AN14806

KE1xF to MCX E24x Migration Guide

Rev. 1.0 — 22 September 2025 Application note

Document information

Information	Content
Keywords	AN14806, KE1xF, MCX E24x, migration
Abstract	This application note describes the key differences between the KE1xF and MCX E24x families and their general considerations when migrating from KE1xF to MCX E24x.

1 Introduction

This application note describes the key differences between the KE1xF and MCX E24x families and their general considerations when migrating from KE1xF to MCX E24x.

The goal of this document is to enable developers to extend their existing KE1xF-based applications by leveraging the MCX E24x family, which offers higher flash and RAM capacity, improved integration, and enhanced long-term support. In many cases, the hardware remains pin-to-pin compatible and the software transition is simplified thanks to the shared MCUXpresso SDK architecture. This document highlights the key hardware and software aspects to achieve a seamless and efficient migration path.

2 High-level change summary

Taking the KE1xF family as a reference, Table 1 presents the high-level changes observed when migrating to the MCX E24x family. It includes a column labeled "Migration impact", which qualifies the level of impact that each change may have on software migration efforts.

The classification used for the "Migration impact" column is as follows:

- High: IP differs significantly (new version, different register map, or new behavior). It requires to rewrite the software component.
- Medium: A major change to an existing IP, such as a new version or extended functionality, potentially impacting driver configuration or register-level access.
- Low: A minor update or no significant change to the IP, typically requiring minimal or no software modifications.
- No Diff: Identical IP with the same software interface (API). No code changes are required when migrating from KE1xF to MCX E24x.
- N/A: The IP does not exist in the source platform (KE1xF), so no migration is applicable.

Table 1. KE1xF and MCX E24x family level comparison

	Family name	KE1xF	Migration impact	MCX E24x
Memory and security	P-flash	256 kB to 512 kB	No diff	512 kB to 2 MB
	D-flash [kB]	Up to 32 kB (Flex NVM)	Low	Up to 64 kB
	EEPROM [kB]	Up to 4 kB (Flex RAM)	No diff	4 kB (FlexRAM)
	Total RAM [kB]	Up to 64 kB	Low	64 kB to 256 kB
	Security solution	MPU, CRC, WDT, EWM	High	MPU, CRC, WDT EWM, EdgeLock Accelerator (CSEC)
Core/performance	Core quantity	1 x CM4F	No diff	1 x CM4F
	Frequency [MHz]	Up to 168 MHz	No diff	Up to 112 MHz
	DMA	16 channels	Low	16 channels
	DMIPS	~ 200	No diff	~ 100
Communication	FlexCAN	2 (No FD)	Medium	3 FlexCAN (1 x FD MCX E245, 2 x FD MCX

Table 1. KE1xF and MCX E24x family level comparison*...continued*

	Family name	KE1xF	Migration impact	MCX E24x
				E246, 3 x FD MCX E247)
	Ethernet	✗ No	N/A	10/100 Mbps IEEE-1588 (MCX E247)
	SAI	✗ No	N/A	2 (MCX E247)
	LIN/LPUART	3	Low	3
	LPSPI	2	Low	3
	LPI2C	2	Low	1 up to 2 (MCX E247)
Extensions	FlexIO	8 channels	Low	8 channels
	QSPI	✗ No	N/A	✓ Yes (only for 144 LQFP and MCX E247)
Analog and timers	12-bit ADC	3	Low	2
	ACMP (instances)	3	Low	1
	TRGMUX, ADC cross triggering	✓ Yes	Low	✓ Yes
	Timer modules/ channels	4 FTM/8 channels each	Low	4 up to 8 FTM/8 channels each
	RTC	✓ Yes	No Diff	✓ Yes
Packaging	48 LQFP	✗ No	N/A	✓ Yes (MCX E245)
	64 LQFP	✓ Yes	Low	✓ Yes (MCX E245/246)
	100 LQFP	✓ Yes	Low	✓ Yes
	144 LQFP	✗ No	N/A	✓ Yes (MCX E246/247)

In summary:

- Depending on the application profile, around 50 % of the IPs and peripherals are reused or functionally equivalent between KE1xF and MCX E24x, while the remaining of 50 % shows functional or architectural changes, including updated peripheral versions, added features (for example, QSPI), or renamed modules. Migration at a software level is facilitated through updated SDKs available in MCUXpresso.
- While both families are based on Arm Cortex-M4F cores, the performance profiles differ, with KE1xF reaching up to 168 MHz versus 112 MHz for MCX E24x. Although this results in a lower raw CPU frequency on MCX E24x, the overall application performance is comparable or improved thanks to enhanced system-level features, such as additional CAN FD interfaces, QSPI, and other peripheral and accelerator improvements.
- Security capabilities differ significantly as MCX E24x integrates the EdgeLock Accelerator (CSEC) and additional memory protection logic compared to KE1xF, which may require a software adaptation for secure boot or key storage use cases.
- Packaging compatibility is maintained for shared options, such as 64 and 100 LQFP, which are pin-to-pin compatible between KE1xF and MCX E24x. Additional package variants introduced with MCX E24x (such as 48 LQFP and 144 LQFP) offer more compact or feature-rich design. A hardware redesign may

be required, but the pin assignment and peripheral distribution remain similar to support easier migration between equivalent packages.

3 Platform architecture

The following figure shows the block diagram for KE1xF. Figure 2 shows the block diagram for MCX E24x. The main differences are shown in Table 2.

