EtherCAT : Errata for Industrial SDK 1.1.0.4

For application developers: Known Issues/Limitations

Single datagram accessing multiple FMMU mapped areas using LRD/LWR commands

- Issue/ Failure Description or state
  - SDOCM00092510 : Single datagram accessing multiple FMMU mapped areas in a single slave will only update the data corresponding to first FMMU in datagram
- Conditions in which failures occur
  - Single datagram accessing multiple FMMU mapped areas in single slave.

    FMMU0(0x1000:0x1007)->SM2 (Write SM)

    FMMU1(0x1000:0x1007)->SM3 (Read SM)

    FMMU2(0x1008:0x100F)->SM4 (Write SM)

    FMMU3(0x1008:0x100F)->SM5 (Read SM)
    - Single LRD to access (0x1000:100F) will only access SM3
    - Single LWR to access (0x1000:100F) will only access SM2
    - 2 LRD/LWRs are required in this case to access both pair of SMs - LRD1/LWR1(0x1000:0x1007) and LRD2/LWR2(0x1008:0x100F)
- Root cause
  - Increased code memory requirements in firmware to implement this support
- Work around
  - For above example instead of one LRD datagram to logical address 0x1000 and length 16, master needs to send two LRD datagrams, one to logical address 0x1000 and length 8 second to logical address 0x1008 and length 8 or use LRW in place of LRD/LWR which is more efficient for most of the use cases

NOTE: This issue will not be fixed

LRW access to non-interleaved input and output process data of multiple slaves does not work

- Issue/ Failure Description or state
  - SDOCM00105048: LRW access to non-interleaved input and output process data of multiple slaves does not work. SOEM accesses slaves in LRW mode this way
- Conditions in which failures occur
  - Single LRW datagram accessing FMMU mapped areas in multiple slaves and PD out is mapped
    FMMU0(0x1000:0x1007)->SM2 #1(Write SM) FMMU1(0x1008:0x100F)->SM2#2 (Write SM)
    FMMU2(0x1010:0x1017)->SM3 #1 (Read SM) FMMU3(0x1018:0x101F)->SM3#2(Read SM)
    - Single LRW access from (0x1000:101F)
- Root cause
  - Increased code memory requirements in firmware to implement this support as well as non-interleaved access I/O data is not a very optimal use of EtherCAT – it increases the cycle time overhead/datagram size and not effective use of LRW datagram which can perform read and write in the same cycle.
- Work around
  - Use LRD/LWR datagram to access process data
  - Use LRW datagram to access process data
    - Input and output overlaid on the same logical address range (TwinCAT usage)
    - Input and output of a given slave back to back in logical address space
      - FMMU0(0x1000:0x1007)->SM2 #1(Write SM) FMMU1(0x1008:0x100F)->SM3#1 (Read SM)
      - FMMU2(0x1010:0x1017)->SM2 #2 (Write SM) FMMU3(0x1018:0x101F)->SM3#2 (Read SM)

NOTE: This issue will not be fixed

SYNC0 jitter increases with distance of slave from reference

- Issue/ Failure Description or state
  - SDOCM00097012 : SYNC0 jitter increases with distance of slave from reference
- Conditions in which failures occur
  - TI ESC in a large network of slaves and away from reference slave shows variable jitter which increases with distance of the slave
- Root cause
  - Drift compensation algorithm in firmware requires optimization

NOTE: This issue will be addressed in subsequent releases

LRD access on unused registers

- Issue/ Failure Description or state
  - SDOCM00098950: LRD access on unused registers results in WKC increment
- Conditions in which failures occur
  - LRD access on unused registers result in WKC increment
- Root cause
  - Firmware does not support register protection in LRD mode at this moment, it requires more firmware footprint to support, this minor spec compliance does not justify the footprint increase and no Write Only registers in ESC

NOTE: This issue will not be fixed

PD/PDI watchdog counter issue

- Issue/ Failure Description or state
  - SDOCM00098105: PDI/PD watchdog counter incremented by 1 whenever PDI/PD watchdog is disabled using EtherCAT master
- Conditions in which failures occur
  - Whenever EtherCAT master disables WD by writing zero to respective Watchdog Time registers (0x410:0x411 or 0x420:0x421)
- Root cause
  - This is PRU-ICSS h/w behavior

