

AN15164

Tamagawa T-format Encoder Host Interface using FlexIO on MCX A366

Rev. 1.0 — 28 September 2026

Application note

Document information

Information	Content
Keywords	AN15164, Tamagawa T-format Encoder, FlexIO, FRDM-MCXA366
Abstract	This application note describes a FlexIO-based Tamagawa T-format host implemented on the FRDM-MCXA366.



1 Overview

This application note describes a FlexIO-based Tamagawa T-format host implemented on the FRDM-MCXA366.

The project performs the following operations:

- Drives the request data (TXD)
- Holds the RS-485 driver-enable signal (DIR) for the exact request duration
- Captures the encoder response data (RXD)
- Validates the returned frame
- Publishes the position and status

The Encoder TS5700N8501 has the following characteristics:

- 17-bit single-turn absolute (ABS) data
- 16-bit multi-turn absolute (ABM) data
- 2.5 Mbit/s differential NRZ communication
- DID3 full-position transaction support

The current source tree is a polling-transfer example.

1.1 Key features

The application provides the following key features for implementing reliable T-format communication using FlexIO:

- FlexIO0 D17 generates TXD, D18 receives RXD, and D16 drives external RS-485 DE/DIR.
- DID3 (CF = 0x1A) returns ABS, encoder ID, ABM, ALMC, SF, and CRC in an 11-byte response.
- One TX shifter, one RX shifter, and three timers implement TX, RX, and direction control.
- The parser checks echoed CF and CRC8, masks ABS to 17 bits, sign-extends ABM from 16 bits, and reports multi-turn validity.
- Polling transfers use bounded retry loops and restore FlexIO state after timeout.

1.2 Scope

This application note documents the supplied MCX A366 polling example and TS5700N8501 profile. It is not a replacement for the encoder specification or a production electrical design. Transceiver selection, cable, supply sequencing, termination, EMC, and reset safety remain system responsibilities.

2 Fundamentals

T-format is a bidirectional asynchronous serial interface. The host transmits a request while its differential driver is enabled, then releases the bus before the encoder sends its response. Each byte uses an idle high, one low-start bit, eight LSB-first data bits, and one high-stop bit.

2.1 T-format physical interface

The TS5700N8501 specification identifies a differential NRZ interface compatible with RS-485. MCX A366 pins are logic-side signals only. Connect D17, D18, and D16 through a direction-controlled transceiver; do not connect MCU pins directly to the encoder SD/DS pair.

Table 1. T-format signal directions

Signal	Function	Direction in this project
TXD	T-format request	MCX A366 to transceiver
RXD	Encoder response	Transceiver to MCXA366
DIR	RS-485 driver enable	MCX A366 to transceiver
SD/DS	Differential serial pair	Transceiver to encoder
GND	Logic reference	Common ground

2.2 DID3 position transaction

DID3 requests absolute single-turn position, encoder ID, multi-turn position, and alarm information. CF is 0x1A and the one-byte request has no request CRC. The encoder starts its response approximately 3 μ s after the request delimiter typically; DIR must already have released the driver.

[Figure 1](#) shows the generic Smart-Abs readout transaction. The host sends only the control field for a position request. The encoder echoes CF and then returns the status, the selected data bytes, and the response CRC.

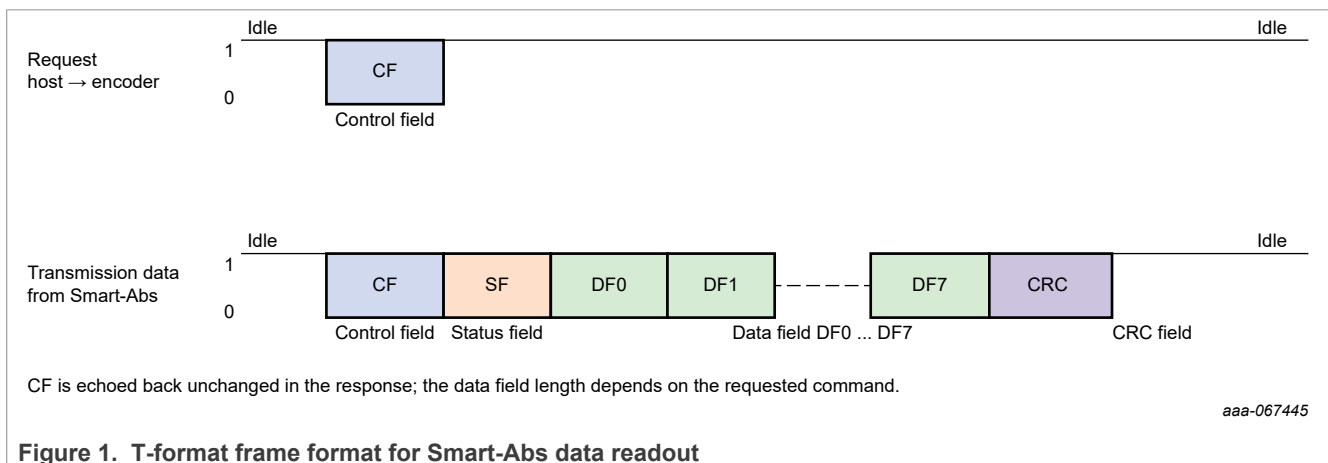


Figure 1. T-format frame format for Smart-Abs data readout

2.3 Frame and CRC8

The DID3 response has eleven bytes: CF, SF, ABS0...2, ENID, ABM0...2, ALMC, and CRC. ABS and ABM are little-endian 24-bit containers; only ABS[16:0] and ABM[15:0] are valid for this encoder. CRC uses $G(x) = x^8 + 1$; the source verifies a complete received response by expecting zero remainder.