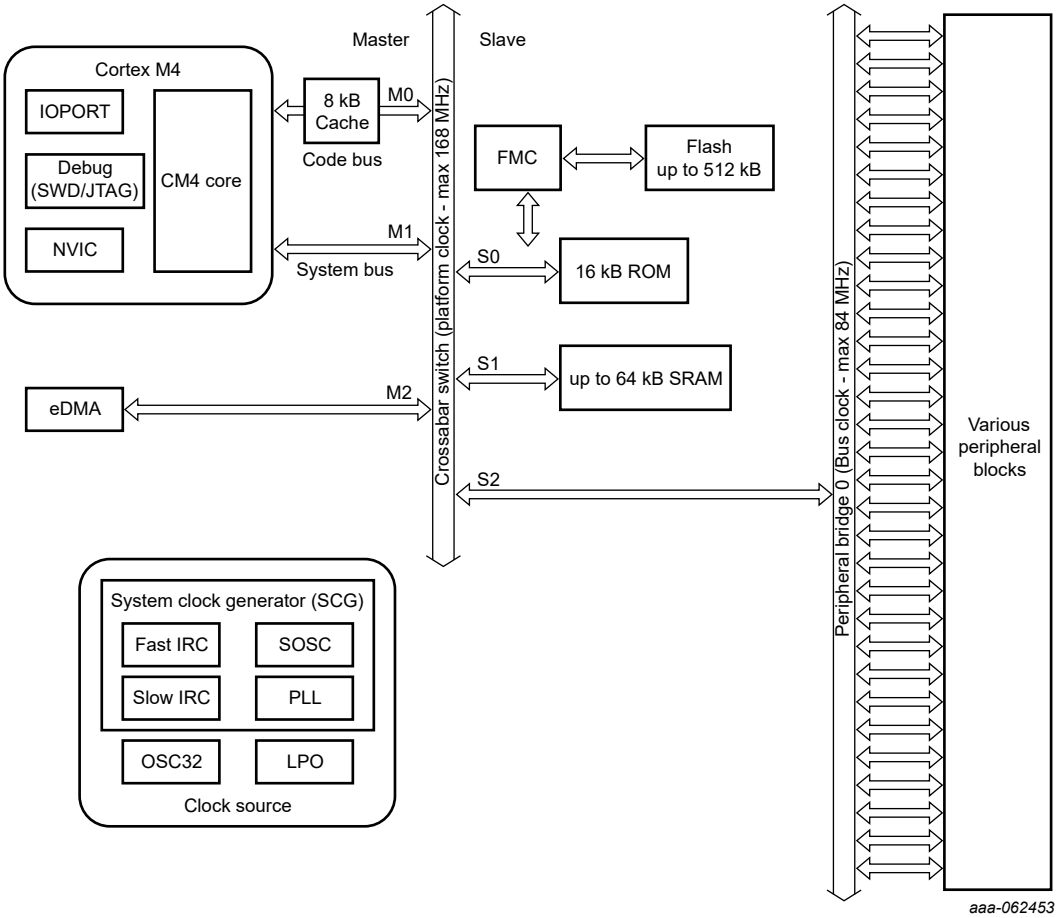

Figure 1. KE1xF block diagram

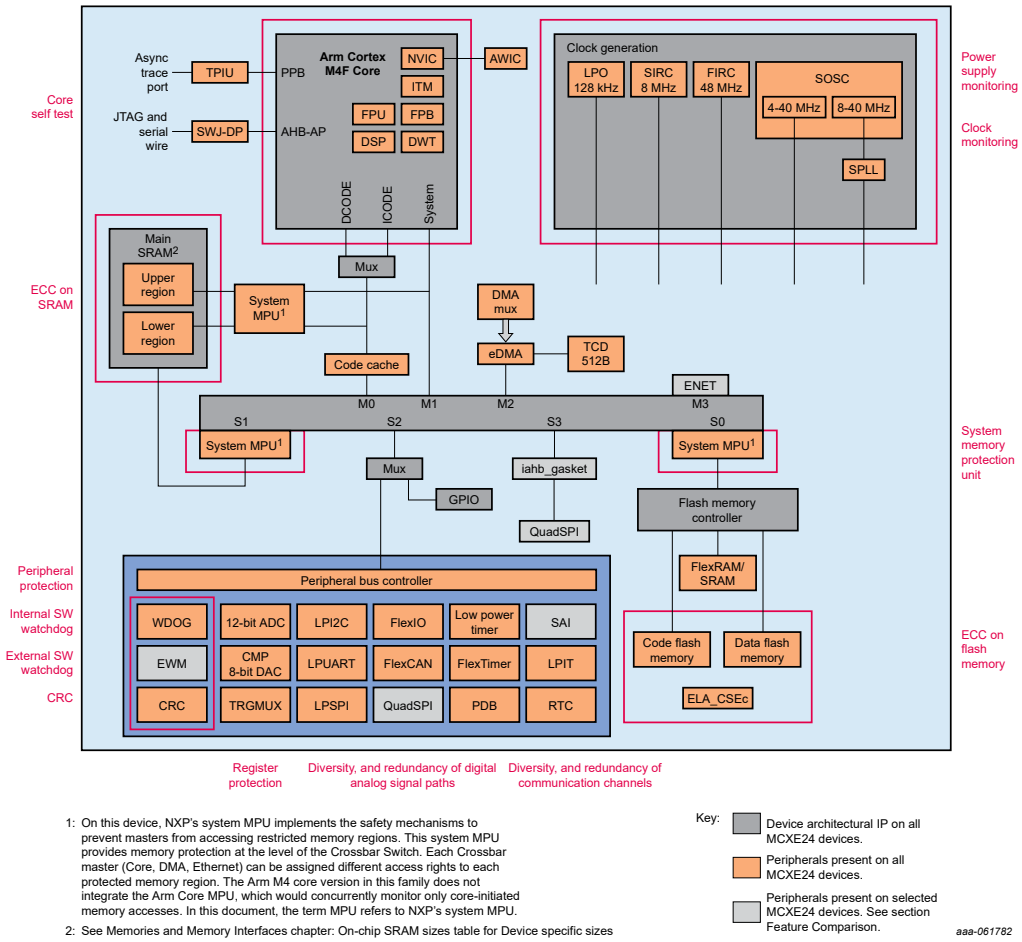

Figure 2. MCX E24x block diagram

Table 2. Platform differences

Feature	KE1xF	MCX E24x
BootROM	✓ Yes	✓ Yes (with flashloader support)
Voltage range	2.7 to 5.5 V	2.7 to 5.5 V
Code cache	8 kB	4 kB

For a typical KE1xF application, migrating to MCX E24x does not require deep architectural changes at the core or bus levels, because both families are based on Cortex-M4F and share similar XBAR architectures. However, additional adaptation may be needed if the application is updated to leverage new features introduced in MCX E24x, such as the EdgeLock Accelerator (CSEC) security module, code cache, QSPI, or advanced clock/power domains.

4 Memory and OTA

The following table shows an extended comparison between the different memories and their interfaces for KE1xF and MCX E24x.

Table 3. Memory general differences

Feature	KE1xF	MCX E24x
Program flash memory	256 to 512 kB	512 kB to 2 MB
FlexMemory	64 kB	Up to 64 kB
Data flash	Up to 64 kB	Up to 64 kB
Flash memory controller cache	Single speculative prefetch buffer	Single speculative prefetch buffer
Random-Access Memory (RAM)	Up to 64 kB	64 kB to 256 kB
FlexRAM (also available as system RAM)	Up to 4 kB, used with FlexNVM for EEPROM emulation	4 kB
Low-leakage stand-by memory	✗ No	RAM retained in all modes
QuadSPI (external memory interface)	✗ No	Supports SDR and HyperRAM modes with up to 4 and 8 bidirectional data lines, respectively
Error correcting code	✓ Yes	✓ Yes
Cache	8 kB I/D	4 kB
EEPROM emulated by FlexRAM	Up to 4 kB EEPROM emulated using up to 32 kB FlexNVM with FlexRAM as the EEPROM buffer (SW-configurable partitioning)	4 kB EEPROM supported via 64 kB D-flash with FlexRAM as the EEPROM buffer; the remaining D-flash can be used as data/program flash
OTA	Some derivatives support it through MCUBoot	Supported via MCUBoot for safe OTA updates

There are significant changes in memory architecture between the KE1xF and MCX E24x devices. The following sections describe the most important features and points in detail.

4.1 RAM

The MCX E24x family offers a significant improvement in the available RAM compared to the KE1xF family. While KE1xF devices include up to 64 kB of RAM, the MCX E24x devices scale from 64 kB up to 256 kB, depending on the derivative. This extended RAM enables more complex applications and supports use cases requiring higher data buffering, such as networking stacks, graphics, or edge processing. Additionally, both families feature 4 kB of FlexRAM, primarily intended for EEPROM emulation, which can also be repurposed as system RAM if the EEPROM function is not used. The increased RAM in the MCX E24x devices is a key differentiator and allows developers to consolidate functionality previously distributed across multiple MCUs.

4.1.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to MCX E24x, account for the following key considerations regarding the RAM architecture:

- RAM size and distribution: MCX E24x offers significantly more RAM (up to 256 kB) compared to KE1xF (up to 64 kB). However, this memory is distributed across multiple blocks, which may require linker script adjustments and code/data relocation planning during migration.
- FlexRAM usage: While both families support FlexRAM, its configuration and integration differ. On KE1xF, FlexRAM is more tightly coupled with FlexNVM for EEPROM emulation, whereas on MCX E24x, FlexRAM can be more flexibly configured for system RAM or EEPROM backup. Developers should carefully verify how the EEPROM emulation is handled and adjust the initialization code, if needed.

- Stand-by memory retention: MCX E24x includes low-leakage stand-by memory retained across low-power modes, which is limited or absent in KE1xF. Applications relying on RAM retention across deep sleep benefit from this, but they may require updates to power-management strategies.
- Cache architecture: KE1xF includes an explicit 8 kB instruction/data cache, whereas MCX E24x provides 4 kB. This can affect performance for memory-intensive applications and optimization may be needed when migrating timing-critical code.
- Toolchain and SDK differences: RAM layout and memory definitions differ between the MCUXpresso SDK for KE1xF and MCX E24x. The migration typically requires updating the linker scripts, startup code, and memory protection configuration (MPU).

