

Collider and Recycler SDA Timing for Pbars

E. Harms
22 October 2002

Current Scheme:

Mode	Arm	Set	End	Disarm
Unstack Pbars	V:CASPBT = 2	V:PBXFER != 0	V:SETPBT = 5 +30s	V:SETPBT = 5
Transfer Pbars from Accum to MI	V:CASPBT = 3	Event \$91 + 0 sec.	V:SETPBT = 5 + 30 sec.	V:SETPBT = 5
Accelerate Pbars in the MI	V:CASPBT = 4	Event \$91 + 17 sec.	V:SETPBT = 5 + 30 sec.	V:SETPBT = 5
Coalesce Pbars in the MI	V:CASPBT = 5	Event \$91 + 17 sec.	V:SETPBT = 5 + 30 sec.	V:SETPBT = 5
Unstack Pbars for RR	V:CASPBT = 6	V:PBXFER != 0	V:SETPBT = 4 + 30 sec.	V:SETPBT = 4
Transfer Pbars from Accum to MI for RR	V:CASPBT = 7	Event \$E0 + 18 sec.	V:SETPBT = 4 +30 sec.	V:SETPBT = 4
RR Inject Pbars	V:CASPBT = 9	Event \$94 + 18 sec.	V:SETPBT = 4 + 30 sec.	V:SETPBT = 4
Stack Pbars in the RR	V:CASPBT = 10	Event \$9A	V:SETPBT = 4 + 30 sec.	V:SETPBT = 4

V:CASPBT is set by the Pbar Sequencer
 = 1 in 'Start Shot Set Up'
 = 2,3,4,5 set concurrently in 'Continue Shot Set Up'
 = 6,7,9,10 in 'Recycler Unstack and Transfer'

V:PBXFER is set to value of A:SHTNUM in 'Load Collider Pbars' and 'Recycler Unstack & Transfer'
 A:SHTNUM is set to 0 at start of Shot Set Up and incremented immediately before each transfer
 V:SETPBT is set by the Pbar Sequencer
 = 1 in 'Start Shot Set Up'
 = 5 in 'Return to Stacking' and 'Recycler Return to Stacking' (should probably be set to 4 in Recycler 'Return to Stacking')

Event \$91 – *Unstack Pbars*, triggered by Sequencer via TLG
 Event \$E0 – *Pbars from MI reset* triggered by Sequencer via TLG
 Event \$9A – *Pbar to MI Xfer Prep* triggered by \$91 + 21.0667 sec.
 Event \$94 – *Injected Pbar Synch*, mirror of MIBS \$7A
 MIBS \$7A – *Accum – MI Pbar Transfer* triggered by \$91 + 22.63333 sec.

Problem:

Pbar shots for both Tevatron and Recycler can show up in multiple places with inconsistent numbering

Proposed Scheme:

Mode	Arm	Set	End	Disarm
Unstack Pbars	V:SEQMD2 = 8	V:PBXFER != 0	V:SEQMD2 != 8 +30s	V:SEQMD2 != 8
Transfer Pbars from Accum to MI	V:SEQMD2 = 8	Event \$91 + 0 sec.	V:SEQMD2 != 8 + 30 sec.	V:SEQMD2 != 8
Accelerate Pbars in the MI	V:SEQMD2 = 8	Event \$91 + 17 sec.	V:SEQMD2 != 2 + 30 sec.	V:SEQMD2 != 8
Coalesce Pbars in the MI	V:SEQMD2 = 8	Event \$91 + 17 sec.	V:SEQMD2 != 8 + 30 sec.	V:SEQMD2 != 8
Unstack Pbars for RR	V:SEQMD2 = 29	V:PBXFER != 0	V:SEQMD2 != 29 + 30 sec.	V:SEQMD2 != 29
Transfer Pbars from Accum to MI for RR	V:SEQMD2 = 29	Event \$E0 + 18 sec.	V:SEQMD2 != 29 +30 sec.	V:SEQMD2 != 29
RR Inject Pbars	V:SEQMD2 = 29	Event \$94 + 18 sec.	V:SEQMD2 != 29 30 sec.	V:SEQMD2 != 29
Stack Pbars in the RR	V:SEQMD2 = 29	Event \$9A	V:SEQMD2 != 29 + 30 sec.	V:SEQMD2 != 29

For Loading the Collider, V:PBXFER will be referenced to V:XFER rather than A:SHTNUM

V:SEQMD2 is set by which Pbar Sequencer aggregate is playing
 = 8 is 'Load Collider Pbars'
 = 29 is 'Recycler Unstack & Transfer'

Using this scheme, Cases are armed only when the Sequencer begins playing the appropriate aggregate and ends when the aggregate is no longer invoked. Question: will this cause SDA problems if an aggregate is ended prematurely or bombs out?

A:SHTNUM will be set = 0 at the end of every aggregate involving Unstacking so that each aggregate will begin with SHTNUM = 1