Table 2. DID3 response fields

Byte	Field	Parser treatment
0	CF	Must echo 0x1A
1	SF	Encoder and communication status
2..4	ABS	Little-endian; mask to 17 bits
5	ENID	TS5700N8501: 0x11
6..8	ABM	Little-endian; mask to 16 bits; sign extend
9	ALMC	Detailed encoder alarm bits
10	CRC	CRC over received response

2.4 Role of FlexIO

FlexIO supplies the transmit shifter, receive shifter, TX baud timer, RX baud timer, and DE-gate timer. The Cortex-M33 composes requests, starts blocking transfers, verifies frames, and performs serial diagnostics and polling delays.

2.5 TS5700N8501 profile notes

The reference encoder runs at 2.5 Mbit/s and supports 17-bit single-turn and 16-bit multi-turn data. A backup battery is required for multi-turn retention through main-power loss. A valid frame can carry a battery alarm/error in SF or ALMC; that encoder state is not itself a link failure.

2.6 Control and data fields

The request begins with a Control Field (CF). The encoder specification defines a fixed start pattern and sink code. It is followed by a four-bit Data ID code, its parity bit, and a fixed delimiter. The source builds CF from the fixed sink code plus the selected data ID and parity. As a result, command definitions are kept centralized and a read command is prevented from being confused with a configuration or reset command.

The extracted timing diagram in [Figure 2](#) shows the 10-bit control field. Its fields, originally presented as part of the specification graphic, are reproduced below as editable content.

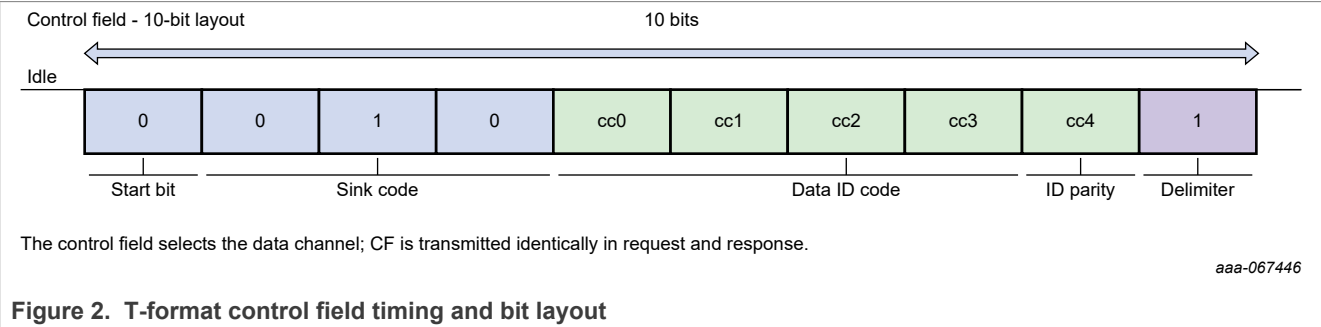


Figure 2. T-format control field timing and bit layout

Table 3. Control-field component definition

Component	Width	Definition
Start bit	1 bit	Fixed start bit; the line transitions from idle high to start the request.
Sink code	3 bits	Fixed Smart-Abs sink code.
Data ID code	4 bits (cc0..cc3)	Selects the requested data set or service operation.
ID parity	1 bit (cc4)	Parity associated with the four-bit Data ID code.
Delimiter	Fixed	Marks the end of the control field before the encoder response.

Table 4. Control-field data ID coding

Application	Data ID	cc0...cc3 (transmitted order)	Parity (cc4)
Data readout	0	0 0 0 0	0
Data readout	1	1 0 0 0	1

Table 4. Control-field data ID coding...continued

Application	Data ID	cc0...cc3 (transmitted order)	Parity (cc4)
Data readout	2	0 1 0 0	1
Data readout	3	1 1 0 0	0
Writing to EEPROM	6	0 1 1 0	0
Readout from EEPROM	D	1 0 1 1	1
Reset	7	1 1 1 0	1
Reset	8	0 0 0 1	1
Reset	C	0 0 1 1	0

For ordinary data acquisition:

- Select DID0 for single-turn ABS only
- DID1 for multi-turn ABM only
- DID2 for encoder ID
- DID3 for the complete ABS/ENID/ABM/ALMC data set

The application uses DID2 during initialization and DID3 for the periodic position poll. DID3 is preferred for diagnostic commissioning because it returns both position domains and detailed alarm state in a single response.

