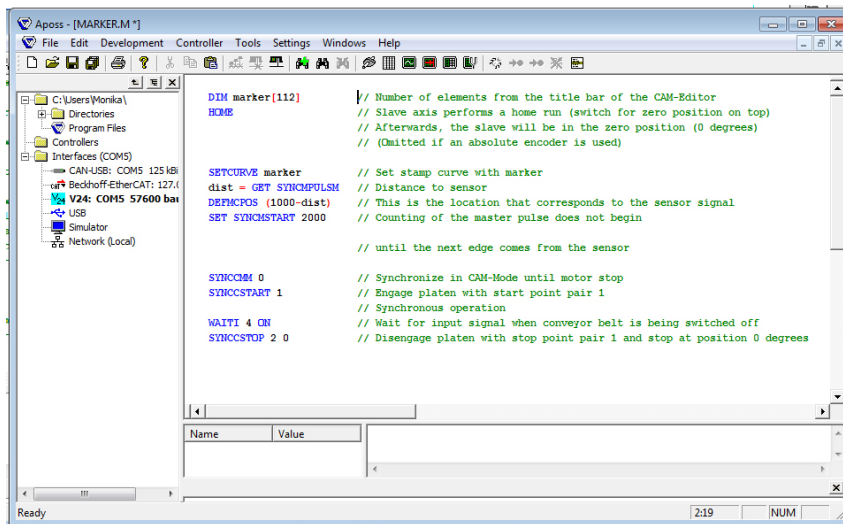


APOSS® win

An brief Overview of all APOSS Commands

Axis Positioning and Synchronization Commands for Drive Engineering



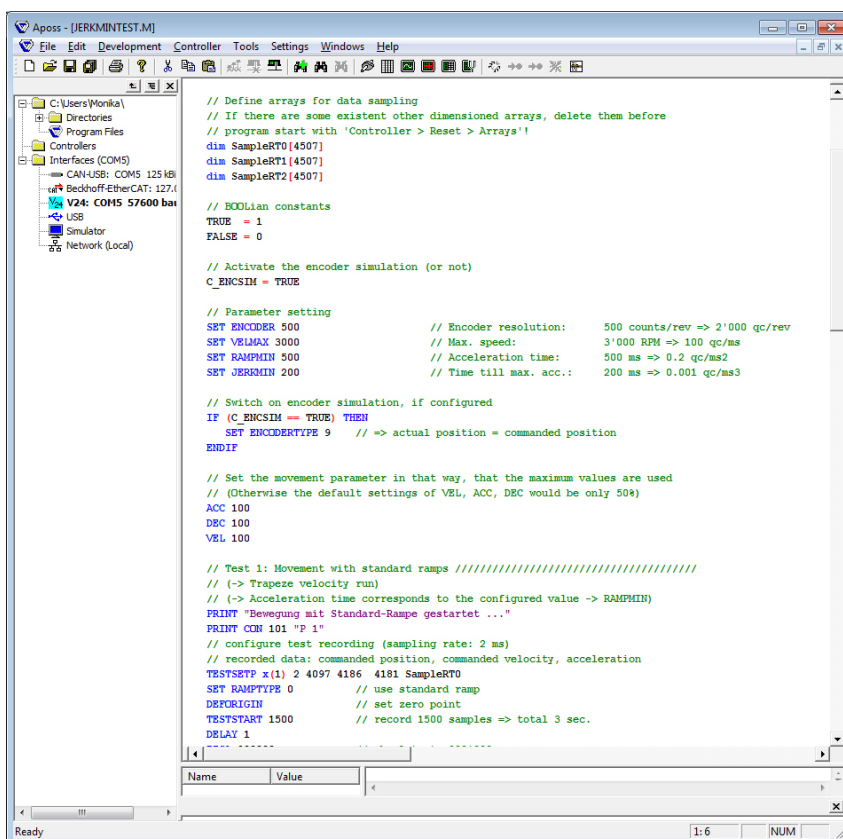
```

DIM marker[112] // Number of elements from the title bar of the CAM-Editor
HOME           // Slave axis performs a home run (switch for zero position on top)
              // Afterwards, the slave will be in the zero position (0 degrees)
              // (Omitted if an absolute encoder is used)

SETCURVE marker // Set stamp curve with marker
dist = GET SYNCMPULSM // Distance to sensor
DEFMCPPOS (1000-dist) // This is the location that corresponds to the sensor signal
SET SYNCMSTART 2000 // Counting of the master pulse does not begin
                  // until the next edge comes from the sensor

SYNCOM 0 // Synchronize in CAM-Mode until motor stop
SYNCSTART 1 // Engage platen with start point pair 1
           // Synchronous operation
WAITI 4 CN // Wait for input signal when conveyor belt is being switched off
SYNCSTOP 2 0 // Disengage platen with stop point pair 1 and stop at position 0 degrees
    
```