A careful review of the MCX E24x reference manual and SDK memory configuration examples is essential to ensure a smooth transition, particularly when working with low-power features or EEPROM emulation.

4.2 Flash

The flash memory architecture differs moderately between KE1xF and MCX E24x, with MCX E24x offering more capacity and expanded flexibility for OTA and EEPROM use cases.

The KE1xF family provides 256 kB to 512 kB of program flash, backed by an 8 kB cache and FlexMemory support, including FlexNVM and FlexRAM. This allows applications to emulate EEPROM in a reliable and configurable manner, suitable for parameters or runtime configuration storage.

The MCX E24x family extends the flash capacity from 512 kB up to 2 MB, while also retaining FlexMemory, including up to 64 kB of FlexNVM and 4 kB of FlexRAM. The flash cache is 4 kB, slightly smaller than KE1xF, but combined with a more advanced memory controller.

4.2.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to MCX E24x, consider the following aspects of the flash architecture:

- Flash size and organization: While KE1xF devices offer between 256 kB and 512 kB of P-flash, MCX E24x devices extend this range up to 2 MB, providing more room for application code, data, and Over-The-Air (OTA) capabilities. This expanded space allows for more flexibility but it may also impact flash partitioning and memory map configuration. MCX E24x introduces the Read-While-Write (RWW) capability, allowing the application to execute code from one flash bank while programming another. This feature is essential to support safe Over-The-Air (OTA) updates, as it enables seamless firmware download and installation without interrupting normal application execution.
- FlexNVM and FlexRAM usage: Both families support 64 kB of FlexNVM and 4 kB of FlexRAM. However, their behavior and configuration may differ in terms of EEPROM emulation or partitioning options. Applications that rely on EEPROM emulation should verify the compatibility and reconfigure the FlexNVM partition as needed on MCX E24x.
- Cache configuration: KE1xF features an 8 kB cache, whereas MCX E24x provides 4 kB. While this is sufficient for most applications, performance-critical code sections must be reviewed and optimized for the new cache size.
- Flash driver migration considerations: Both KE1xF and MCX E24x families use the MCUXpresso SDK flash driver, meaning that the core flash API (such as initialization, sector erase, program, and verify operations) remains consistent across both families. Developers migrating applications from KE1xF will find the flash interface largely compatible, with minimal or no changes required at the API level. Differences may exist in the peripheral base addresses and flash layout, but the overall software migration is straightforward. However, the MCX E24x family introduces new use cases, enabled by the additional boot and memory management features. MCX E24x supports integration with the open-source MCUBoot second-stage bootloader. This enables more advanced firmware-update mechanisms, including Over-The-Air (OTA) updates with support for dual image slots and image validation before activation. The MCUXpresso SDK for MCX E24x includes a preconfigured example (ota_mcuboot_basic) that demonstrates how to configure and manage the OTA-

capable flash partitions. This capability is not natively supported on KE1xF, where any firmware update mechanism must be fully implemented by the user application. As a result, while the flash driver APIs are similar, developers migrating to MCX E24x must also consider the opportunity to adopt a more robust and secure firmware update strategy using MCUBoot, which may require adaptations in memory layout, linker scripts, and boot flow.

- Flash performance considerations: Both KE1xF and MCX E24x feature flash memory designed for read access at system clock speeds. However, to maintain reliable operation and optimal performance, the flash controller must be configured with appropriate wait states depending on the system clock frequency. On MCX E24x, this is managed via the Flash Memory Controller (FMC), which dynamically adjusts access latency. MCX E24x integrates a prefetch buffer mechanism within the FMC that improves performance by reducing wait states during instruction fetches and data reads.

4.2.2 KE1xF versus MCX E24x reliability and time specifications

Table 4. Endurance and performance

Feature	KE1xF	MCX E24x	Units
Data retention up to 1 K cycles	20	20	Years
Cycling endurance (min)	10 K	10 K	Cycles
Erase flash block execution time (512 kB)	435	250	ms
Program section execution time (1 kB)	5	5	ms
Erase all blocks execution time	500	400/700/1400 (MCX E245/E246/E247 flash size: 512 kB/1 MB/2 MB)	ms

4.3 Over-The-Air (OTA)

The Over-The-Air (OTA) firmware update capability is a key feature for modern embedded systems that require remote firmware updates, especially in automotive and IoT applications. The KE1xF family does not natively support OTA-specific hardware features, meaning that any OTA mechanism must be fully implemented in software by the application developer. The MCX E24x family offers a structured solution using a two-stage bootloader approach:

1. ROM bootloader (first stage): Provides basic functionality, such as code signature validation and factory image management.
2. MCUBoot (second stage): Integrated in the MCUXpresso SDK, it handles a full OTA flow with signed image verification and flash partitioning into primary/secondary slots.

This feature does not rely on a hardware A/B flash swap. Instead, MCX E24x uses a software-driven dual-slot partitioning strategy implemented via MCUBoot (included in the MCUXpresso SDK).

Table 5. OTA features

OTA feature	KE1xF	MCX E24x
A/B swap	✕ No	✕ No natively (supported through MCUBoot)
Number of flash partitions	1 main, optional D-flash	2 (or more) via partitioning
Read while write	Limited support	Enhanced support
Image verification	By application	EdgeLock Accelerator (CSEC) functions

Table 5. OTA features*...continued*

OTA feature	KE1xF	MCX E24x
Lockable flash regions	Limited via FPROT	Enhanced
Update while the application is running	Not safe	Possible (with constraints)
Secure boot	✕ No	EdgeLock Accelerator (CSEC) is supported

4.3.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to MCX E24x, you can leverage several improvements in the Over-The-Air (OTA) update capabilities, although they are not natively implemented in the hardware. Unlike KE1xF, which requires a fully custom implementation for features such as dual-image management and update rollback, MCX E24x offers a more modern hardware foundation with better integration possibilities for secure boot, flash partitioning, and secure provisioning tools.

The secure boot mechanisms and cryptographic verification can be integrated easier on MCX E24x using the MCUXpresso Secure Provisioning Tool or Flashloader.

Migrating to MCX E24x requires rethinking the flash layout and linker scripts to accommodate multiple firmware images and version metadata. Support for Position-Independent Code (PIC), secure boot, image verification, and rollback logic must be added at the software level. While this adds complexity, it enables more robust and secure update flows than what is typically feasible on KE1xF.

5 Clocking, power management, reset, and boot

This section describes the clocking, power management, reset, and boot.

5.1 Clocking architecture

The clocking architecture defines how the microcontroller distributes and manages clock signals across its core and peripherals, impacting both performance and power consumption. Both the KE1xF and MCX E24x platforms use a System Clock Generator (SCG) and Peripheral Clock Controller (PCC) scheme, but the MCX E24x family introduces notable enhancements in configurability and fine-grained control.

KE1xF provides essential clock sources (FIRC, SIRC, SOSC, and SPLL) and allows dynamic switching between them using the SCG. However, its configuration is relatively static and centralized, with limited flexibility per a peripheral domain.

In contrast, MCX E24x significantly improves on this model by introducing a hierarchical and modular clocking architecture. It supports multiple clock roots per subsystem, allowing independent configuration for different groups of peripherals. This is complemented by additional modules such as the Clock Monitor Unit (CMU) for enhanced fault detection and the Clock Control Module (CCM) for managing complex scenarios like low-power transitions or domain-specific clock gating. The MCX E24x platform also supports more extensive use of fractional dividers and more fine-tuned PLL/DIV settings, enabling precise clock tailoring for performance or energy optimization.

These architectural improvements enable the MCX E24x to better support advanced low-power modes, improve startup behavior, and offer greater flexibility in mixed-performance applications.