Table 5. T-format Data ID selection

Data ID	CF	Returned data	Normal use
0	0x02	ABS0...ABS2	Single-turn-only acquisition
1	0x8A	ABM0...ABM2	Multi-turn-only acquisition
2	0x92	ENID	Identity check during the startup
3	0x1A	ABS0...2, ENID, ABM0...2, ALMC	Normal full-position acquisition
6	0x32	EEPROM write acknowledgment	Service/configuration only
D	0xEA	EEPROM read the response	Service/configuration only
7	0xBA	Clear latch errors	Controlled service action
8	0xC2	Set single-turn zero	Controlled service action
C	0x62	Reset ABM and all errors	Controlled service action

The encoder transports ABS and ABM in three data bytes, the least significant byte first. For the TS5700N8501, ABS2 uses only bit 0; the upper seven bits of the 24-bit ABS container are zero. ABM is similarly transported in three bytes but only its lower 16 bits are valid. Therefore, the application must first perform explicit little-endian assembly. It must then mask the fields rather than treating the 24-bit transport width as the encoder resolution.

Table 6. DID3 data-field interpretation

Field	Transport order	TS5700N8501 meaning	Application treatment
ABS0...ABS2	Little-endian 24-bit	17-bit single-turn binary value	Mask with 0x0001FFFF
ENID	One byte	Fixed encoder identifier; 0x11 for this model	Check at startup
ABM0...ABM2	Little-endian 24-bit	16-bit multi-turn binary count	Mask with 0x0000FFFF; sign extend in current API
ALMC	One byte	Detailed alarm field	Use with SF to determine data validity

Table 6. DID3 data-field interpretation...continued

Field	Transport order	TS5700N8501 meaning	Application treatment
CRC	One byte	CRC8 for all non-CRC fields	Reject frame if the full-frame remainder is non-zero

The data-field and ALMC information are shown in Table 7. The DID3 frame used by this project is the complete position-and-diagnostic selection.

Table 7. Data field mapping by Data ID

Data ID	DF0	DF1	DF2	DF3	DF4	DF5	DF6	DF7
0	ABS0	ABS1	ABS2					
1	ABM0	ABM1	ABM2					
2	ENID							
3	ABS0	ABS1	ABS2	ENID	ABM0	ABM1	ABM2	ALMC
7	ABS0	ABS1	ABS2					
8	ABS0	ABS1	ABS2					
C	ABS0	ABS1	ABS2					

A blank field in the table means that no data byte is transmitted for that Data ID. ABS0...ABS2 represent absolute single-turn data; ABS0 is the least significant byte and ABS2 is the most-significant byte. ABM0...ABM2 use the same byte order for multi-turn data. ENID is the fixed encoder identifier, and ALMC is the encoder alarm byte.

Table 8. ALMC bit definitions

Bit	Symbol	Condition when set
d0	OS	Overspeed
d1	FS	Full absolute status
d2	CE	Counting error
d3	OF	Counter overflow
d4	OH	Overheat
d5	ME	Multi-turn error
d6	BE	Battery error
d7	BA	Battery alarm

The physical frame has no explicit end-of-frame field. After a valid start and delimiter, the encoder detects the end when no further start bit appears. The idle state is logic high. Therefore, the host maintains an idle-high TXD condition after the stop bit. It must not issue a second request while the encoder is still driving its response.

2.7 EEPROM and reset transactions

T-format also supports EEPROM access and reset requests. EEPROM user data occupies addresses from 0 to 126 on pages from 0 to 5. Address 127 selects a page. The default page is page 0 after the main power on.

The example reads temperature using page 7, address 5; it does not need user EEPROM writes for normal position acquisition.

Figure 3 shows the EEPROM readout transaction. Unlike the ordinary position readout, the request includes a control field, address field, and CRC. The response echoes the requested control/address information and returns EEPROM data plus CRC.

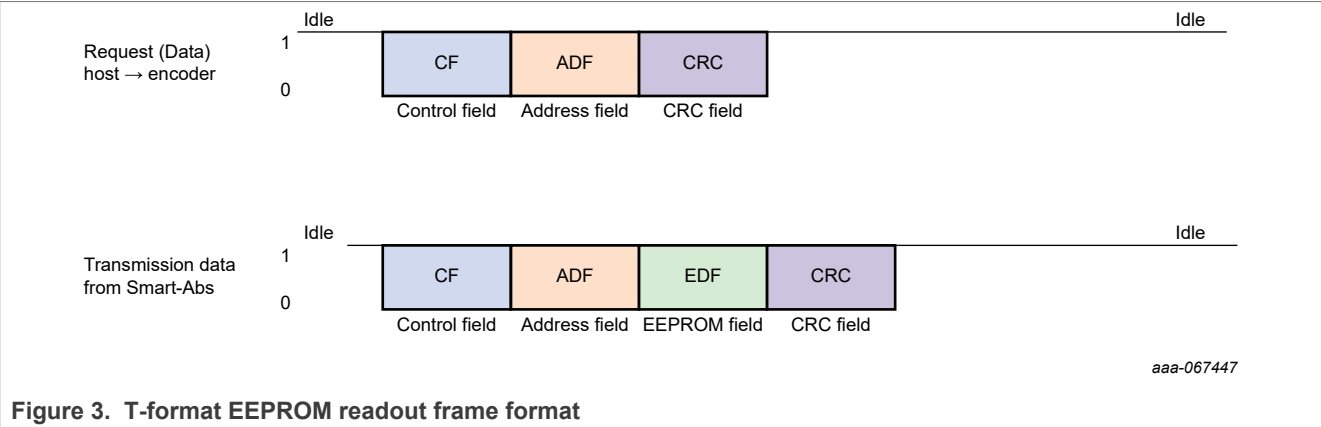


Figure 3. T-format EEPROM readout frame format

3 Hardware pinout and wiring

Figure 4 shows the required logic-to-differential signal path. Connect by signal name and confirm the actual pin mux and transceiver schematic before applying encoder power.