Sample:
Synchronise a belt with marker



```

// Define arrays for data sampling
// If there are some existent other dimensioned arrays, delete them before
// program start with 'Controller > Reset > Arrays!'
dim SamplerT0[4507]
dim SamplerT1[4507]
dim SamplerT2[4507]

// BOOLian constants
TRUE = 1
FALSE = 0

// Activate the encoder simulation (or not)
C_ENCDSIM = TRUE

// Parameter setting
SET ENCODER 500 // Encoder resolution: 500 counts/rev => 2'000 qc/rev
SET VELMAX 3000 // Max. speed: 3'000 RPM => 100 qc/ms
SET RAMPHIN 500 // Acceleration time: 500 ms => 0.2 qc/ms2
SET JERKMIN 200 // Time till max. acc.: 200 ms => 0.001 qc/ms3

// Switch on encoder simulation, if configured
IF (C_ENCDSIM == TRUE) THEN
  SET ENCODERTYPE 9 // => actual position = commanded position
ENDIF

// Set the movement parameter in that way, that the maximum values are used
// (Otherwise the default settings of VEL, ACC, DEC would be only 50%)
ACC 100
DEC 100
VEL 100

// Test 1: Movement with standard ramps //////////////////////////////////////
// (-> Trapeze velocity run)
// (-> Acceleration time corresponds to the configured value -> RAMPHIN)
PRINT "Bewegung mit Standard-Rampe gestartet ..."
PRINT CN 101 "P 1"

// configure test recording (sampling rate: 2 ms)
// recorded data: commanded position, commanded velocity, acceleration
TESTSETP x(1) 2 4097 4186 4181 SamplerT0
SET RAMPTYPE 0 // use standard ramp
DEPORIGIN // set zero point
TESTSTART 1500 // record 1500 samples => total 3 sec.
DELAY 1
    
```

Sample:
Test program for recording of
movements with different ramp
types of parameter RAMPTYPE.

Command Reference

All available APOSS commands are organized below into general functional groups:

Initialization	Commands to initialize the axes and the control unit start up and define the zero point(s).
Flow Control	Commands for controlling the program flow, and structuring the programs
Input/Output	Command to set and read inputs and outputs.
System Information	Commands for querying system status information such as drive position and velocity, system clock, error status, etc.
Interrupt Functions	Commands to enable and disable interrupts as well as to call subroutines.
Parameter	All global and axis parameters with a parameter code can be set and read with these commands.
Simple Move	Commands to set speed and velocity, to start, stop, and to position.
Control Loop	Turns on motor control on with user defined position.
CANopen Basic	This library contains all the functions necessary to do raw CAN telegram transfer and buffered messages.
CANopen Master	This library has functions to support the usage of CANopen Slaves like I/O modules or encoders or even drives. Those functions do not only allow the configuration with SDOs but do allow the usage of process data through PDOs.
CANopen Slave	This library contains all functions which are necessary to act as a CANopen Slave. With special application software, it would even be possible to behave like a DS402 drive.
EtherCAT Master	Commands to configure and control the EtherCAT masters, to read and set SDOs as well as SDO2 to communicate with debug devices like SDO monitors.
Synchronization	Commands to synchronize the slave with the master or the master simulation.
Curve Control	Commands for synchronization in CAM mode (CAM control).
Memory	Commands to define and save arrays.
Communication	Commands to handle the RS485 and RS233 interfaces as wells as the parameter handling of Danfoss VLTs.
Debug	Printing commands and commands for configuring and using arrays for recording test results.
State Machine	Commands related to state machines.
Deprecated	Commands are no longer supported.