Table 6. KE1xF and MCX E24x clock sources

KE1xF	MCX E24x	Key comments
Fast IRC 48 MHz	Fast IRC 48 MHz	Default clock source after reset in both architectures

Table 6. KE1xF and MCX E24x clock sources...*continued*

KE1xF	MCX E24x	Key comments
Slow IRC 8 MHz/2 MHz	Slow IRC 8 MHz	-
OSC 4 to 40 MHz	SOSC 4 to 40 MHz	-
PLL up to 50 MHz	SPLL up to 112 MHz (HSRUN) Up to 160 MHz (RUN)	MCX E24x: VCO frequency up to 320 MHz (KE1xF up to 360 MHz)
LPO 128 kHz	LPO 128 kHz	Intended RTC clock source for both architectures. No software trimming is supported.

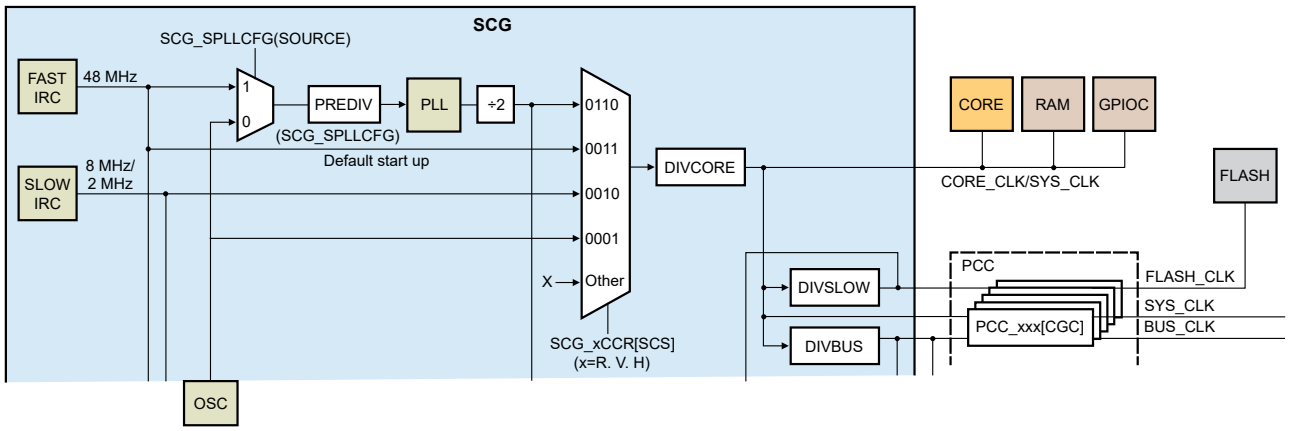

Figure 3. KE1xF system clocking

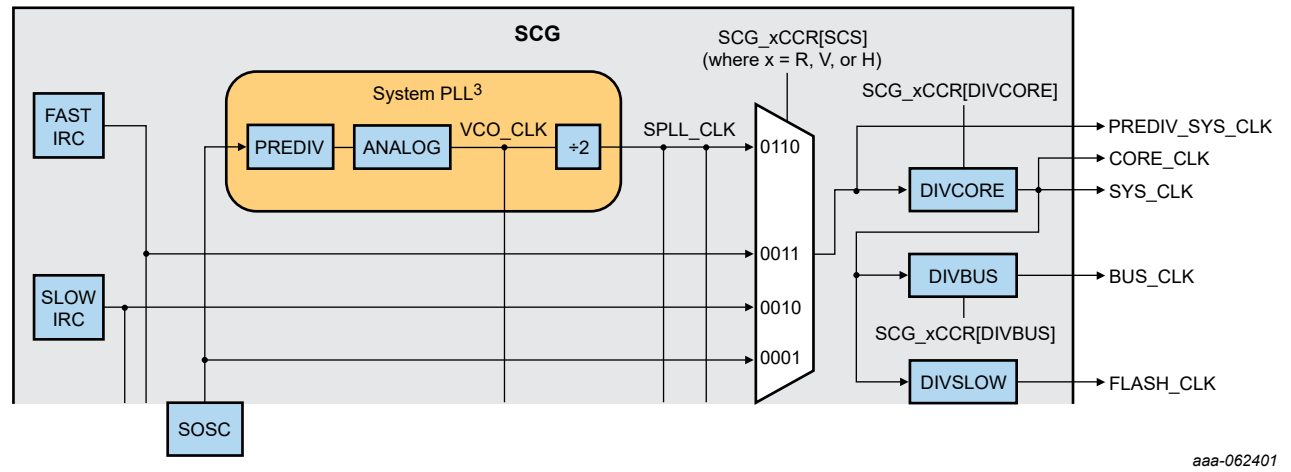

Figure 4. MCX E24x system clocking

5.1.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to MCX E24x, developers must adapt to a more modular and flexible clocking system. While both platforms share a similar System Clock Generator concept with key components like FIRC, SIRC, and SPLL, MCX E24x introduces enhancements in configurability and behavior.

Key differences include additional tuning and trimming capabilities for internal oscillators, a more advanced handling of clock stabilization, and subtle changes in register fields. Although the MCX E24x retains familiar modules, clock sources may behave differently during low-power modes or boot sequences.

Clock initialization and transitions must be reviewed carefully, especially for applications sensitive to oscillator stability or timing precision. The greater flexibility offered by MCX E24x comes with additional responsibility in configuration.

5.2 Power management and operating modes

One of the most eye catching differences in the transition from KE1xF to MCX E24x is the number of supported power modes. The implemented power modes in both architectures and their relationship with Cortex M4 are shown in the following figure.

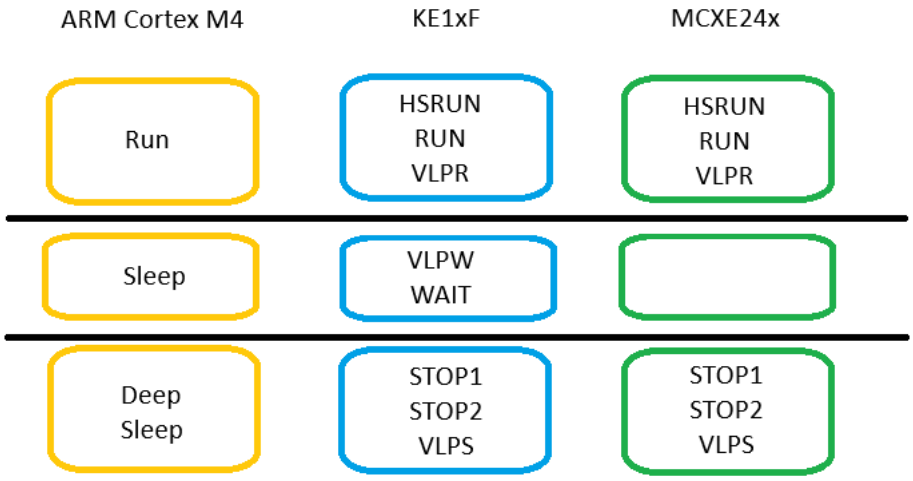

Figure 5. Power modes

5.2.1 Considerations for migrating to MCX E24x

Both KE1xF and MCX E24x devices implement multiple low-power operating modes and follow a similar power mode state machine structure: any reset leads the system back to the Normal Run mode, and the power regulation is active in the RUN and STOP modes. However, a few differences must be considered during migration.

On KE1xF, modes like WAIT, STOP, VLPR, and VLPW are available. VLPR and VLPW modes are frequency-limited and suitable for applications requiring reduced energy consumption while keeping the core active.

On MCX E24x, while modes such as RUN, STOP, and VLPR are still present, some intermediate modes are removed and VLPR is still frequency-limited. The power mode control is more tightly integrated with enhanced wake-up and retention features, although the power-regulation concept remains consistent (active in RUN and STOP modes).

Because of these similarities, transitioning existing low-power applications from KE1xF to MCX E24x is conceptually straightforward.