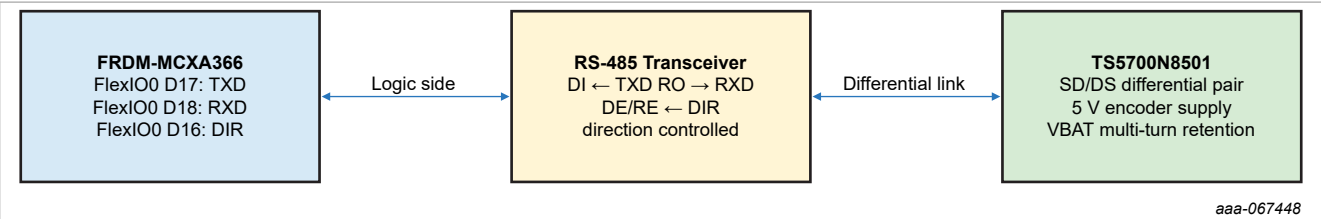


Figure 4. FRDM-MCXA366, T-format electrical interface, and TS5700N8501 signal mapping

3.1 MCU-side signal mapping

FlexIO resource mapping.

Table 9. MCX A366 T-format pin mapping

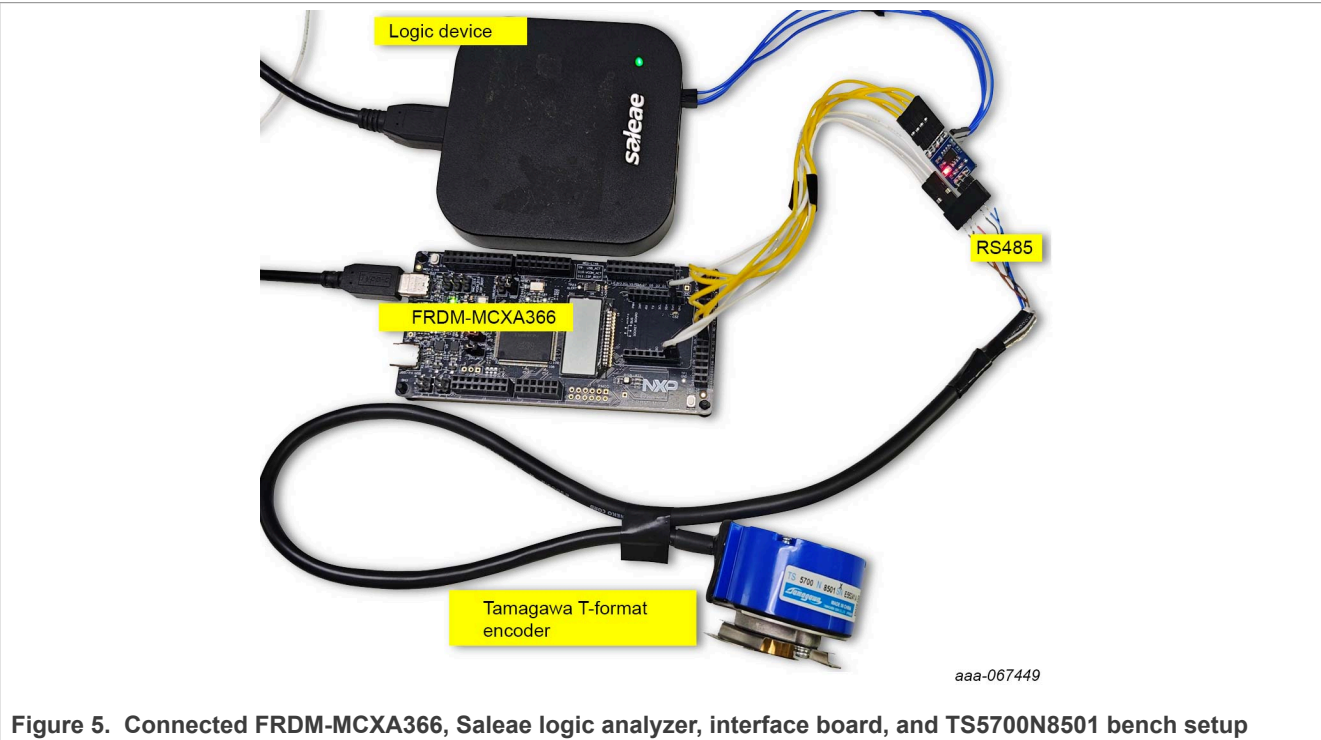
MCX A366 resource	Project symbol	Function
FLEXIO0_D16	FLEXIO_T_FORMAT_DR_PIN	RS-485 DE/DIR
FLEXIO0_D17	FLEXIO_T_FORMAT_TX_PIN	Host TXD
FLEXIO0_D18	FLEXIO_T_FORMAT_RX_PIN	Encoder RXD
Shifter 0/1	T_FORMAT_*_SHIFTER_INDEX	TX and RX
Timer 0/1/2	T_FORMAT_*_TIMER_INDEX	DE, TX, RX

3.2 Electrical interface and direction control

TIMER0 drives DIR. Config_DR_length() updates TIMCMP0 before each request so the transceiver is enabled only for the request frame count. The host must not drive the bus after request completion, because the encoder begins response transmission after the delimiter.

