

Yes, there are several tools available to monitor and troubleshoot the operation of the function blocks. The function block operation is mainly monitored via RDA area (MW00100-MW03199, 100 offset per axis, or MW03200-MW03999, 50 offset per Master/Slave area).

Function Block RDA Memory Map - Summary			
MW00000			
↑	Reserved Data Area (RDA) Global RDA data		↑
↓			
MW00099			
MW00100			
↑	Reserved Data Area (RDA) Axis #1 data		RDA Area
↓			
MW00199			
MW00200			
↑	Reserved Data Area (RDA) Axis #2..#31 data		↓
↓			
MW03199			
MW03200			
↑	Reserved Data Area (RDA) Master/Slave Pair #1 data		
↓			
MW03249			
MW03250			
↑	Reserved Data Area (RDA) Master/Slave Pair #2..#10 data		
↓			
MW03999			

The following is a list of conditions and their related registers:

**1. Function block/Motion Error:**

A. Refer to [FAQ: MTN-5WKP9C - What are the recommended programming practices when using function blocks?](#)

B. When the error bit is turned ON in the function block, it is often useful to check Error ID1 register, MW\*\*81 (\*\* is the axis number. Ex. MW0381 contains the error status of axis 3). These RDA registers indicates the classification of the block error when the block error output turns on. The following is the register bit description:

- b0: No Home Switch
- b1: Over Travel
- b2: Time Out
- b3: Value to Great
- b4: Direction Not Allowed
- b5: Pos Error
- b6: No Motor Power
- b7: Servo Alarm
- b8: Clear Pending error
- b9: Track Fail
- bA: RDA error
- bB: Error Stop
- bC: Table Error

C. When unexpected outcome of a move request is occurred, the axis state, the move command status, and the move command type can be checked for possible cause. Register MW\*\*49 indicates the axis state as an integer:

- 1: Error Stopped Motion
- 2: Stopped Motion
- 3: Standstill
- 4: Discrete Motion
- 5: Continuous Motion

Register MW\*\*80 indicates the status of the commanded move as an integer:

- 0: Running
- 1: Faulted
- 2: Aborted

Register MW\*\*78 indicates the last commanded move type as an integer:

- 0: Stop
- 1: Jog
- 2: Move Relative
- 3: Move Absolute
- 4: Move Continuous
- 5: Home
- 6: Latch
- 7: Gear
- 8: CAM

D. In case of a programming error that causes a move function to continuously be executed and consequently disallowing other functions from running, Bit MB\*\*702 should be monitored. This bit holds one scan pulse of motion block execution, if the motion block is continuously executed, this bit will be ON, indicating a programming error.

## **2. Tracing Function Block Execution:**

Blocks running can be traced via MW\*\*79 register, which indicates the number of the block running as an integer value (1-28). This register is particularly useful in determining when an error has occurred or which function block caused the fault.

Block # ID	Single Axis Function Block Name	Multi Axis Function Block Name SVA	Multi Axis Function Block Name SVB	Multi Axis Function Block Name SVB
1	MOVABS	AMOVABS	BMOVABS	Absolute index move
2	MOVINC	AMOVINC	BMOVINC	Relative index move
3	MOVCON	AMOVCON	BMOVCON	Continuous interpolated index move
4	MOVVEL	AMOVVEL	BMOVVEL	Jog
5	HOME	AHOME	BHOME	Home
6	CHNG_DYN	ACHNG_DY	BCHNG_DY	Change dynamics Accel/Decel/Spd change on fly
7	STOP	ASTOP	BSTOP	Stop all motion blocks
8	SVON	ASVON	BSVON	Servo Enable
9	SCLRPNDG	CLRPNDG*	CLRPNDG*	Clear pending axis cmd actions of move continuous
10	RDERROR	x	BRDERROR	Fault monitoring
11	RDPARAM	RDPARAM*	RDPARAM*	Read parameter from table
12	WRTPARAM	WRTPARAM*	WRTPARAM*	Write parameter to table
13	GEAR	AGEAR	BGEAR	Position based gearing
14	CAM	ACAM	BCAM	Position based Camming
15	MOD_ENG*	MOD_ENG*	MOD_ENG*	Modulus engine (saw tooth)
16	SCAMOFF	CAMOFFST*	CAMOFFST*	CAM shift operation
17	SSLAVOFF	SLAVOFF*	SLAVOFF*	Slave offset operation for CAM
18	SCAMSCAL	CAMSCALE*	CAMSCALE*	CAM scale operation
19	PLS*	PLS*	PLS*	Programmable Limit Switch
20	LATCH	ALATCH	BLATCH	Simple latch enable/disable
21	LTCHTRGT	ALTCHTGT	BLTCHTGT	Registration move to target
22	ASCII-IN*	ASCII-IN*	ASCII-IN*	Ascii communications input
23	ASCIIOUT*	ASCIIOUT*	ASCIIOUT*	Ascii communications output
24	ALMRST	AALMRST	BALMRST	Reset of alarms
25	SRDANIT1	RDANIT1*	RDANIT1*	Initialize RDA part 1
26	SRDANIT2	RDANIT2*	RDANIT2*	Initialize RDA part 2
27	x	x	BTUNING	Network Servo tuning gain set
28	x	AAXISEND	BAXISEND	Axis scan end block

### 3. Monitoring Slaving conditions:

When Gearing or camming is used in the application, it is often useful to monitor the state of the slave axis, especially in the programming phase of the project. The gearing and the camming state is indicated in registers MW3\*\*0 and MW3\*\*2, respectively (\*\* is the slave/master pair computation area. MW3302 contains the cam status of master/slave pair in the 3rd computation area).

Register MW3\*\*0 indicates the operational state of the gear function:

- b0: Disengaged
- b1: Accelerating
- b2: Gearing Locked
- b3: Decelerating

Register MW3\*\*2, indicates the operational state of the cam function

- b0: Disengaged
- b1: Waiting to Engage
- b2: Camming Locked
- b3: Waiting to disengage