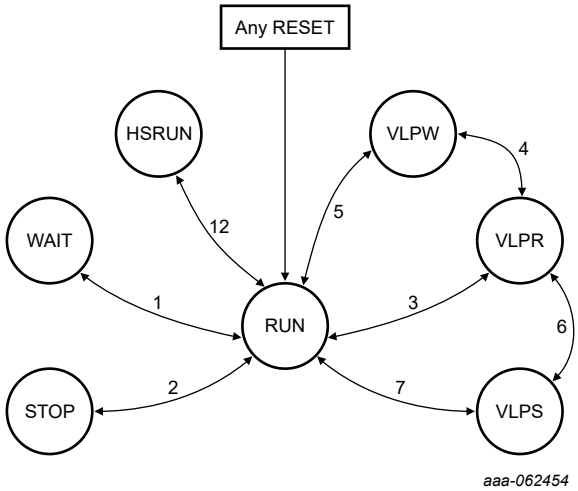

Figure 6. KE1xF power mode state transition diagram

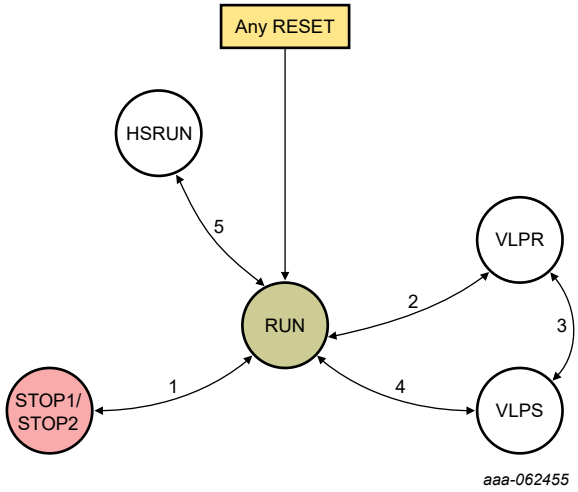

Figure 7. MCX E24x power mode state transition diagram

The following table lists the modules available during the deep sleep mode.

Table 7. KE1xF and MCX E24x module operation in low-power mode (deep sleep)

Modules	KE1xF VLPS	MCX E24x VLPS
Core, NVIC	Static	OFF
LVD/LVR	LVR active	LVR active
DMA	Async Op	Async Op
Watchdog	FF	Async Op
LPO	FF	FF
SIRC	Optional ON	FF
FIRC	Optional ON	OFF
SOSC	Optional ON	OFF
LPUART	Async Op	Async Op
LPSPI	Async Op	Async Op

Table 7. KE1xF and MCX E24x module operation in low-power mode (deep sleep)*...continued*

Modules	KE1xF VLPS	MCX E24x VLPS
LPI2C	Async Op	Async Op
FlexIO	Async Op	OFF
CAN	Wakeup	OFF
FTM	Static	OFF
LPIT	Async Op	Async Op
RTC	Async Op	FF
CRC	Static	OFF
CMP	LS compare only	LS compare only
ADC	FF	OFF

The legend for the above table is the following:

- *FF: Full functionality
- **Async Op: Fully functional with alternate clock source
- ***Static: Module register states and associated memories are retained

5.3 Reset and boot

Both KE1xF and MCX E24x families implement multiple types of reset to ensure reliable system startup and fault recovery. These resets are typically triggered by power conditions, external signals, software requests, or internal system events. While the core concept remains similar across both platforms, the underlying reset sources and their handling mechanisms may slightly differ.

The following table summarizes the primary reset sources available on each platform.

Table 8. KE1xF and MCX E24x reset sources

Event type	KE1xF	MCX E24x
Power-on reset	**Power supply is initially applied**	
	Supply voltage drops below VPOR	Supply voltage drops below VPOR
System reset	Asserting RESET_B pin	
	WDOG Software reset Loss of clock Loss of lock Core LOCKUP	
	Core attempts to enter the Stop mode, but not all modules acknowledge the request	
Debug reset	MDM-AP (reset via JTAG/SWD) JTAG module	

Note: *Many reset functions for the chip as well as the reset sequence monitoring are managed by the Reset Control Module (RCM) in both families.*

5.3.1 Secure installer for unsecure manufacturing lines

Figure 8 illustrates the difference between the KE1xF and MCX E24x flash layouts in the context of OEM factory provisioning using the Flash_loader. KE1xF devices are delivered with blank flash, whereas MCX E24x devices come preloaded with a one-time Flash_loader and Flash_loader_loader image in the flash memory (starting at address 0x0000_0000), as provided by NXP SCE.

Upon boot, the loader is copied from the flash to the RAM and executed from there. This loader interacts with a host PC (via UART or SPI) using tools like blhost, allowing flash programming and CSEc key provisioning. The setup also requires a boot hold from the factory for testing and IFR locking. The Flash_loader execution is prioritized to ensure proper initialization and provisioning before any application runs.

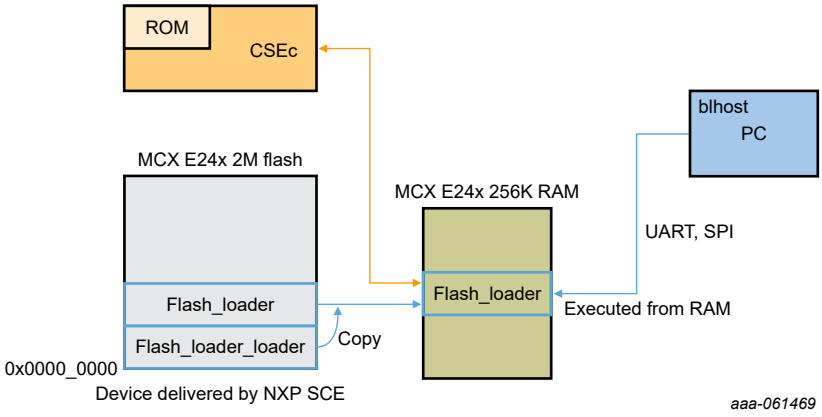

Figure 8. MCX E24x flashloader

5.3.2 MCUBoot support

On the MCX E24x devices, the MCUBoot open-source bootloader is provided as a secondary bootloader to enable the secure firmware boot and update capabilities. It is a widely adopted, vendor-agnostic solution that can work with Zephyr, FreeRTOS, or bare-metal MCUXpresso applications.

5.3.2.1 Key features

The key features are as follows:

- Authenticated firmware execution
  - Only digitally signed images are allowed to run (preventing unauthorized or tampered code)
- Cryptographic algorithms
  - Signing: ECDSA, ED25519, RSA
  - Integrity: SHA-256, SHA-512
  - Optional encryption: AES
  - Crypto libraries: Mbed TLS or TinyCrypt
- Firmware updates
  - UART, USB, or Over-The-Air (OTA) support
  - Rollback mechanism in case of a boot failure (not default behavior, configurable)
  - Supports direct update strategies
- Partition management; uses predefined flash areas for application images and metadata:
  - primary_slot (active image)
  - secondary_slot (update candidate)
  - scratch

5.4 Migration considerations

When migrating from the KE1xF platform using the MCUXpresso SDK to the MCX E24x family with MCUBoot integration enabled, consider the following adjustments for boot and image management:

- Image format: The application binary must now be wrapped in a MCUBoot-compatible format, which includes a header, optional cryptographic signature, and metadata. This packaging can be handled using imgtool.py, west, or integrated build system steps.
- Signing process: Establish a signing mechanism using a private key (via imgtool or Zephyr's west toolchain).
- Bootloader configuration: Make sure that linker files match the expected layout of MCUBoot (for example, vector table offset).
- Tools: Flashing may now involve MCUBoot-specific tools (blhost) instead of traditional flashing utilities.

6 Security

The KE1xF family provides a basic security model focused on flash memory protection and CRC. In contrast, the MCX E24x family introduces a more robust and modern security architecture that aligns with evolving market demands. This includes enhanced features, such as secure boot, key management, and hardware cryptographic acceleration.

This section describes the security features offered by both families and highlights the key differences. It also outlines considerations for a secure migration from KE1xF to MCX E24x.

