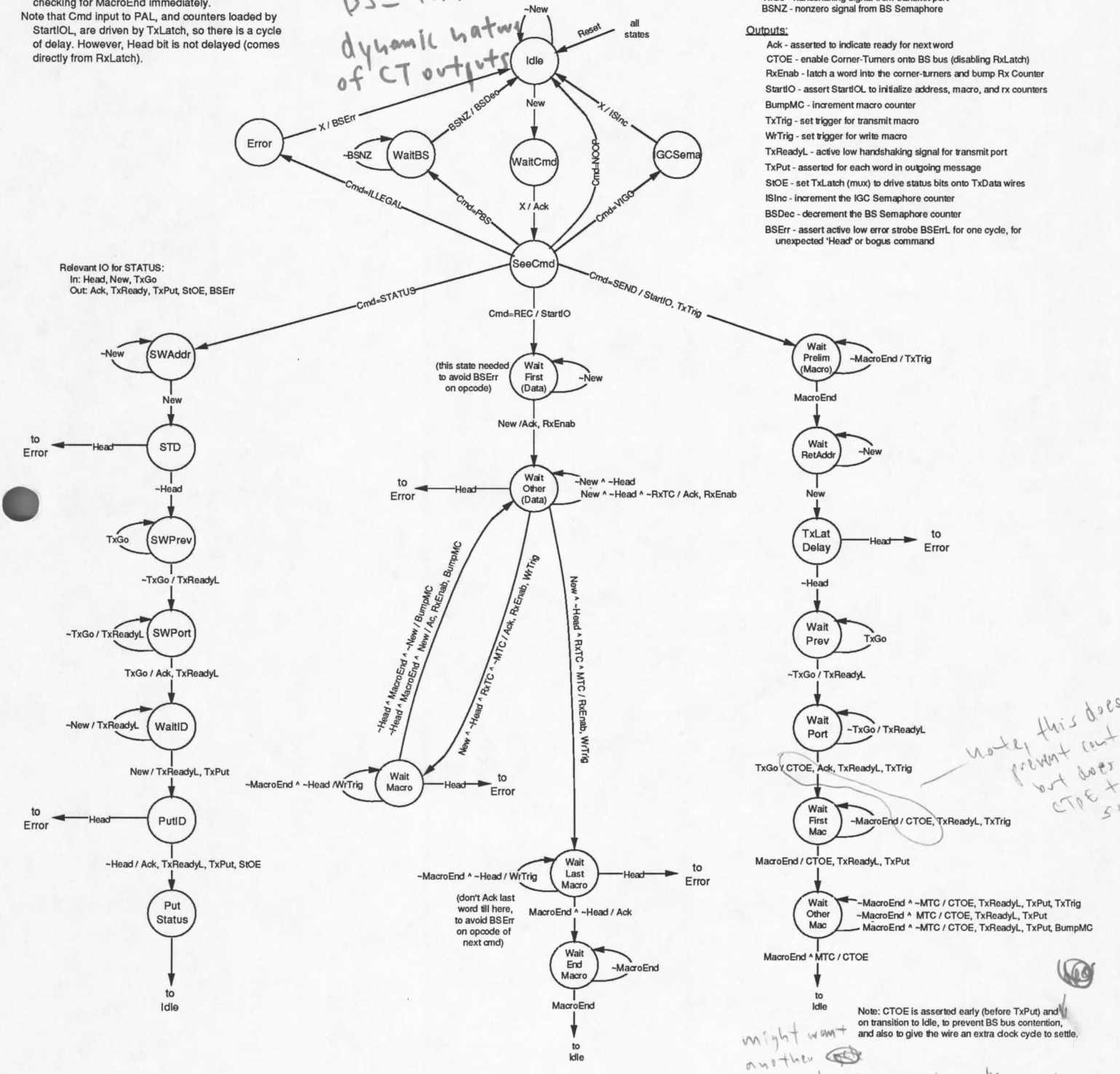
Inputs:

Reset - unconditonally resets to Idle state
Head - resets to Idle and generates BS error if occurs at other than opcode position
New - indicates next word available in RxLatch
Cmd - BS command op code, ILLEGAL for undefined command
- NOOP for no-operation
- VIGC for V op on IGC semaphore
- PBS for B op on BS semaphore
- STATUS for send-status command
- REC for receive BS data command
- SEND for send BS data command

MTC - Macro-counter terminal count MTCL asserted
MacroEnd - asserted during last cycle of a macrocycle
RxTC - terminal count RxTCL from 5-bit Rx Counter is assertted
TxGo - handshaking signal from transmit port
BSNZ - nonzero signal from BS Semaphore

Outputs:

Ack - asserted to indicate ready for next word
CTOE - enable Corner-Turners onto BS bus (disabling RxLatch)
RxEnab - latch a word into the corner-turners and bump Rx Counter
StartIO - assert StartIOL to initialize address, macro, and rx counters
BumpMC - increment macro counter
TxTrig - set trigger for transmit macro
WrTrig - set trigger for write macro
TxReadyL - active low handshaking signal for transmit port
TxPut - asserted for each word in outgoing message
StOE - set TxLatch (mux) to drive status bits onto TxData wires
ISInc - increment the IGC Semaphore counter
BSDec - decrement the BS Semaphore counter
BSErr - assert active low error strobe BSErrL for one cycle, for unexpected 'Head' or bogus command

