Required sequence of operations for the optical WF recontruction process and proposed implementation

Prepared by	M. Xompero, M. Andrighettoni, D. Zanotti, A. Riccardi
Approved by	A. Riccardi
Released by	

INAF – Osservatorio Astrofisico di Arcetri
Largo E. Fermi, 5 - 50125 Firenze - ITALY
http://adopt.arcetri.astro.it

ABSTRACT

The present document reports the required sequence of operations that have to be implemented in the LBT672 crate electronics, Switch BCU and Slope Computer firmware in order to perform the WF reconstruction and handle safety controls on data. Moreover the required storage of diagnostics in internal buffers is described. Requirements are specified by Arcetri. Firmware has to be implemented by Microgate srl following the present requirements.

Modification Record

Version	Date	Author	Section/Paragraph affected	Reason/Remarks
1.0	30/12/2004	A. Riccardi		first issue
1.1	03/02/2005	A. Riccardi		update
2.0	01/04/2005	M. Xompero		update
2.1	23/06/2005	M. Andrighettoni		update
3.0	24/05/2006	M. Xompero		update
3.1	03/10/2006	M. Andrighettoni		update
3.2	30/10/2006	M. Andrighettoni		update
3.3	10/01/2007	M. Xompero		confirmation
3.4	15/01/2007	M. Andrighettoni		Map DSP names – equation names inserted and checked.
3.5	06/03/2007	M. Andrighettoni		fullbias_curr modified: now it contains only offload currents and bias current no more.
3.6	01/10/2007	M. Xompero		25, fixed formula:changed f_mol with full_bias_current of previous step 28, comment added 32, 33, fixed formula: fixed symbols for ff_cmd_current and ff_ud_current Legend updated Title changed.
D	30/01/2008	A. Riccardi		English translation of the Rel.3.6 document. Title changed.
E	25/11/2009	M. Xompero		Changes in computations and actions made at steps 17, 19, 20 related to new_delta_command

Abbreviations, acronyms and symbols

Symbol	Description
LBT	Large Binocular Telesecope
WF	Wavefront

Contents

1 Legend

italic lowercase: scalar quantities
boldface lowercase: columns vectors
boldface UPPERCASE: matrices

2 Symbols that are not explicitly defined in the text:

TBD: to be defined.
SC: slope computer operation
N_{CCD} = number of pixels of CCD
N_{SUB} = number of used sub-apertures
N_{ACT} = numbers of actuators
N_{MOD} = number of used modes
N = number of slope frames of previous steps to be used in the temporal digital recursive filter
M = number of modal coefficient frames of previous steps to be used in the temporal digital recursive filter. The recursive filter has N+M+1 taps including the weight of the current slope frame.

3 Glossary

parallel operation: operation made in parallel among the DSPs of the adaptive secondary crates for the computation of a vector. Every DSP performs the computations related to 4 elements of the vector to be later re-circulated.

local operation: every DSP of the adaptive secondary crates computes the locally used element of the vector and the result is not required to be re-circulated.

on-fly changeable: the variable, vector or matrix are managed using a double block of memory. That permits, while the optical loop is closed, the use of the first or the second block of memory depending on a dedicated trigger value. For every double block should be assigned a configuration variable of the trigger with, at least, 3 statuses: 0=do nothing (i.e. it continues to use the currently used block), 1=swap to the other memory block, 2=change the used block when the most-significative bit of the frame identification number *f* is set.