Table 9. KE1xF and MCX E24x security differences

		KE1xF	MCX E24x
Security system		✕ No	EdgeLock Accelerator (CSEC)
Location in SoC		✕ No	EdgeLock Accelerator (CSEC) tightly coupled to flash controller
Firmware upgradable		Not natively	Not natively - through secure boot
Security ciphers	Symmetric	✕ No	AES-128
	Cipher modes	✕ No	ECB, CBC, and CMAC
	Hash	Software implementation only	Miyaguchi-Prenell No SHA HW in EdgeLock Accelerator (CSEC), must use software implementation
Secure boot		✕ No	Three different modes: parallel boot, sequential boot, strict sequential boot
Random number generator		✕ No	TRNG and PRNG
Attack resistance		✕ No	Secure key storage
Requirement coverage		Basic reliability	SHE - automotive-grade intermediate security

6.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to MCX E24x, consider the following key security enhancements:

- Transition from CRC to real cryptography: KE1xF relies solely on a CRC module for error detection, which, while suitable for basic integrity checks (IEC 60730), offers no cryptographic security or protection against intentional tampering. MCX E24x integrates the EdgeLock Accelerator (CSEC), enabling hardware-accelerated AES-128 encryption, CMAC-based authentication, and a true RNG.

- Secure boot and image authenticity: KE1xF lacks a secure boot mechanism and starts executing code immediately. MCX E24x employs a ROM-based secure boot that uses EdgeLock Accelerator (CSEC) to verify signed firmware images before execution. MCX E24x supports multiple secure boot strategies (parallel, sequential, and strict sequential), balancing security and boot latency.
- Firmware update security: Unlike KE1xF, all firmware updates on MCX E24x must be cryptographically signed and validated before flashing. The system also enforces antirollback through image versioning and key management, preventing unauthorized downgrades.
- Hardware-enforced key protection and debug lockdown: KE1xF offers no secure key storage. On MCX E24x, EdgeLock Accelerator (CSEC) provides hardware key slots, including one for the boot MAC key, and prohibits key use in case of failed authentication or debug detection. This includes the debug interface lockdown and permanent hardware lock after entering locked life cycle modes.
- Upgrade path and tool support: Adopting MCX E24x requires a transition to NXP's MCUXpresso Secure Provisioning workflow, including key generation, image signing, and configuration of secure flash regions via PRINCE/IPED. Developers must be familiar with the partitioning of secure memory regions and secure key life cycles.

7 Safety

The KE1xF family integrates a robust set of IEC 60730 Class B safety features, including comprehensive self-test routines (CPU, flash, RAM, clock, I/O), watchdog timers, CRC engines, and ECC support, all enabled via NXP's safety library.

These capabilities make KE1xF ideal for consumer appliances and general industrial applications where reliable error detection and system integrity are paramount.

The MCX E24x family enhances safety support with compliance to IEC 60730 (class B-certified) and provides features and documentation aligned with IEC 61508 requirements. It is ready for system-level SIL2 certification through NXP's SafeAssure program, offering key safety features such as error-correcting code on flash and SRAM, internal and external watchdogs (WDOG/EWM), system memory protection unit, and cyclic redundancy check module. Software libraries and guidelines are available to support integration in functional safety applications:

- Industrial Structural Core Self-Test (ISCST) library, which is designed to detect permanent faults in the Cortex-M4F core and is qualified to support SIL2 certification.
- Error Reporting Module (ERM) for real-time detection of single-bit faults and an Error Injection Module (EIM) to validate the fault-detection mechanisms.

These additions provide a much stronger foundation for industrial safety applications, especially in systems that must meet the IEC 61508 SIL2 certification requirements.

Table 10. KE1xF and MCX E24x safety-related differences

	KE1xF	MCX E24x
Memory protection unit	NXP proprietary	NXP proprietary
Error injection module Error reporting module	✗ No	✓ Yes (both)
Temperature sensor	✓ Yes, monitored via ADC	✗ No
eDMA controller	✓ Yes	✓ Yes
Clock monitoring units	1 (on-chip clock loss monitoring)	2
Interconnect bus	Crossbar switch (AXBS-Lite) and peripheral bridge (AIPS-Lite)	
External watchdog monitor	✓ Yes	

Table 10. KE1xF and MCX E24x safety-related differences*...continued*

	KE1xF	MCX E24x
PMC (Power Management Control)	Hardware protection against undervoltage conditions	Hardware-level fault-detection capabilities

8 Timing, Cross Triggering, and ADC

In general industrial applications, such as motor control, power conversion, and high-speed sensor interfacing, tight integration between timing, analog sampling, and event coordination is essential. Key modules like ADC, PDB, FTM, and TRGMUX are commonly used together to implement synchronized control loops and real-time monitoring. This section describes the architectural and functional differences between the KE1xF and MCX E24x families, focusing on how these IP blocks interact to support deterministic timing, cross-triggering mechanisms, and analog signal acquisition in complex industrial systems.

8.1 TRGMUX

The Trigger Multiplexer (TRGMUX) is a central routing IP that enables flexible interconnection of peripheral and timer events without CPU intervention. By selecting and forwarding trigger signals from sources like timers, PWM modules, and ADCs to target peripherals, TRGMUX facilitates precise hardware synchronization in tasks like synchronized sampling, multi-phase motor control, and time-critical event chaining. KE1xF features a two-stage TRGMUX (prestage and main stage), while MCX E24x platforms implement only one stage with up to 72 sources and multiple outputs, ensuring consistent trigger-routing capabilities. This section describes the configuration model, signal mapping, and migration considerations to leverage TRGMUX for deterministic cross-peripheral orchestration on MCX E24x.

Table 11. KE1xF and MCX E24x TRGMUX differences

TRGMUX	KE1xF	MCX E24x
Internal and external routing options	Inputs:32 Outputs:112	Inputs: 72 Outputs: 128

8.1.1 Considerations for migrating to MCX E24x

The TRGMUX module on MCX E24x provides a single-stage, centralized trigger routing system that simplifies the configuration compared to the two-stage TRGMUX architecture on KE1xF. While KE1xF uses a prestage (TRGMUX1) to multiplex a limited number of sources into TRGMUX0, MCX E24x consolidates the whole trigger routing into a unified matrix capable of supporting up to 72 input sources with direct mappings to peripheral targets. This not only streamlines the routing logic, but it also expands the range of available trigger sources and destinations. When migrating, review how the triggers are cascaded across both stages on KE1xF and adjust the configurations to leverage the single-stage routing model on MCX E24x, reducing complexity and improving determinism.

8.2 ADC

Analog-to-Digital Converters (ADCs) are essential components in many industrial applications that require precise analog signal sampling, such as motor control, sensor interfacing, or power monitoring. The KE1xF and MCX E24x families integrate SAR-based ADCs with comparable resolution, conversion modes, and hardware trigger support. While the architectural foundations are similar, there are a few implementation differences, such as the internal temperature sensing support. This section describes the key similarities and migration considerations between the ADC modules on both platforms.

Table 12. KE1xF and MCX E24x ADC differences

ADC	KE1xF	MCX E24x
Instances/channels	ADC_0: 16 ADC_1: 11-16 (following package) ADC_2: 11-16 (following package)	ADC_0: 14-32 (following package) ADC_1: 9-32 (following package)
Resolution	12-bit	12-bit
Temperature sensor	✓ Yes	✗ No

8.2.1 Considerations for migrating to MCX E24x

When migrating from the KE1xF family to the MCX E24x family, the ADC modules present a largely compatible architecture with a 12-bit resolution, multiple input channels, and similar trigger and conversion capabilities. However, there are two key differences. First, the number of available ADC channels may vary depending on the package, with MCX E24x offering up to 32 input channels per instance, which enable broader analog coverage in certain designs. Second, the KE1xF ADC supports an internal temperature sensor (accessible via a dedicated ADC channel), but this feature is not available on MCX E24x. Applications relying on internal thermal monitoring must therefore use an external temperature sensing solution when migrating.