Initialization Commands

Commands to initialize the axes and the control unit start up and define the zero point(s).

CANOPENSLAVE	Initializes the CANopen functionality with the set CAN number (id).
DEFCORIGIN	Sets the command position as real zero point.
DEFMORIGIN	Set the current master position as the zero point for the master.
DEFORIGIN	Sets the actual position as real zero point.
ERRCLR	Clears error.
HOME	Moves to machine zero point.
INDEX	Moves to next index position.
MOTOR	Turns the motor on and off.
RSTORIGIN	Reset temporary zero point.
SETMORIGIN	Set the current position as the zero point for the master.
SETORIGIN	Set temporary zero point.

Flow Control Commands

Commands for controlling the program flow, and structuring the programs.

BREAK	Break of command execution inside a WHILE or REPEAT loop, or a SWITCH statement.
CONTINUE	Continue the positioning from point of interruption, e.g. following a motor-stop.
DELAY	Time delay.
DELAYUS	Time delay in Microseconds.
EXIT	Desired premature program termination.
GOSUB	Calling up a subroutine.
GOTO	Jumping within a program.
IF THEN	Conditional program execution.
... ELSE ... IF ... THEN	... with conditional alternative branching.
... ELSE	... with alternative branching.
... ENDIF	End of the conditional program execution.
LOOP	Repeats loop.
NOWAIT ON/OFF	On/off switch for waiting for the command execution.
REPEAT	Beginning of repeat loop.
REPEAT ... UNTIL	Conditional loop, with an end criteria.
SUBMAINPROG	Start of the subroutine definition area.
... ENDPROG	End of the subroutine definition area.
SUBPROG	Begins a subroutine.

... RETURN	Ends a subroutine.
SWITCH	Conditional command execution, ...
... CASE	... command execution depending on the value of a variable.
... DEFAULT	... command execution without a fitting branch.
ENDSWITCH	... end of conditional command execution.
WAITAX	Wait until target position is reached.
WAITI	Wait for input.
WAITNDX	Wait until the next index position is reached.
WAITP	Wait until a certain position is reached.
WHILE ... DO	Conditional loop with commencement criteria.
... ENDWHILE	End of the loop.

Input/Output Commands

IN	Reads input bit-by-bit (individually).
INAD	Reads analog input and process data objects (PDO) of CAN objects.
INB	Reads input by bytes (units of 8).
OUT	Sets digital outputs bit-by-bit (single).
OUTAN	Sets analog output (allocated to an axis).
OUTB	Sets digital outputs by bytes (units of 8).
OUTDA	Sets analog output (freely available) and an output, which is initialized as a drive.

System Information Commands

Commands for querying system status information such as drive position and velocity, system clock, error status, etc.

APOS	Reads actual position.
APOSDIFF	Overflow handling of incremental encoders in applications.
AVEL	Queries actual velocity of an axis.
AXEND	Reads info on status of program execution.
CPOS	Reads set position.
CPOSDIFF	Overflow handling of incremental encoders in applications.
ERRAX	Number of the axis which triggers the error.
ERRNO	Reads error number.
_GETVEL	Changes sample time for AVEL and MAVEL .
IPOS	Queries last index or marker position of the slave.
IPOSDIFF	Overflow handling of incremental encoders in applications.