NOTE: This issue will be not be fixed

Rx Time Port1 (0x904) time stamping not done correctly when write to 0x900 is done cyclically

- Issue/ Failure Description or state
  - SDOCM00108541: Rx Time Port1 (0x904) time stamping is missed when write (BWR) to 0x900 is done cyclically
- Conditions in which failures occur
  - When EtherCAT master cyclically write to 0x900 register and two ports of EtherCAT slave is active
- Root cause
  - Firmware monitors writes to 0x900, misses to update Port1 time stamp under increased traffic conditions

NOTE: Fix is under evaluation

Codesys master reads back wrong EEPROM data when scan devices is done continuously

- Issue/ Failure Description or state
  - SDOCM00108191: Codesys master reads back wrong EEPROM data when scan devices is done continuously
- Conditions in which failures occur
  - From Codesys master, When we scan for devices continuously, The vendor code, Product code and revision number was getting wrong values randomly.
- Root cause
  - Fixed a firmware issue while indicating EEPROM CTRL(0x503) register write from Master to A8 stack, firmware gave an incorrect indication to A8 on a write to EEPROM ADDR(0x504). This caused original write to 0x503 to be missed depending on the timing and frequency of reads from Master etc. TwinCAT and CTT did not have this problem as they did not write to 0x504 during EEPROM operations

NOTE: This issue is fixed in 1.1.0.5

PHY address range supports only values from 0 to 7

- Issue/ Failure Description or state
  - SDOCM00108358: PHY address range supports only values from 0 to 7
- Conditions in which failures occur
  - Depending upon the use case, PHY address can vary from 0-31
  - The PHY address can be > 7 depending SYSBOOT bootstrap resisters on the AM3357
    - If h/w prototype does not have the ability to disconnect SYSBOOT resistors while PHY is booting
- Root cause
  - Fixed EtherCAT firmware to use 32-bit mask for MDIO LINK status monitoring

NOTE: This issue is fixed in 1.1.0.5

APWR to 0x900 updates Rx Time Port 0/1 time stamp

- Issue/ Failure Description or state
  - SDOCM00108540: APWR to 0x900 updates Rx Time Port 0/1 time stamp
- Conditions in which failures occur
  - Do APWR to 0x900 from EtherCAT master
  - Monitor Rx Time Port 0/1 time stamp registers (0x900 and 0x904)
- Root cause
  - EtherCAT IP core spec was unclear regarding APWR to 0x900, Beckhoff clarified that only BWR and FPWR (if node address matches) to 0x900 updates Rx Time Port time stamp register

NOTE: This issue is fixed in 1.1.0.5

Interrupt and watchdog trigger are not generated for single-byte process data SM

- Issue/ Failure Description or state
  - SDOCM00112753: Interrupt and watchdog trigger are not generated for single-byte process data SM
- Conditions in which failures occur
  - When EtherCAT Process Data Output length is set to 1 byte and slave transitions from INIT to OP
- Root cause
  - Firmware disables SyncManager when length is equal to 1. Interrupt and WD trigger is not generated as there is no SM

NOTE: This issue is fixed in 1.1.0.5

TI ESC go out of sync with Acontis master in DC synchronization mode

- Issue/ Failure Description or state
  - SDOCM00112947: TI ESC go out of sync with Acontis master in DC synchronization mode
- Conditions in which failures occur
  - Acontis EtherCAT master in DC sync mode (uses PC time base) connected to TI ESC forming EtherCAT slave network daisy chain
  - Easier to reproduce when control system is loaded
- Root cause
  - Drift compensation logic is not able to catch up fast enough when reference clock is jittery

NOTE: This issue is fixed in 1.1.0.5

For stack integrators: Known Issues/Limitations

TwinCAT can't detect EtherCAT slave when the NIC is forced to 100 Mbps Full duplex mode

- Issue/ Failure Description or state
  - SDOCM00113197: TwinCAT can't detect EtherCAT slave when the NIC is forced to 100 Mbps Full duplex mode
- Conditions in which failures occur
  - Force Ethernet NIC in 100 Mbps FD mode and scan and connect to slaves
- Root cause
  - TLK110 configuration issue, when link partner is in forced 100 Mbps Full Duplex mode, TLK110 negotiates 100Mbps Half Duplex link by default

NOTE: This issue is fixed in 1.1.0.5