8.3 FTM

The FlexTimer Module (FTM) is a versatile timer peripheral widely used in industrial and control applications that require precise pulse generation, capture, and modulation. It supports key functionalities such as PWM generation, input capture, output compare, and quadrature decoding. The KE1xF and MCX E24x families integrate FTM modules, enabling applications such as motor control, signal measurement, and actuator control. While the core capabilities remain similar in both families, there are differences in the number of channels, instances, and signal routing options, depending on the specific device and package. The following table summarizes the differences between the FTM on KE1xF and MCX E24x.

Table 13. KE1xF and MCX E24x timer modules differences

Timer modules	KE1xF	MCX E24x
FTM		
Instances and channels	4 FTMs, 8 channels each	4 to 8 FTMs, 8 channels each
Prescaler	Prescaler divide-by 1, 2, 4, 8, 16, 32, 64, or 128	Prescaler divide-by 1, 2, 4, 8, 16, 32, 64, or 128
Operation modes	• Input capture • Output compare • Edge-aligned PWM	• Input capture • Output compare • Edge-aligned PWM
Fault control	Up to four fault inputs for global fault control	Up to four fault inputs for global fault control

8.3.1 Considerations for migrating to MCX E24x

While the general functionality of the FTM module is preserved on both platforms, supporting edge-aligned PWM, center-aligned PWM, and input capture modes, there are some differences. The MCX E24x devices may integrate newer or more power-efficient timer implementations. The total number of FTM instances or channels per instance may vary depending on the selected package. The peripheral signal routing via the TRGMUX and pin multiplexing configuration may differ and require changes in software. A careful review of the channel count and TRGMUX mappings is recommended during migration planning to preserve timing behavior consistency.

8.4 PDB

The Programmable Delay Block (PDB) is a timer-based triggering module designed to coordinate and schedule the operation of analog peripherals, such as ADCs and DACs. It enables precise, hardware-controlled triggering of conversions, often synchronized with timer events, such as PWM edges. This is particularly useful in applications requiring deterministic sampling, such as motor control, power monitoring, or industrial signal processing. The KE1xF and MCX E24x families integrate the PDB module with comparable features, allowing for consistent timing behavior and synchronization across migrations.

The following table indicates the main differences.

Table 14. KE1xF and MCX E24x PDB differences

Timer modules	KE1xF	MCX E24x
PDB		
Trigger inputs	Up to 15 trigger input sources 1 software trigger source	Up to 2 trigger input sources 1 software trigger source
ADC hardware trigger	Up to 8 configurable PDB channels for the ADC hardware trigger	Up to 8 configurable PDB channels for the ADC hardware trigger

8.4.1 Considerations for migrating to MCX E24x

The Programmable Delay Block (PDB) IP is the same on both KE1xF and MCX E24x: a single module instance offering multiple programmable delay channels, hardware-trigger generation for ADC (and DAC), prescaler/divider configuration, and interrupt/DMA support. The migration is straightforward at the driver level. For example, all key registers (PDBx_MCR, PDBx_MOD, PDBx_CHnC1, PDBx_IDLY) retain identical offsets and bit fields. However, there are a few platform-specific details:

- Trigger routing: MCX E24x introduces tighter integration with its TRGMUX, allowing PDB triggers to feed directly into the system's flexible trigger mux, whereas KE1xF uses fixed trigger outputs that must be remapped via SIM settings.
- Clock and reset control: On KE1xF, the PDB clock and gating are managed by the SIM module, while MCX E24x uses the PCC. The migrating code must update the clock-enable calls accordingly.

Overall, the application code that configures PDB delays, prescalers, and channel triggers is largely unchanged; only the trigger-routing setup and peripheral-enable calls require an adjustment to match the MCX E24x's SIM→PCC differences and enhanced TRGMUX connectivity.

9 Communication modules

This section describes the communication modules.

9.1 Ethernet MAC

The MCX E24x family introduces an integrated 10/100 Mbit/s Ethernet MAC with IEEE-1588 support (available on MCX E247 derivatives), which is a significant enhancement compared to the KE1xF family that lacks native Ethernet capability. This addition enables real-time Ethernet-based industrial communication, supporting precise time synchronization. The presence of the Ethernet MAC makes the MCX E24x more suitable for connected industrial applications, diagnostics, and gateway functionalities, reducing the need for external Ethernet controllers in many use cases.

The following table provides a high-level overview of the MCX E24x Ethernet IP.

Table 15. Ethernet MAC on MCX E247

Features	MCX E247
Ethernet MAC controller	ENET
Supported speeds	10/100 Mbit/s
TX queues	1
RX queues	1
Network acceleration features	CRC generation and checking (Pv4, IPv6, TCP, UDP, ICMP) Additional padding processing Payload alignment (32-bit)
Address filtering	MAC address Multicast and unicast, based on a 64-bit hash filter
VLAN tags	RX frames: detection
Time stamping (IEEE 1588)	✓ Yes
Time Sensitive Networking (TSN)	✗ No

9.1.1 Considerations for migrating to MCX E24x

The KE1xF family does not feature an Ethernet MAC controller, whereas the MCX E247 introduces a 10/100 Mbit/s ENET controller with support for the IEEE 1588 time stamping. This significantly broadens the use cases for industrial and connected applications, enabling real-time Ethernet communication and protocol synchronization. The ENET controller includes basic network acceleration features, such as hardware CRC generation/checking and payload alignment, as well as multicast/unicast address filtering using a 64-bit hash table. However, advanced features such as Time-Sensitive Networking (TSN), VLAN tag insertion/replacement, and layer 3/4 filtering are not supported. VLAN tags are only detected on RX frames. Applications migrating from KE1xF to MCX E247 must therefore take into account the software implementation effort required to fully leverage Ethernet features, particularly for VLAN-based or time-sensitive communication.

9.2 FlexCAN

The KE1xF and MCX E24x families integrate the FlexCAN module, supporting CAN protocol communication in line with ISO 11898-1. Widely used in industrial and automotive systems, FlexCAN offers robust, deterministic messaging for real-time applications. While the core FlexCAN functionality remains similar on both platforms, supporting classical CAN and CAN FD, the MCX E24x introduces enhancements aligned with modern embedded system requirements. These include improved message buffer handling, enhanced filtering capabilities, and tighter integration with the system DMA and interrupt structures. Migrating from KE1xF to MCX E24x ensures continuity of CAN features, but it also offers the opportunity to optimize system performance and safety by leveraging the newer architecture's added features and configurability.

The following table summarizes the differences between FlexCAN on KE1xF and MCX E24x.

Table 16. Changes of the FlexCAN peripheral between KE1xF and MCX E24x

Features	KE1xF	MCX E24x
CAN-FD	✗ No	3 for MCX E247 2 for MCX E246 1 for MCX E245
Reception FIFO (RX FIFO)	For CAN 2.0 B frames	For CAN 2.0 B frames
Timestamping timer	16 bits wide with optional external time tick	16 bits wide with optional external time tick

Table 16. Changes of the FlexCAN peripheral between KE1xF and MCX E24x*...continued*

Features	KE1xF	MCX E24x
DMA transfers	Only for CAN 2.0 B	Only for CAN 2.0 B
Message buffers	16	Up to 32
Pretended Networking (PNET)	✗ No	Only for CAN 2.0 B

9.2.1 Considerations for migrating to MCX E24x

The migration from KE1xF to MCX E24x offers significant enhancements in CAN capabilities. While the KE1xF platform supports only Classic CAN (CAN 2.0 B), the MCX E24x family introduces the CAN-FD support, with up to three FlexCAN instances on the MCX E247. This enables higher data throughput and more efficient communication for industrial applications requiring frequent or large data exchanges. The number of message buffers increases from 16 to up to 32, allowing for improved buffering and message handling. The timestamping timer remains similar on both platforms, with a 16-bit width and optional external clock input. The DMA transfers and RX FIFO remain limited to CAN 2.0B frames and the Pretended Networking (PNET), useful for reducing power consumption, is now supported on MCX E24x, though still only for Classic CAN. The developers benefit from enhanced scalability and improved CAN peripheral flexibility when migrating to MCX E24x.