JERKFINVEL	Calculates the final velocity for a jerk-limited stop with maximum acceleration/deceleration.
JERKSTOPDIST	Calculates the necessary distance for a jerk-limited stop with maximum deceleration.
MAPOS	Queries actual position of the master.
MAPOSDIFF	Overflow handling of incremental encoders in applications.
MAVEL	Queries actual velocity of the master.
MIPOS	Queries last index or marker position of the master.
MIPOSDIFF	Overflow handling of incremental encoders in applications.
PID	Executes PID calculation.
STAT	Reads axis status.
STATCLR	Clears the slave and master index flags.
SYNCERR	Queries actual synchronization error of a slave axis.
SYSVAR	System variable (pseudo array) reads system values.
TIME	Reads internal controller time.
TIMEHW	Reads internal system timer.
TRACKERR	Queries actual position error of an axis.
USRSTAT	Sets user status.

Interrupt Functions

DISABLE ...	Locks the execution of interrupts.
ENABLE ...	Enables locked interrupts.
ON CANINPUT id GOSUB	Calls up a subprogram when a CAN telegram type 'id' arrives.
ON CANMSG GOSUB	Verifies arrival of a buffered CAN message.
ON COMBIT ... GOSUB	Calls up a subprogram when Bit n of the communication buffer is set.
ON DELETE .. GOSUB	Deletes a position-interrupt: ON posint GOSUB.
ON DELETE SETOUT	Deletes all interrupts which set or reset an output.
ON ERROR GOSUB	Calls subroutine in the event of an error.
ON .. GOSUB DELETE	Eliminates a single interrupt.
ON INT ... GOSUB	Calls subroutine depending on input signal.
ON KEYPRESSED GOSUB	Interrupts when a character appears via the serial interface when any key is pressed.
ON PARAM ... GOSUB	Calls up a subprogram when a parameter is altered.
ON PERIOD GOSUB	Calls subroutine at regular intervals.
ON posint ... GOSUB	Calls up a subprogram when a position interrupt occurs:

- ON APOS when the slave position xxx is passed.
- ON IPOS when the distance between the last marker position and the actual position is reached.
- ON MAPOS when the master position xxx [mu] is passed.
- ON MIPOS when the distance between two markers is reached.

ON posint ... GOSUBOT	Defines one time position interrupts.
ON posint SETOUT (TOIN)	Simulates a cam box (all types of position interrupts).
ON posint ... SETOUTOT	Defines one time position interrupts.
ON STATBIT ... GOSUB	Calls up a subprogram when bit n of the status word is set.
ON TIME ... GOSUB	Calls subroutine after single timing.

Parameter Commands

All global and axis parameters with a parameter code can be set and read with these commands.

GET	Reads parameter values (axis and global parameter).
LINKPDO	Link the system variable with PDO and copy in the internal parameters.
LINKSDO	Copying an internal object in the PDO.
SET	Sets parameter values (axis and global parameter).

Simple Move Commands

ACC	Sets acceleration.
DEC	Sets deceleration.
VEL	Sets velocity for relative and absolute motions and set maximum allowed velocity for synchronizing.
POSA	Positions axis absolutely.
POSR	Positioning relative to the actual position.
LINA	Synchronously positions axis absolutely.
LINR	Synchronized relative positioning of several axes.
CSTART	Starts the continuous movement in RPM mode.
CSTOP	Stops the drive in speed mode.
CVEL	Sets the velocity for speed regulation.
MOTOR STOP	Motor-stop with a programmed delay (with ramp).