4 Symbols and related DSP variables

Symbol	DSP variable	Description
$\mathbf{p}$	m2_distance	Position read by the capacitive sensor
$\bar{\mathbf{p}}^{(f)}$	m2_dist_average	Mean position at the current step
$\mathbf{e}^{(f-1)}$	m2_precalc_modes	Pre-computed modes for the next step
$\mathbf{d}^{(f-1)}$	m2_precalc_slope	Pre- computed slopes for the next step
$\mathbf{m}^{(f)}$	m2_modes	Vector of calculated modes
$\bar{\mathbf{c}}_1^{(f)}$	m2_new_delta_command	Vector of absolute commando respect to bias command.
$\bar{\mathbf{c}}_1^{(f)}$	m2_new_delta_command	
$\Delta\mathbf{c}_1^{(f)}$	m2_ff_command	Delta-commands respect to previous step
$\Delta\mathbf{f}_F^{(f)}$	m2_ff_current	Delta-force given with feed-forward
$\mathbf{f}_{FC}^{(f)}$	m2_int_control_current	Mean control current integral force
$\bar{\mathbf{f}}_C^{(f)}$	m2_mean_control_current	Mean control current
$\mathbf{f}_{FF}^{(f)}$	m2_ff_pure_current	Feed-forward integrated force
$\mathbf{f}_0$	m2_bias_current	Bias current (ex: flattening force)
$\bar{\mathbf{f}}_0^{(f)}$	m2_fullbias_curr	Force offload current (TBC: and accelerometers contribution ??)
$\mathbf{f}_F^{(f)}$	m2_ff_ud_current	New feed-forward current without bias
$\mathbf{f}_{CMD}^{(f)}$	m2_cmd_current	Final new feed-foward current
$\mathbf{c}_0$	m2_bias_command	Bias command (ex: flattening command)
$\bar{\mathbf{c}}_0^{(f)}$	m2_fullbias_cmd	Final bias command (with command offload and accelerometer contributions)
$\mathbf{c}_1^{(f)}$	m2_pos_command	New position command/
$\mathbf{c}_{MOL}^{(f)}$	m2_cmol_a/b	Offload position command.
$\mathbf{f}_{MOL}^{(f)}$	m2_fmol_a/b	Offload force command.
$\mathbf{c}_{NOISE}^{(f)}$	m2_disturb_buffer	Disturbance command.
$\mathbf{c}_{TIP}, \mathbf{c}_{TILT}, \mathbf{c}_{PIST}$	TBD	Fixed tip/tilt/piston commando.
$\alpha_{TIP}, \alpha_{TILT}, \alpha_{PIST}$	TBD	Tip/tilt/piston coefficients from accelerometers
$\mathbf{G}$	m2_gain_G_a/b	Reconstructor filter gain.
$\mathbf{M}$	m2_matrix_M2C	Transformation matrix from modes to commands
$\mathbf{K}$	m2_matrix_K	Feed-forward matrix
$\mathbf{B}_i, \mathbf{A}_i$	m2_matrix_B0a/b m2_matrix_Ba/b_delay m2_matrix_A_delay	Recontruction matrix. Slopes filter matrix Modes filter matrix
$\mathbf{BF}_i, \mathbf{AF}_i$	m2_gain_BF1_a/b m2_gain_BF0_a/b m2_gain_AF1_a/b	Control current pseudo-integrator coefficients

$m_{MAX,i}, \Delta m_{MAX,i}$	m2_max_mode m2_dmax_mode	Thresholds on mode application
$c_{MIN,i}, c_{MAX,i}, \Delta c_{MAX,i}$	m2_min_cmd m2_max_cmd m2_dmax_cmd	Thresholds on command application
$f_{MAX,i}, f_{MIN,i}$	m2_min_curr m2_max_curr	Thresholds on force application