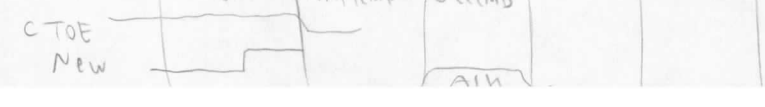


Note: CTOE is asserted early (before TxPut) and on transition to Idle, to prevent BS bus contention, and also to give the wire an extra clock cycle to settle.

note, this doesn't prevent contention but does allow CTOE to settle

might want another state here before Idle, to make sure RxLatch OE'd for next opcode

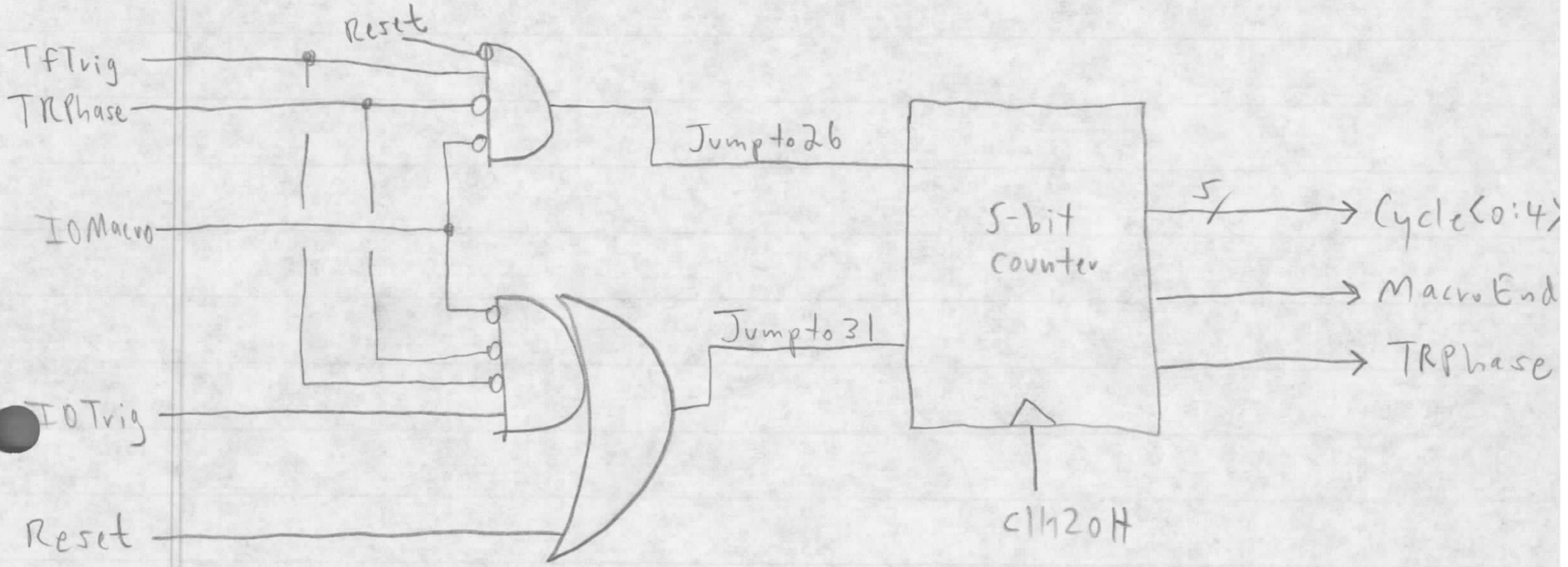
State Diagram for Backing Store Port FSM



Macrocycle Counter

are TFTrig & IOMacro ready soon enuf?

22V10 (probably could do using 16R8D)



$MacroEnd = Jumpto31 + (Count == 30)$

$TrPhase = Jumpto31 + Jumpto26 + (Count \in [25, 30])$

worst → $Count4 = Jumpto31 + Jumpto26 + (\overline{Count4} \cdot Count3 \cdot Count2 \cdot Count1 \cdot Count\phi)$
$+ Count4 \cdot \overline{Count3} + Count4 \cdot \overline{Count2} + Count4 \cdot \overline{Count1} + Count4 \cdot \overline{Count\ldots}$

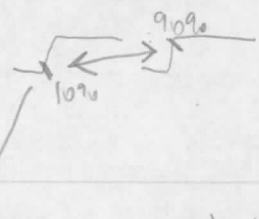
Count3 = ...

Count2 =

Count1 =

Count0 =

on transfers, may need for TRG to be low during CAS low time (cycle 30)
(Hitachi for Read to F cycle)



Allowable Slop in VRAM Controls

	min ↔ max	absolute max
RAS hi time for -12 = 90ns	10	
CAS↓ to RAS↑ for -12 = 35ns	15	
RAS↓ to CAS↓ = 120ns	30	
access tm for CAS↓ = 30ns		45
RAS↑ to WE↓ = 10ns		40
RAS↓ to TRG↑ (TF cycle) = 90ns (95 for waived TE)	10	
TRG↑ to next RAS↓ = 90ns	10	