3.3 Power, ground, and wiring

Use the specified 5 V encoder supply and common logic reference. An external VBAT source is required for multi-turn retention during main-power-off operation. Confirm SD/DS polarity, transceiver biasing and termination, and supply connections before the polling program is run.



3.4 Encoder profile and source traceability

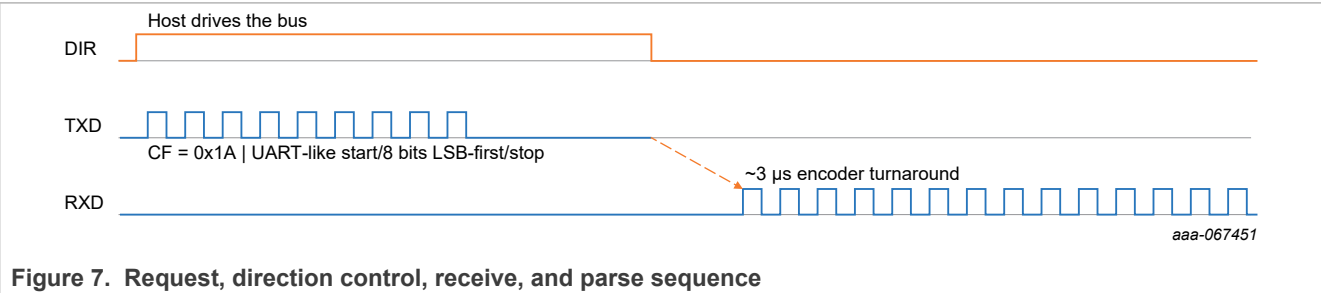
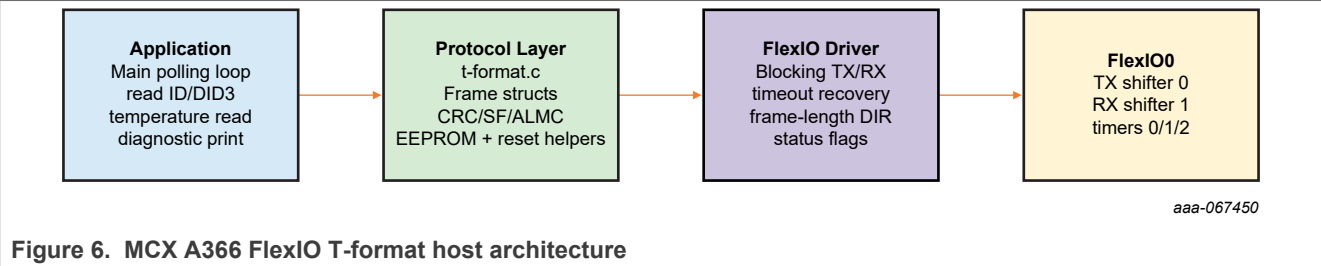
Table 10 shows encoder profile and source traceability.

Table 10. Encoder profile and source traceability

Item	Value	Source
Encoder	Tamagawa TS5700N8501	TS5700N8501 specification
Bit rate	2.5 Mbit/s	fsl_flexio_t-format.c
ABS/ABM length	17 bits/16 bits	t-format_polling_transfer.c
Host project	frdmmcxa366_t-format_polling_transfer	Workspace source
Register values	TX/RX shifters and TIMER0/1/2	Debug information

4 Software architecture

Figure 6 and Figure 7 summarize the implementation in source/t-format_polling_transfer.c, source/t-format.c, drivers/fsl_flexio_t-format.c, and board/app.h.



4.1 Software modules

Table 11 shows the responsibility of software module.

Table 11. Software module responsibilities

Module	Responsibility
t-format_polling_transfer.c	Board startup, polling cadence, diagnostic output, optional startup reset
t-format.c/t-format.h	DID request composition, response parsing, CRC, EEPROM, temperature, reset helpers
fsl_flexio_t-format.c/.h	FlexIO configuration, blocking transfers, timeout recovery
board/app.h	Pin and shifter/timer allocation

4.2 Initialization and runtime flow

BOARD_InitHardware() initializes clocks, pins, and the console. The application assigns pins, shifters, and timers, then calls FLEXIO_T_Format_Init(). It reads the encoder ID and DID3 at startup. The periodic loop delays 90 ms, reads DID3, delays 10 ms, reads temperature, and reports changed or periodic values.

4.3 CPU and FlexIO resource allocation

Table 12. CPU and FlexIO resource allocation

Resource	Assignment
Cortex-M33	Starts transfers, validates and parses frames, prints diagnostics.
Shifter 0	TXD request serialization.
Shifter 1	RXD response capture.
TIMER0	DE/DIR active window.
TIMER1	TX 2.5 Mbit/s baud/bit timing.

Table 12. CPU and FlexIO resource allocation...continued

Resource	Assignment
TIMER2	RX 2.5 Mbit/s baud/bit timing and byte synchronization.

5 Key implementation details

This section describes the key FlexIO configuration, timing, direction-control, register, frame-processing, and timeout mechanisms used by the T-format host implementation.