5 Requested sequence of operations

5.1 In the Slope Computer

1. An identification number *f* for the current frame of the CCD is generated (a 32-bit countercontatore with a unit increment for each step) together with a ParamSelector 32-bit dword for action triggers (each bit has a different meaning).
2. Input of pixel $\phi_i^{(f)}$ from CCD. *i*=1,2,...,N_{CCD} is the pixel index, *f* is the index of the current frame;
3. Correction of flat-field e bias: $q_i^{(f)} = a_i \phi_i^{(f)} + q_{0i}$. The $\mathbf{q}_0$ vector is on-fly changeable.
4. Sorting or rejecting of pixel using a user defined LUT. For simplifying the notation the pixels can be considered re-indexed as $q_{k,j}^{(f)}$ where *k* is the index of the sub-aperture and *j*=1,2,3,4 is the index of the corresponding pixel related to the sub-aperture *k*.
5. Computation of the signals $\sigma_{x,k}^{(f)}, \sigma_{y,k}^{(f)}$, *k* is the index of the sub-aperture. The signals are computed using one of the following three algorithms (it is not required the change of the algorithm while the optical loop is closed)
   a. $$\sigma_{x,k}^{(f)} = b_k (q_{k,1}^{(f)} + q_{k,2}^{(f)} - q_{k,3}^{(f)} - q_{k,4}^{(f)}),$$
   $$\sigma_{y,k}^{(f)} = b_k (q_{k,1}^{(f)} - q_{k,2}^{(f)} - q_{k,3}^{(f)} + q_{k,4}^{(f)}),$$
   $$b_k^{(f)} = \frac{1}{q_{k,1}^{(f)} + q_{k,2}^{(f)} + q_{k,3}^{(f)} + q_{k,4}^{(f)}}$$
   b. $$\sigma_{x,k}^{(f)} = m^{(f)} (q_{k,1}^{(f)} + q_{k,2}^{(f)} - q_{k,3}^{(f)} - q_{k,4}^{(f)}),$$
   $$\sigma_{y,k}^{(f)} = m^{(f)} (q_{k,1}^{(f)} - q_{k,2}^{(f)} - q_{k,3}^{(f)} + q_{k,4}^{(f)}),$$
   $$\mu^{(f)} = \frac{N_{SUB}}{\sum_{k=1}^{N_{SUB}} q_{k,1}^{(f)} + q_{k,2}^{(f)} + q_{k,3}^{(f)} + q_{k,4}^{(f)}}.$$
   The use of the $\mu^{(f)}$ in the formulas for $\sigma_{x,k}^{(f)}, \sigma_{y,k}^{(f)}$ is preferable. In the case that will not be possible, $\mu^{(f-1)}$ can be used.
   c. $$\sigma_{x,k}^{(f)} = u (q_{k,1}^{(f)} + q_{k,2}^{(f)} - q_{k,3}^{(f)} - q_{k,4}^{(f)}),$$
   $$\sigma_{y,k}^{(f)} = u (q_{k,1}^{(f)} - q_{k,2}^{(f)} - q_{k,3}^{(f)} + q_{k,4}^{(f)}),$$
   u = const.
6. To simplify the notation, we define the vector $\boldsymbol{\sigma}^{(f)}$ as the join of the $\boldsymbol{\sigma}_x^{(f)}, \boldsymbol{\sigma}_y^{(f)}$ signal vectors in a single column. The slope offset $\mathbf{s}_{off}^{(f)}$ has to be subtracted from the signal vector:
   $$\mathbf{s}^{(f)} = \boldsymbol{\sigma}^{(f)} - \mathbf{s}_{off}^{(f)},$$
   In the more general case, the slope-offset depends on time (in terms of *f* index) and has the following behavior:
   $$\mathbf{s}_{off}^{(f)} = \mathbf{s}_{off,0} + \alpha(f)\, \mathbf{s}_{off,1},$$
   where the component $\alpha(f)\, \mathbf{s}_{off,1}$ is used only during the calibration procedure of the on-sky optical interation

matrix in closed-loop. In the ordinary operations in closed-loop $\mathbf{s}_{off}^{(f)}$ matches $\mathbf{s}_{off,0}$ (the usual vector of slope-offset) and it is not depending on time. When the on-sky calibration is used, $\alpha(f)$ is a scalar value that changes with a sinus behavior with time and $\mathbf{s}_{off,1}$ represents the slope pattern that generates the mode to calibrate. The $\mathbf{s}_{off,0}$ vector id on-fly changable.