Control Loop Commands

MOTOR USERREFPOS	Turns on motor control on with user defined reference position.
MOTOR USERREFVEL	Turns motor control on with user defined velocity reference

CANopen Basic Commands

CANDEL	Erases all or single CAN objects.
CANIN	Reads an object.
CANOPENSLAVE	Initializes the CANopen functionality with the set CAN number (id).
CANOUT	Sends message (active).
DEFCANIN	Defines a receive object.
DEFCANOUT	Defines a transmit object in the CAN controller.
INGLB	Reads Glb CAN message.
INMSG	Reads CAN message (polling).
MSGVAL	Supplies the long value of the last message.
ON CANINPUT	Calls up a subprogram when a CAN telegram type 'id' arrives.
OUTMSG	Sends CAN message (polling).
USRSTAT	Sets user status.

CANopen Master Commands

CANIN (>100)	Gathers all Transmit-PDOs of digital input modules or CAN drive status by using only one CAN telegram.
CANINI	Initializes the necessary objects (PDOs) for data exchange of CANopen nodes, or enables extended CANINI, CANIN function.
SDOREAD	Reads SDO of a connected CANopen device.
SDOREADSEG	Segmented read of SDOs (unpacked).
SDOREADSEGP	Segmented read of SDOs (packed).
SDOSTATE	Checks the result of an active communication.
SDOWRITE	Sets SDO of a connected CANopen device.
SDOWRITEN	Sets a SDO and specifies the number of valid data bytes (= "Data set size is indicated").
IN	Reads status of digital input.
INB	Reads one byte from digital inputs.
OUT	Set or re-set digital outputs.
OUTB	Alteration of the condition of a digital output byte.
SET ENCODERTYPE	SET parameter ENCODERTYPE.
SET DRIVETYPE	SET parameter DRIVETYPE.

CANopen Slave Commands

LINKPDO	Link the system variable with RxPDO and copy in the
---------	---

	internal parameters, or link part of an array into the PDO.
LINKSDO	Link TxPDO with internal system variable, or link part of an array out of the PDO.
PDO[]	Pseudo array for direct access to the CANopen PDOs.
SYSVAR[0x01...]	System variable (Pseudo array) reads system values; see SDO Dictionary .

EtherCAT Master Commands

ECATMASTERCOMMAND	Controls the EtherCAT master.
ECATMASTERCONFIG	Configures the EtherCAT master.
ECATMASTERINFO	Reads information from the EtherCAT master.
PDO[]	Pseudo array for direct access to the CANopen PDOs.
SDOREAD	Reads SDO of a connected CANopen device.
SDOREADSEG	Segmented read of SDOs (unpacked).
SDOREADSEGP	Segmented read of SDOs (packed).
SDOSTATE	Checks the result of an active communication.
SDOWRITE	Sets SDO of a connected CANopen device.
SDOWRITEN	Sets a SDO and specifies the number of valid data bytes (= "Data set size is indicated").

Synchronization Commands

Commands to synchronize the slave with the master or the master simulation.

DEFMORIGIN	Set the current master position as the zero point for the master.
DEFSYNCORIGIN	Defines master-slave relation for the next SYNCP or SYNCM command.
MOVESYNCORIGIN	Relative shifting of the origin of synchronization.
PULSACC	Sets acceleration for the virtual master.
PULSVEL	Sets velocity for virtual master.
SETMORIGIN	Set the current position as the zero point for the master.
SYNCM	Synchronization of angle/position with marker correction.
SYNCMARKERSTART	Resets a marker or resets marker handling.
SYNCP	Synchronization of angle/position.
SYNCSTAT	Queries flag for synchronization status.
SYNCSTATCLR	Resetting of the flags MERR and MHIT.
SYNCV	Synchronization of velocity.
ACC	Sets acceleration.
DEC	Sets deceleration.

VEL Sets velocity for relative and absolute motions and set maximum allowed velocity for synchronizing.

Curve Control Commands

Commands for synchronization in CAM mode (CAM control).

CURVEPOS Retrieve slave curve position that corresponds to the current master position of the curve.

DEFMCPOS Define initial position of the master.

POSA CURVEPOS Move slave to the curve position corresponding to the master position.