5.1 FlexIO Clock and 2.5 Mbit/s bit rate

FLEXIO_T_FORMAT_GetTimerCompare() derives the timer divider from the FlexIO source clock and fixed 2.5 Mbit/s target. The recorded register image uses 0x0F11: the low byte is the half-bit divider and the high byte is the dual-8-bit timer bit-count encoding for 8 data bits.

Table 13. 2.5 Mbit/s timer calculation

Parameter	Value/interpretation
Target bit rate	2,500,000 bit/s
Frame format	1 start + 8 LSB-first data + 1 stop
TX/RX TIMCMP	0x0F11
Timer mode	Dual 8-bit baud/bit
Baud error allowed in driver	3 %

5.2 Direction control and receive timing

TIMER0 starts from TIMER1 TX activity and drives D16 high for the calculated request length. RXD arms the TIMER2 and resynchronizes on the timer-pin rising edge. SHIFTER1 samples on the RX timer negative edge, placing sampling near the middle of each received bit.

5.3 Register configuration

[Table 14](#) to [Table 18](#) are incorporated from the current FlexIO driver resource assignments. TIMCMP0 is deliberately dynamic: FLEXIO_T_Format_Config_DR_length() rewrites it before requests of different lengths.

Table 14. TX data - SHIFTER0 register configuration

Register	Field	Bits	Value	Description
SHIFTCTL0	TIMSEL	[25:24]	1	Selects the TIMER1/TX timer
	TIMPOL	[23]	0	Shifts on TX timer positive edge
	PINCFG	[17:16]	3	Shifter pin output
	PINSEL	[12:8]	17	Selects the FLEXIO0_D17/TX
	PINPOL	[7]	0	TX active high, not inverted
	SMOD	[2:0]	2	Transmit mode
SHIFTCFG0	PWIDTH	[20:16]	0	Parallel width 1 bit
	INSRC	[8]	0	Not applicable to TX
	SSTOP	[5:4]	3	Stop bit logic high

Table 14. TX data - SHIFTER0 register configuration...continued

Register	Field	Bits	Value	Description
	SSTART	[1:0]	2	Start bit logic low

Table 15. RX data - SHIFTER1 register configuration

Register	Field	Bits	Value	Description
SHIFTCTL1	TIMSEL	[25:24]	2	Selects the TIMER2/RX timer
	TIMPOL	[23]	1	Samples on RX timer negative edge
	PINCFG	[17:16]	0	Shifter pin output disabled
	PINSEL	[12:8]	18	Selects the FLEXIO0_D18/RX
	PINPOL	[7]	0	RX active high, not inverted
	SMOD	[2:0]	1	Receive mode
SHIFTCFG1	PWIDTH	[20:16]	0	Parallel width zero
	INSRC	[8]	0	Input from selected RX pin
	SSTOP	[5:4]	3	Stop bit check logic high
	SSTART	[1:0]	2	Start bit check logic low

Table 16. RS-485 DE gate - TIMER0 register configuration

Register	Field	Bits	Value	Description
TIMCTL0	TRGSEL	[29:24]	7	Internal trigger selects TIMER1 output
	TRGPOL	[23]	0	Trigger active high
	TRGSRC	[22]	1	Internal FlexIO trigger
	PINCFG	[17:16]	3	Timer drives output pin
	PINSEL	[12:8]	16	Selects FLEXIO0_D16/DE
	PINPOL	[7]	0	High polarity
	TIMOD	[2:0]	3	Single 16-bit timer
TIMCFG0	TIMOUT	[25:24]	0	Logic one when enabled; not affected by timer reset
	TIMDIS/TIMENA	[14:12]/[10:8]	2/2	Disable on compare; enable on trigger high
	TSTOP/TSTART	[5:4]/[1]	0/0	No frame start/stop bits
TIMCMP0	Compare	[15:0]	0x0194/0x0464/0x05CC	Dynamic request-length values

Table 17. TX baud rate - TIMER1 register configuration

Register	Field	Bits	Value	Description
TIMCTL1	TRGSEL	[29:24]	1	Internal trigger selects SHIFTER0 status
	TRGPOL	[23]	1	Trigger active low; status clear means data pending
	TRGSRC	[22]	1	Internal FlexIO trigger
	PINCFG	[17:16]	0	Timer does not drive pin
	PINSEL	[12:8]	17	D17 selector with SHIFTER0 output
	TIMOD	[2:0]	1	Dual 8-bit baud/bit mode
TIMCFG1	TIMDIS/TIMENA	[14:12]/[10:8]	2/2	Disable on compare; enable on trigger high
	TSTOP/TSTART	[5:4]/[1]	2/1	Generate stop and start bits
TIMCMP1	Compare	[15:0]	0x0F11	0x0F bit count and 0x11 half-bit divider

Table 18. RX baud rate - TIMER2 register configuration

Register	Field	Bits	Value	Description
TIMCTL2	TRGSEL	[29:24]	36 (0x24)	External trigger selects D18
	TRGPOL/TRGSRC	[23]/[22]	0/0	Active-high external trigger
	PINCFG	[17:16]	0	Timer does not drive pin
	PINSEL/PINPOL	[12:8]/[7]	18/1	D18; active low for start detection
	TIMOD	[2:0]	1	Dual 8-bit baud/bit mode
TIMCFG2	TIMOUT	[25:24]	2	Output set high on timer reset
	TIMRST	[18:16]	4	Reset on pin rising edge
	TIMDIS	[14:12]	2	Timer disabled on timer compare
	TIMENA	[10:8]	4	Timer enabled on pin rising edge
	TSTOP/TSTART	[5:4]/[1]	2/1	Stop/start-bit check enabled
TIMCMP2	Compare	[15:0]	0x0F11	Same 2.5 Mbit/s timing as TX, 8-bit

The TIMCMP0 value is 0x0194 for a one-byte request, corresponding to 4.5 μ s. For a three-byte request, the value is 0x0464, corresponding to 12.5 μ s. For a four-byte request, the value is 0x05CC, corresponding to 16.5 μ s. Config_DR_length() overwrites the initial 0x0167 value.