7. The CCD frame is saved together the ID frame *f*.
8. The signal frame $\mathbf{s}^{(f)}$ is saved together the ID *f*.
9. The signal vector $\mathbf{s}^{(f)}$ is transferred to the Switch BCU together the ID *f* using the write_same_dsp command.

5.2 In the Switch BCU (SW) and in the secondary mirror electronics (M2)

10. -SW- The Switch BCU is waiting for a new frame (set of slopes or a standalone frame).
 -M2- The secondary mirror is waiting for a start_rtr, start_mm, start_ff, start_dl setting event.
11. -SW- The switch BCU receives the slope vector and and forwards it to the secondary mirror appending on the vector tail the start_rtr (bit#0 = 1) signal.
12. -M2- In case the start_rtr (bit#0 = 1) signal is set; M2 computes the new vector of command modal coefficients $\mathbf{m}^{(f)}$ (parallel operation):
 $\mathbf{m}^{(f)} = \mathbf{G}\left(\mathbf{B}_0\mathbf{s}^{(f)} + \mathbf{d}^{(f-1)}\right) + \mathbf{e}^{(f-1)}$.
 The components $\mathbf{d}^{(f-1)} = \mathbf{B}_1\mathbf{s}^{(f-1)} + \cdots + \mathbf{B}_N\mathbf{s}^{(f-N)}$ and $\mathbf{e}^{(f-1)} = \mathbf{A}_1\mathbf{m}^{(f-1)} + \cdots + \mathbf{A}_M\mathbf{m}^{(f-M)}$ have been already computed during the previous step. **G** is a diagonal matrix of gains that is on-fly changeable (diagonal with N_{MOD} elements), $\mathbf{B}_0,\ldots,\mathbf{B}_N$ are $N_{MOD}\times(2N_{SUB})$ matrixes of the filter weights for the input signal and $\mathbf{A}_0,\ldots,\mathbf{A}_N$ are $N_{MOD}\times N_{MOD}$ matrixes of the filter weights for the output signals. Le matrixes $\mathbf{B}_0,\ldots,\mathbf{B}_N$ are on-fly changeable (that because the reconstructor should be on-fly changeable).
13. -M2- In case any of the $\mathbf{m}^{(f)}$ modes gives $\left|m_i^{(f)}\right| > m_{MAX,i}$ or $\left|m_i^{(f)} - m_i^{(f-1)}\right| > \Delta m_{MAX,i}$, the skip frame (bit#1 = 1) is set to 1 in the start_rtr word.
14. –M2- For each DSP the corresponding 4 values of the $\mathbf{m}^{(f)}$ vector are saved together the ID *f* in the parking memory slot of the diagnostic buffers. When the computation is ended the signal start_rtr (bit#0 = 0) is set to zero and the process goes back to the point 10.
15. -SW- The Switch BCU polls the start_rtr location. In case at least one DSP board has the skip-frame flag enabled (bit#1 = 1), the Switch BCU stops the sequence, increase the safe_skip_frame counter of one unit, transfers the signal start_dl with the only-diagnostics flag (bit#1 = 1) enabled. That generates the storing of the current diagnostic frame and jumps to the step 37. In case no skip-frame is flagged, the modal vector $\mathbf{m}^{(f)}$ is read by the Switch BCU and re-circulated to M2 appending on the tail the start_mm (bit#0 = 1) signal.
16. –M2- Computes the command vector with respect to the flattening command (parallel operations):
 $\mathbf{c}_1^{(f)} = \mathbf{M}\mathbf{m}^{(f)}$. M is a $N_{ACT}\times N_{MOD}$ matrix.
17. –empy.
18. –M2- Computation of the full_bias (command) values to be used in the following:
 $\overline{\mathbf{c}}_0 = \mathbf{c}_0 + \mathbf{c}_{MOL} + \alpha_{TIP}\mathbf{c}_{TIP} + \alpha_{TILT}\mathbf{c}_{TILT} + \alpha_{PIST}\mathbf{c}_{PIST}$
19. –empty.
20. –M2- The vectors $\Delta\mathbf{c}_1^{(f)} = \mathbf{c}_1^{(f)} + \mathbf{c}_{NOISE}^{(f)} - \overline{\mathbf{c}}_1^{(f-1)}$ and $\overline{\mathbf{c}}_1^{(f)} = \mathbf{c}_1^{(f)} + \mathbf{c}_{NOISE}^{(f)}$ are computed (parallel operation), in alternative we propose to implement the computation of the vectors

