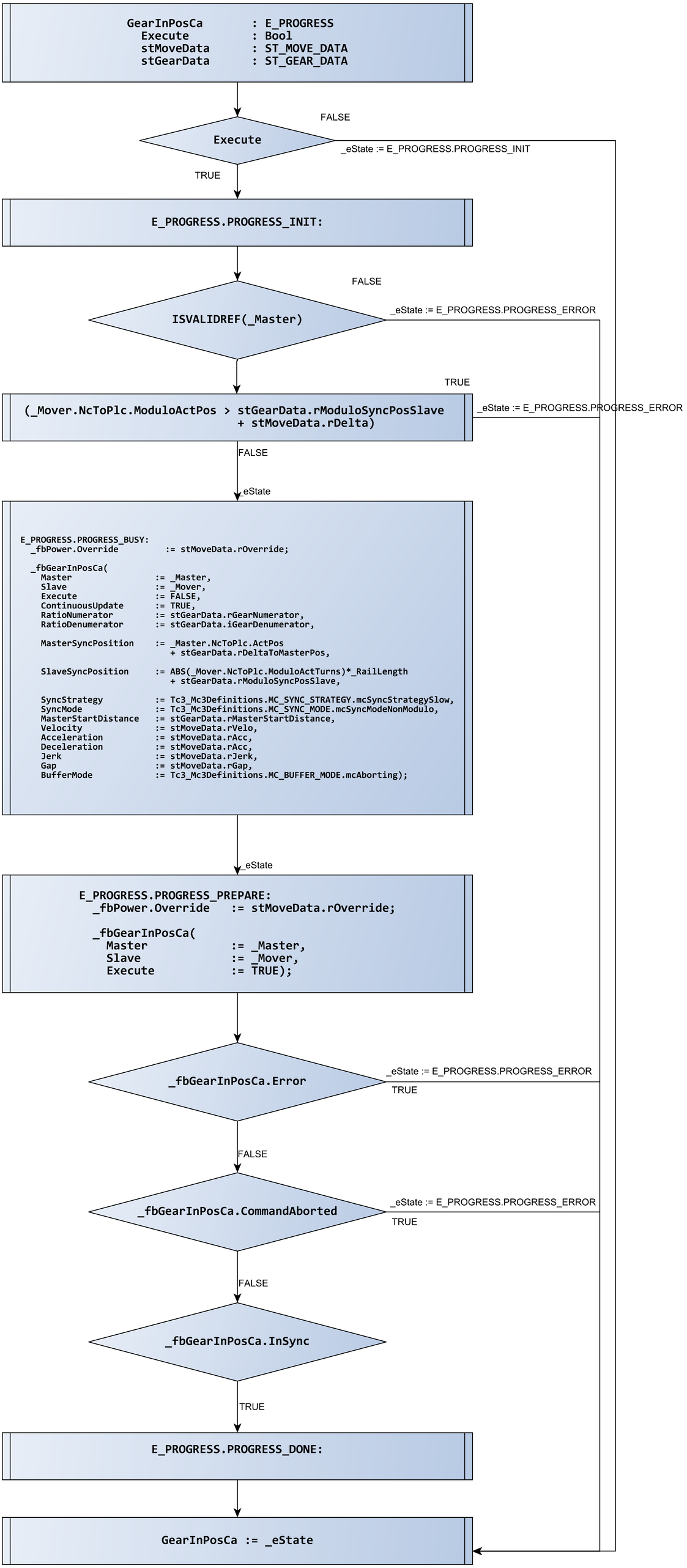
GearInPosCa : E_PROGRESS
Execute : Bool
stMoveData : ST_MOVE_DATA
stGearData : ST_GEAR_DATA
Execute
FALSE
_eState := E_PROGRESS.PROGRESS_INIT
TRUE
E_PROGRESS.PROGRESS_INIT:
ISVALIDREF(_Master)
FALSE
_eState := E_PROGRESS.PROGRESS_ERROR
(_Mover.NcToPlc.ModuloActPos > stGearData.rModuloSyncPosSlave + stMoveData.rDelta)
TRUE
_eState := E_PROGRESS.PROGRESS_ERROR
FALSE
eState
E_PROGRESS.PROGRESS_BUSY:
_fbPower.Override := stMoveData.rOverride;
_fbGearInPosCa(
Master := _Master,
Slave := _Mover,
Execute := FALSE,
ContinuousUpdate := TRUE,
RatioNumerator := stGearData.rGearNumerator,
RatioDenumerator := stGearData.iGearDenumerator,
MasterSyncPosition := _Master.NcToPlc.ActPos + stGearData.rDeltaToMasterPos,
SlaveSyncPosition := ABS(_Mover.NcToPlc.ModuloActTurns)*_RailLength + stGearData.rModuloSyncPosSlave,
SyncStrategy := Tc3_Mc3Definitions.MC_SYNC_STRATEGY.mcSyncStrategySlow,
SyncMode := Tc3_Mc3Definitions.MC_SYNC_MODE.mcSyncModeNonModulo,
MasterStartDistance := stGearData.rMasterStartDistance,
Velocity := stMoveData.rVelo,
Acceleration := stMoveData.rAcc,
Deceleration := stMoveData.rAcc,
Jerk := stMoveData.rJerk,
Gap := stMoveData.rGap,
BufferMode := Tc3_Mc3Definitions.MC_BUFFER_MODE.mcAborting);
_eState
E_PROGRESS.PROGRESS_PREPARE:
_fbPower.Override := stMoveData.rOverride;
_fbGearInPosCa(
Master := _Master,
Slave := _Mover,
Execute := TRUE);
_fbGearInPosCa.Error
_eState := E_PROGRESS.PROGRESS_ERROR
TRUE
FALSE
_fbGearInPosCa.CommandAborted
_eState := E_PROGRESS.PROGRESS_ERROR
TRUE
FALSE
_fbGearInPosCa.InSync
TRUE
E_PROGRESS.PROGRESS_DONE:
GearInPosCa := _eState