5.4 Frame parsing and CRC8

T_Format_Readout_ABS_ABM_Parse() verifies CF and CRC, copies SF/ALMC/ENID, then assembles ABS and ABM from byte arrays. ABS is unsigned and masked to 17 bits. ABM is masked to 16 bits and sign-extended. multiTurnValid is cleared for multi-turn error or battery error.

Table 19. Frame parsing and CRC8 processing

Check	Result
CF echo	Frame error if response CF is not DID3 CF

Table 19. Frame parsing and CRC8 processing...continued

Check	Result
CRC	Frame error if complete CRC remainder is non-zero
SF	Returns the encoder or communication status for a good frame
ALMC	Reports OS/FS/CE/OF/OH/ME/BE/BA bits
ABS/ABM	Publishes parsed values only after a valid response

5.5 Timeout and diagnostic output

Blocking read/write loops use T_FORMAT_RETRY_TIMES. On timeout, FLEXIO_T_Format_RecoverFromTimeout() disables configured shifter/timer modes, clears stale flags and errors, and restores the saved configuration before the next attempt.

6 Main function reference

This section summarizes the primary functions used to initialize the T-format interface, acquire encoder data, and perform diagnostic and reset operations.

Table 20. Main function reference

Function	Purpose
FLEXIO_T_Format_Init()	Configures the control, TX/RX shifters, and DE/TX/RX timers.
T_Format_Get_Encoder_ID()	DID2 request and ID validation.
T_Format_Readout_ABS_ABM()	DID3 full-position request and parse.
T_Format_Get_Temperature()	EEPROM page 7/address 5 temperature access.
T_Format_Reset_Error()	DID7 error reset; ten frames.
T_Format_Reset_Multiturn_And_Error()	DIDC multi-turn and error reset; ten frames.

7 Build, flash, and run

This section explains how to prepare the development environment, build and download the application, run it on the target hardware, and verify the resulting serial output.

7.1 Development environment

The development and validation setup consists of the following software environment, target board, and encoder hardware:

- Development environment: MCUXpresso IDE with the MCX A366 SDK
- Target board: FRDM-MCXA366
- Encoder system: Tamagawa TS5700N8501 connected through an RS-485-compatible, direction-controlled differential interface

7.2 Build

1. Search for "an-mcxa366-tamagawa-tformat-interface-using-flexio" on <https://github.com/nxp-appcodehub>, and import it into MCUXpresso IDE.

2. Select the Debug build configuration for the FRDM-MCXA366 target.
3. Run a Build Project and wait for the build to complete without errors.
4. Confirm that the Debug output directory contains the generated axf executable.
5. If the build reports a FlexIO resource or pin conflict, verify the D16/D17/D18 pin assignment and the shifter/timer definitions in board/app.h before rebuilding.

7.3 Download and run

To download and run the application safely, verify the hardware setup before programming the target and monitoring the serial output:

1. With board power off, verify encoder supply, common ground, SD/DS differential wiring, and the RS-485 transceiver direction connection.
2. Connect the MCU-Link USB cable and confirm that the MCUXpresso IDE detects the board.
3. Start the serial terminal.
4. Download the Debug axf to the FRDM-MCXA366 target.
5. Reset and run the target, then observe the DID2/DID3 and temperature messages in the serial terminal.

Note: Keep `T_FORMAT_RESET_ERRORS_AT_STARTUP` disabled during normal operation. Any reset command changes the encoder state and an authorized service procedure with the shaft stationary must issue it.

7.4 Expected serial output

The following validated serial capture was obtained with the TS5700N8501 at a 90 MHz FlexIO root clock. This capture intentionally enables the optional DIDC path for controlled testing. Because DIDC resets multi-turn data and errors, it must remain disabled during normal operation. Enable it only when authorized by an explicit service procedure.