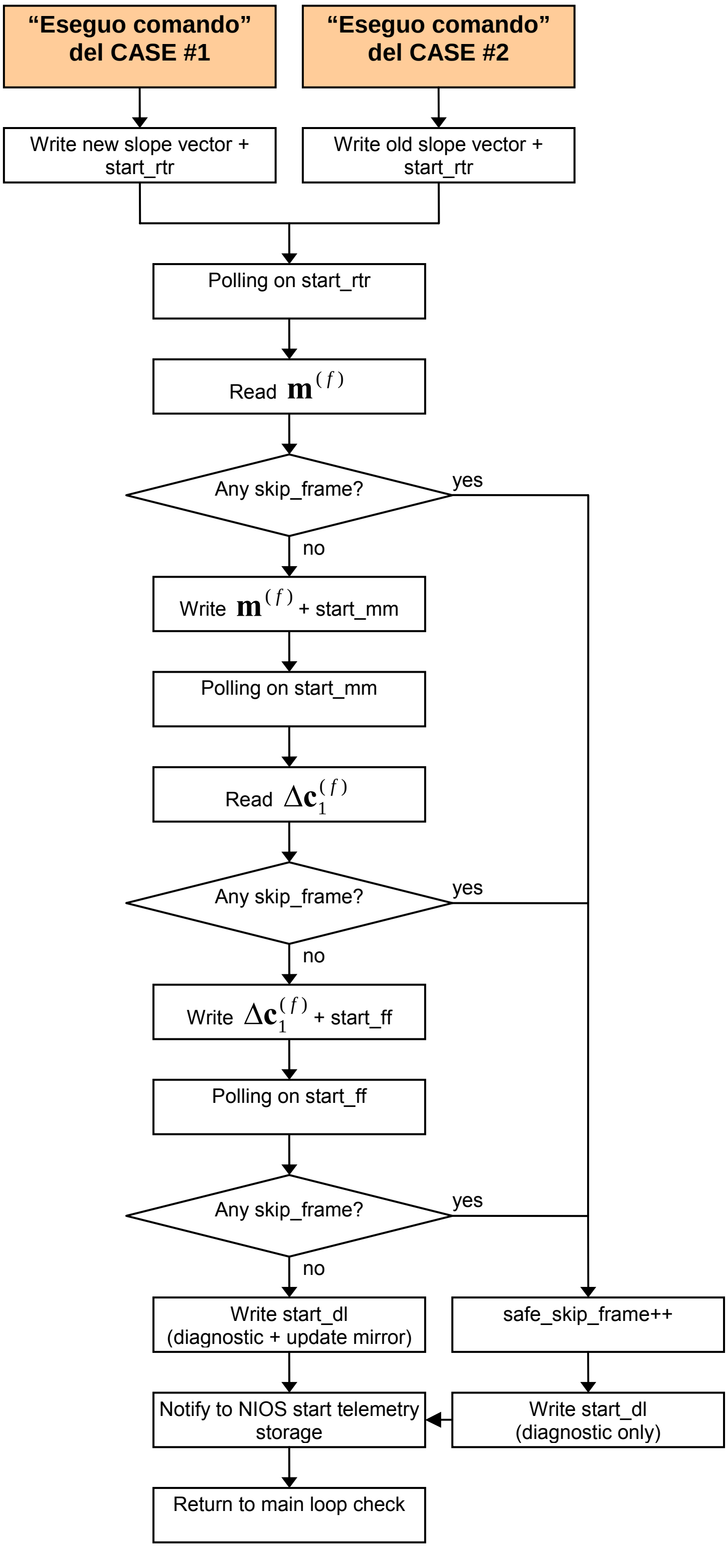
$\Delta\mathbf{c}_1^{(f)} = \mathbf{c}_1^{(f)} + \mathbf{c}_{NOISE}^{(f)} + \overline{\mathbf{c}}_0 - \overline{\mathbf{p}}^{(f)}$ and $\overline{\mathbf{c}}_1^{(f)} = \mathbf{c}_1^{(f)} + \mathbf{c}_{NOISE}^{(f)}$ where $\overline{\mathbf{p}}^{(f)}$ is the current average position (so called Mantegazza-style). Note: the assignment $\overline{\mathbf{c}}_1^{(f)} = \mathbf{c}_1^{(f)} + \mathbf{c}_{NOISE}^{(f)}$ overwrites the old value of $\overline{\mathbf{c}}_1$.