SETCURVE Sets CAM curve.

SYNCC Synchronization in CAM mode.

SYNCCMM Synchronization in CAM mode with master marker correction.

SYNCCMS Synchronization in CAM mode with slave marker correction.

SYNCCSTART Starts slave for synchronization in CAM mode.

SYNCCSTOP Stops slave after the CAM synchronization.

ACC Sets acceleration.

DEC Sets deceleration.

VEL Sets velocity for relative and absolute motions and set maximum allowed velocity for synchronizing.

Memory Commands

CAMDIM Define a global CAM array.

DELETE ARRAYS Delete all arrays in the RAM.

DIM Define a global array.

DIMARRAY Pass a DIM array to functions by referencing the array number of the DIM array rather than the array name.

MEMORYDUMP Save SDO data on the Micro-SD card.

MEMORYDUMPSTATE This command can be used to poll the state/result of a MEMORDUMP command.

SAVE ARRAYS Save arrays in the flash memory permanently.

SAVE AXPARS Save current axis parameters in the flash memory permanently.

SAVE GLBPARS Save current global parameters in the flash memory permanently.

SAVEPROM Save RAM in the flash memory permanently.

Communication Commands

DCPINIT	Initializes the RS485 (MACS4) or RS232/RS485 (MACS5) interface handler.
DCPREADCOMM	DCP command for MACS4 RS485 or MACS5 RS232 interface: Checks for an incoming telegram.
DCPWRITECOMM	DCP command for MACS4 RS485 or MACS5 RS232 interface: Sends telegram.
INKEY	Reads ASCII-character from interface (RS232).
VLTCOMMAND	Allows control word and references to be sent to Danfoss VLT via RS485.
VLTCOMSTATE	Polls for the result of the last parameter transaction.
VLTPREAD	Reads parameters from a Danfoss VLT via RS485.
VLTPSTATUS	Reads status and actual speed of Danfoss VLT over RS485.
VLTPWRITE	Writes parameters to a Danfoss VLT over RS485.

Debug Commands

Printing commands and commands for configuring and using arrays for recording test results.

PRINT	Output (display) texts and variables.
PRINTDEV	Stops information output.
PRINTF	Formatted information output.
SPRINTF	Formatted information output to a string array.
TESTSETDEST	Defines the memory section for saving recorded data.
TESTSETINDEX	Specifies system variables for data recording.
TESTSETTIME	Defines the sampling period for data recording.
TESTSETTYPE	Defines "one time" or "cyclic" recording.
TESTSTART	Starts the data recording.
TESTSTOP	Stops the data recording.

State Machine Commands

SmClear	Clear the event queue.
SmConfig	Define the state machine configuration parameters.
SmEvent	Define an event and its parameters.
SmInput	Request an event when a digital input changes.
SmMachine	Define a state machine variable.
SmParam	Request an event when a parameter value changes.
SmPeriod	Start a periodic timer.
SmPosition	Request an event when the motor passes a specified position.
SmPositionOnce	Request an event when the motor passes a specified

	position for the first time.
SmPost	Create an event and post it to the end of the event queue.
SmPublish	Create an event and publish for all state machines.
SmRun	Start state machines running.
SmState	Define a state machine state, including its event handlers and substates.
SmStop	Stop running state machines.
SmSubscribe	Subscribe to receive published events.
SmTimer	Start a "once only" timer.
SmUnsubscribe	Request to stop receiving published events.
SmUrgent	Create an event and post it to the head of the event queue.

Deprecated Commands

#DEBUG	Command up to firmware version < 5.22
SWAPMENC	Encoder assignment
TESTSETP	Specifies recording data for test run.
WAITT	Time delay in milliseconds.
GETSYS/SETSYS	Save and read userdefined SYSVARS for controls up to MACS4. From MACS5 controls onwards the still necessary functions have been implemented in SDO 0x2203.