Validated serial-output capture is as follows:

```
Encoder T-format example
FlexIO Root Clock is 90 MHz
[T-format] Encoder ID: 0x11
        No error occurs
[T-format] Multi-turn data: 0, single-turn data: 73169
        SF: 0x00 (no error)
        ALMC: 0x00 (no encoder error)
[T-format] Temperature: 29
[T-format] Sending multi-turn and error reset (IDC, CF 0x62), 10 frames, keep the shaft
still...
[T-format] Reset request returned status = 0
[T-format] After reset, multi-turn data: 0, single-turn data: 73169
        SF: 0x00 (no error)
        ALMC: 0x00 (no encoder error)
[0.30s] Encoder ID: 0x11
        Multi-turn data: 0, single-turn data: 75279
        Temperature: 29
[0.40s] Encoder ID: 0x11
        Multi-turn data: 0, single-turn data: 80474
        Temperature: 29
[0.50s] Encoder ID: 0x11
        Multi-turn data: 0, single-turn data: 83602
        Temperature: 29
[0.60s] Encoder ID: 0x11
        Multi-turn data: 0, single-turn data: 89305
        Temperature: 29
```

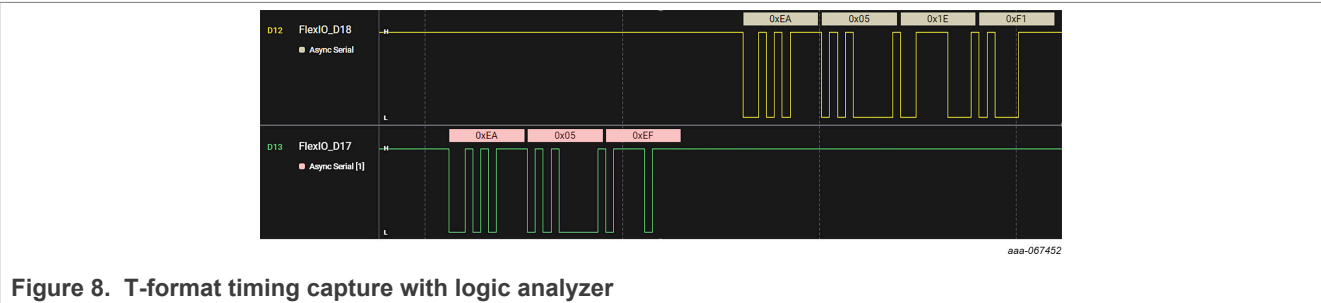
8 Results and observations

The validated capture shows a successful DID2 ID read (0x11) and a successful DID3 frame with SF=0x00 and ALMC=0x00. It also shows a temperature reading of 29 and a successful ten-frame DIDC operation.

After DIDC completes with status 0, repeated DID3 reads show the single-turn position increasing from 73169 to 89305 while multi-turn remains 0.

A separate representative TS5700N8501 DID3 response at 2.5 Mbit/s is RX=1a 20 c5 12 01 11 01 00 00 c0 3c. It decodes to ST=70341, MT=1, position=0x00000000000312c5, ID=0x11, SF=0x20, ALMC=0xC0, with matching CF and CRC.

This frame validates request, response length, byte order, receive path, parser, and CRC. SF=0x20 and ALMC=0xC0 are battery-related encoder conditions. They require system treatment for multi-turn retention but do not imply a serial-link failure when CF and CRC are valid.



9 Conclusion

The MCX A366 polling project implements a T-format host with FlexIO TX, RX, and direction-control resources. A compliant differential transceiver is required.

DE must be released before the encoder responds, and the timer must be configured for 2.5 Mbit/s operation. The implementation must also validate the CRC and distinguish valid frames that report encoder alarms from invalid frames.

10 References

This application note references the FlexIO T-format protocol implementation for NXP i.MX RT1186 and the Tamagawa TS5700N8501 encoder specification.

11 Acronyms

Table 21 lists the acronyms used in this document.

Table 21. Acronyms

Term	Definition
ABS	Single-turn Absolute
ABM	Multi-turn Absolute
API	Application Programming Interface
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check

Table 21. Acronyms...continued

Term	Definition
DIR	Driver-enable Signal
EMC	Electromagnetic compatibility
IDE	Integrated Development Environment
LSB	Least significant Bit
NRZ	Non-return-to-zero
SDK	Software Development Kit
UART	Universal Asynchronous Receiver-transmitter
USB	Universal Serial Bus

12 Revision history

[Table 22](#) summarizes the revisions to this document.

Table 22. Revision history

Document ID	Release date	Description
AN15164 v1.0	28 September 2026	Initial public release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at <https://www.nxp.com/profile/terms>, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Tamagawa T-format Encoder Host Interface using FlexIO on MCX A366

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μ Vision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

Contents

1	Overview	2
1.1	Key features	2
1.2	Scope	2
2	Fundamentals	2
2.1	T-format physical interface	2
2.2	DID3 position transaction	3
2.3	Frame and CRC8	3
2.4	Role of FlexIO	4
2.5	TS5700N8501 profile notes	4
2.6	Control and data fields	4
2.7	EEPROM and reset transactions	6
3	Hardware pinout and wiring	7
3.1	MCU-side signal mapping	7
3.2	Electrical interface and direction control	7
3.3	Power, ground, and wiring	8
3.4	Encoder profile and source traceability	8
4	Software architecture	8
4.1	Software modules	9
4.2	Initialization and runtime flow	9
4.3	CPU and FlexIO resource allocation	9
5	Key implementation details	10
5.1	FlexIO Clock and 2.5 Mbit/s bit rate	10
5.2	Direction control and receive timing	10
5.3	Register configuration	10
5.4	Frame parsing and CRC8	12
5.5	Timeout and diagnostic output	13
6	Main function reference	13
7	Build, flash, and run	13
7.1	Development environment	13
7.2	Build	13
7.3	Download and run	14
7.4	Expected serial output	14
8	Results and observations	15
9	Conclusion	15
10	References	15
11	Acronyms	15
12	Revision history	16
	Legal information	17

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.