21. –M2- In case any element of the $\overline{\mathbf{c}}_1^{(f)}$ vector gives:
    $\overline{c}_{1,i}^{(f)} + \overline{c}_{0,i} > c_{MAX,i}$ o $\overline{c}_{1,i}^{(f)} + \overline{c}_{0,i} < c_{MIN,i}$ o $\left|\overline{c}_{1,i}^{(f)} - \overline{c}_{1,i}^{(f-1)}\right| > \Delta c_{MAX,i}$
    the skip frame flag (bit#1 = 1) is set in the start_mm dword.
22. –M2- For each DSP the corresponding 4 components of the $\Delta\mathbf{c}_1^{(f)}$ vector are stored in the parking memory slot for the diagnostig buffer. At the end of the computation the signal start_mm is set to zero (bit#0 = 0) and the process go back to the step 10.
23. –SW- The Switch BCU polls the start_mm memory location. In case at least one DSP board enabled the skip frame (bit#1 = 1) flag, the Switch BCU stops the sequence, increase the safe_skip_frame counter of one unit, transfers the start_dl signal with the flag of only-diagnostics enabled (bit#1 = 1); that triggers the storage of the current diagnostic frame and jumps to the step 37.In case no skip frame flag is detected, the delta command $\Delta\mathbf{c}_1^{(f)}$ is read back by the Switch BCU from M2 and re-circulated to M2 appending on the vector tail the signal of start_ff (bit#0 = 1).
24. –M2- The feed-forward (FF) delta-force is computed (parallel operation) $\Delta\mathbf{f}_F^{(f)} = \mathbf{K}\Delta\mathbf{c}_1^{(f)}$.
25. –M2- The contribution to the FF of the integral of the average control current is computed (parallel operation):
    $\mathbf{f}_{FC}^{(f)} = \mathbf{B}_{F0}\overline{\mathbf{f}}_C^{(f)} + \mathbf{B}_{F1}\overline{\mathbf{f}}_C^{(f-1)} + \mathbf{A}_{F1}\mathbf{f}_{FC}^{(f-1)}$
    Matrixes $\mathbf{B}_{F0}$, $\mathbf{B}_{F1}$ and $\mathbf{A}_{F1}$ are diagonal. The diagonal elements are on-fly changeable. Internally the average control force $\overline{\mathbf{f}}_C^{(f)}$ is computed from the force of the accumulators $\overline{\mathbf{f}}^{(f)}$ as come $\overline{\mathbf{f}}_C^{(f)} = \overline{\mathbf{f}}^{(f)} - \mathbf{f}_0 - \mathbf{f}_{FF}^{(f-1)} - \overline{\mathbf{f}}_0^{(f-1)}$ where $\mathbf{f}_{FF}^{(f)}$ is the integral contribution of the FF forces, given by: $\mathbf{f}_{FF}^{(f)} = \mathbf{f}_{FF}^{(f-1)} + \Delta\mathbf{f}_F^{(f)}$.
26. –M2- For each DSP the corresponding 4 values of the $\mathbf{f}_{FC}^{(f)}$ and $\mathbf{f}_{FF}^{(f)}$ vectors are stored in the parking memory slot of the diagnostic buffer.
27. –M2- Computation of the full_bias (current) value, including also a vector of force offload (aka modal offload) that will be used in the following: $\overline{\mathbf{f}}_0^{(f)} = \mathbf{f}_{MOL}^{(f)}$
28. –M2- In case any element of the vector $\mathbf{f}_{FF}^{(f)} + \mathbf{f}_{FC}^{(f)} + \overline{\mathbf{f}}_0^{(f)} + \mathbf{f}_0$ gives:
    $f_{FF,i}^{(f)} + f_{FC,i}^{(f)} + \overline{f}_0^{(f)} + f_0^{(f)} > f_{MAX,i}$ or $f_{FF,i}^{(f)} + f_{FC,i}^{(f)} + \overline{f}_0^{(f)} + f_0^{(f)} < f_{MIN,i}$, the skip frame flag (bit#1 = 1) in the start_ff flag is set. Note that the bias current is applied in any case without checks because it is supposed to be constant, so inside the thresholds from the beginning (and also because that is managed in the 70 Khz loop).
29. –M2- At the end of the computations the start_ff signal is set to zero (bit#0 = 0) and the process jumps back to the step 10.
30. –SW- The Switch BCU polls the start_ff memory location. in case at least one of the DSP boards has enabled the skip frame flag (bit#1 = 1), the Switch BCU stops the sequence, increases the safe_skip_frame counter by one unit, transfers the start_dl signal with the only-diagnostics flag (bit#1 = 1) enabled. That generates the storing of the current diagnostic frame and the process jumps to the step 37. In case no skip frame is detected the Switch BCU transfers the complete start_dl (bit#0 = 1) signal: that triggers also the update of the new commands of force and position for the local loop of the actuators. Moreover (bit#1 = 1) triggers also the storage of the current diagnostic frame.
31. –M2- Latches the accumulators and initalize the new diagnostic record before the updating of the new commands.

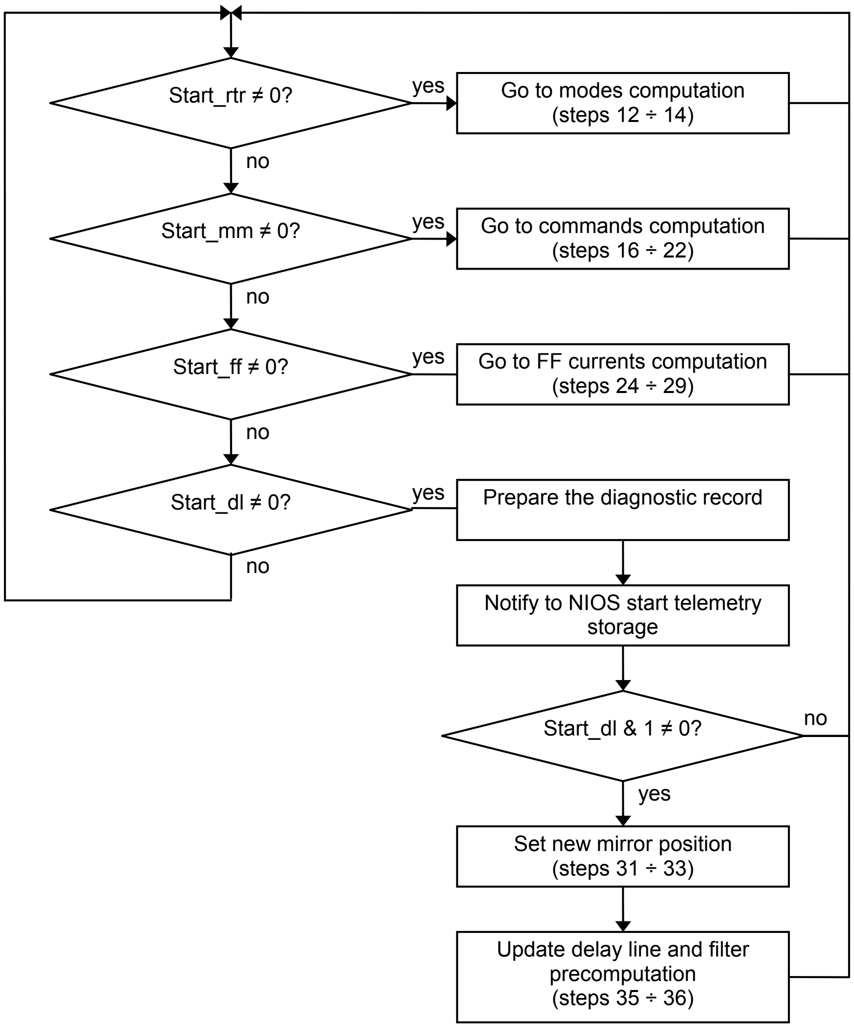
32. –M2- Computation of the new position command (parallel operation): $\mathbf{c}^{(f)} = \overline{\mathbf{c}}_1^{(f)} + \overline{\mathbf{c}}_0^{(f)}$ and computation of the new FF force command $\mathbf{f}_F^{(f)} = \mathbf{f}_{FF}^{(f)} + \mathbf{f}_{FC}^{(f)}$; moreover the final force command $\mathbf{f}_{CMD}^{(f)} = \mathbf{f}_F^{(f)} + \overline{\mathbf{f}}_0^{(f)}$.is computed
33. –M2- New FF force $\mathbf{f}_{CMD}^{(f)}$ and position $\mathbf{c}^{(f)}$ command are applied and the accumulators are started.
34. –M2- Sends the notification to the NIOS to start the DMA transition to upload the local (in the DSP memory) diagnostic buffer in the SDRAM memory.
35. –M2- Salves the new $\mathbf{m}^{(f)}$ and $\mathbf{s}^{(f)}$ vectors in the circular buffers of the delay-lines for the time filtering.
36. –M2- Computes (local operation) $\mathbf{d}^{(f)} = \mathbf{B}_1\mathbf{s}^{(f)} + \cdots + \mathbf{B}_N\mathbf{s}^{(f-N+1)}$ and $\mathbf{e}^{(f)} = \mathbf{A}_1\mathbf{m}^{(f)} + \cdots + \mathbf{A}_M\mathbf{m}^{(f-M+1)}$ to be used in the next recontruction step. At the end of the computation the signal start_dl is set to zero (bit#0 = 0) and the process jumps back to step 10.
37. –SW- Sends the notifucation to the NIOS to start the DMA transition to upload the local (in the DSP memory) diagnostic buffer in the SDRAM memory.
38. –SW- jumps back to the step 10.

6 Flow chart of the DSP code in the Switch BCU

\



7 Flow chart of the DSP code in the DSP boards



Doc_info_start
Title: Required sequence of operations for the optical WF recontruction process and proposed implementation
Document Type: Technical Report
Source: INAF-Osservatorio Astrofisico di Arcetri
Issued by: M. Xompero, M. Andrighettoni, D. Zanotti, A. Riccardi
Date_of_Issue: 25 Nov 2009
Revised by:
Date_of_Revision:
Checked by:
Date_of_Check:
Accepted by:
Date_of_Acceptance:
Released by:
Date_of_Release:
File Type:MS-WORD
Local Name:
Category: 400 Telescope Azimuth Platform
Sub-Category: 480 Telescope Controls
Assembly: 486 Adaptive Optics Software
Sub-Assembly:
Part Name:
CAN designation: 486f015
Revision: E
Doc_info_end