9.3 FlexIO

The FlexIO peripheral is a highly configurable serial and parallel interface module designed to emulate a wide variety of communication protocols such as UART, SPI, and I2C. Present on the KE1xF and MCX E24x families, FlexIO provides a flexible alternative to dedicated peripherals, especially useful in industrial applications where customization and scalability are required. Its capability to be reconfigured entirely through software allows developers to adapt to evolving protocol needs without changing hardware.

The following table outlines the key similarities and enhancements brought by the MCX E24x implementation compared to KE1xF.

Table 17. FlexIO differences

Features	KE1xF	MCX E24x
Number of pins	8	8
Number of timers	4	4
Number of shifter registers	4	4
Number of bits in counter	16	16
Interrupt request	✓ Yes	✓ Yes
DMA request	✓ Yes	✓ Yes

Table 18. FlexIO protocol support

Protocol	KE1xF	MCX E24x
UART	✓ Yes	✓ Yes
I2C	✓ Yes	✓ Yes
SPI	✓ Yes	✓ Yes
I2S	✓ Yes	✓ Yes
PWM	✓ Yes	✓ Yes

9.3.1 Considerations for migrating to MCX E24x

The FlexIO peripheral on MCX E24x devices provides a feature parity with KE1xF in terms of pin count, number of shifters, timers, and supported protocols. Both platforms support the emulation of UART, I2C, SPI, I2S, and PWM protocols through software configuration. As such, migrating FlexIO-based implementations from KE1xF to MCX E24x is straightforward, with no significant architectural or capability differences. Developers can reuse their existing FlexIO logic while benefiting from the enhanced peripheral integration and toolchain support of the MCX E24x family.

9.4 QuadSPI

Although QuadSPI (QSPI) is a communication protocol, its main use case is as a memory interface, providing greater possibilities to store and execute code from an external memory connected. The table below shows the main differences.

Table 19. KE1xF and MCX E247 QSPI differences

Feature	KE1xF	MCX E247
QSPI max communication clock	✗ No	Side A: up to 80 MHz Side B: up to 20 MHz
Data lines		4 for SDR and 8 for HyperRAM (bidirectional)
SDR support		✓ Yes
DDR support		✓ Yes: Only side B
AHB access read/write		✓ Yes/✓ Yes
AHB buffer		1 kB
QSPI and ENET simultaneously		✓ Yes

9.4.1 Considerations for migrating to MCX E24x

MCX E247 introduces a versatile QuadSPI module that significantly enhances external memory interfacing capabilities compared to the KE1xF family, which lacks such a peripheral. The MCX E247 QuadSPI supports both the SDR and DDR modes, with up to four data lines for standard SPI and 8 lines for the HyperRAM in a bidirectional mode. The interface provides two sides: side A supports clock frequencies up to 80 MHz, while side B operates at up to 20 MHz with a DDR capability. The QuadSPI interface allows AHB read and write access with a 1 kB internal buffer, offering efficient data transactions between the processor and external memory. QuadSPI can operate simultaneously with the Ethernet and SAI modules when the external voltage is within 3.3 V ± 10 %, ensuring reliable communication across these high-speed peripherals.

9.5 UART/LIN

The LPUART module includes registers to control the baud rate, select options, report status, and store transmit/receive data. Access to an address outside the valid memory map generates a bus error. The LPUART supports full-duplex, asynchronous, NRZ serial communication and comprises a baud rate generator, a transmitter, and a receiver block. The transmitter and receiver operate independently, although they use the same baud rate generator. The following table describes each block of the LPUART module.

Table 20. LPUART functional blocks

Feature	KE1xF	MCX E24x
LIN support	✓ Yes	✓ Yes

Table 20. LPUART functional blocks*...continued*

Feature	KE1xF	MCX E24x
LIN master	✗ No	✓ Yes
LIN slave	✗ No	✓ Yes
Autosynchronization	✓ Yes	✓ Yes
RXFIFO	4 words	4 words
TXFIFO	4 words	4 words
Parity	Even/odd	Even/odd
DMA support	✓ Yes	✓ Yes
Number of instances	3	3

9.5.1 Considerations for migrating to MCX E24x

When migrating from KE1xF to the MCX E24x family, the LPUART module maintains a similar structure, with three instances, four-word TX and RX FIFOs, support for even and odd parity, and DMA support. However, the MCX E24x family introduces extended functionality, most notably the native support for both the LIN master and LIN slave modes, which are not fully supported on KE1xF (only basic LIN communication primitives are available). Automatic resynchronization is supported on both families, helping to maintain communication robustness in systems subject to clock drift. Overall, the enhancements in MCX E24x make the LPUART module more capable for serial communication in applications such as automotive or industrial systems requiring full LIN support.

9.6 SPI

The LPSPI is a low-power Serial Peripheral Interface (SPI) module that supports an efficient interface to an SPI bus, either as a master and/or as a slave.

Table 21. LPSPI features

Feature	KE1xF	MCX E24X
Word size	32 bits	32 bits
Slave operation	✓ Yes	✓ Yes
Master operation	✓ Yes	✓ Yes
DMA	✓ Yes	✓ Yes
Command/transmit	4 words	4 words
Receive FIFO	4 words	4 words

9.6.1 Considerations for migrating to MCX E24x

The SPI modules on the KE1xF and MCX E24x families are very similar, ensuring a smooth migration path. Both families support master and slave modes, DMA for data transfers, and 32-bit word size. The command/transmit and receive FIFOs are four words deep in both cases, offering consistent buffering capabilities. As the register structure and core functionalities remain aligned, software migration typically requires minimal changes. Review instance-specific details, such as pin multiplexing and clock sources, to ensure full compatibility on the MCX E24x platform.

9.7 LPI2C

LPI2C is a low-power Inter-Integrated Circuit (I2C) module that supports an efficient interface to an I2C bus as a master and/or as a slave.

- The LPI2C implements logic support for the standard-mode, fast-mode, fast-mode plus, and ultrafast modes of operation.
- The LPI2C is designed to use little CPU overhead, with the DMA offloading of FIFO register accesses.

The LPI2C module also complies with the System Management Bus (SMBus) specification, version 2. The SMBus is a single-ended, simple, two-wire bus, which is typically used for low-bandwidth communications.

Table 22. LPI2C features

Feature	KE1xF	MCX E24x
I2C max communication speed (slave mode)	3.4 MHz	3.4 MHz
I2C max communication speed (master mode)	5 MHz	5 MHz
SMBus support	✓ Yes	✓ Yes
RXFIFO	4 words	4 words
TXFIFO	4 words	4 words
DMA support	✓ Yes	✓ Yes
Number of instances	2	3

9.7.1 Considerations for migrating to MCX E24x

When migrating from the KE1xF family to the MCX E24x series, the LPI2C module offers improved communication performance and enhanced flexibility. MCX E24x supports higher I2C communication speeds (up to 12 MHz in both the master and slave modes) when compared to the 5 MHz maximum supported by KE1xF. This allows faster data transfers, particularly beneficial in applications requiring high-speed sensor or peripheral communication. Both families support the SMBus protocols and provide RX and TX FIFOs of four words each, along with DMA support for efficient data handling. The MCX E24x family also increases the number of LPI2C instances from two (on KE1xF) to three, enabling more simultaneous I2C connections. These improvements can reduce communication bottlenecks and allow for more complex and scalable designs.

9.8 SAI

The Synchronous Audio Interface (SAI) module provides an interface that supports full-duplex serial interfaces with frame synchronization, different protocol support (I2S, AC97, TDM), and codec/DSP interfaces.

Table 23. SAI features

Feature	KE1xF	MCX E247
Word size	✗ No	Between 8 and 32 bits
Number of communications channels		4 of SAI0 and 1 of SAI1
Frame synchronization		I2S/AC97/TDM/Codec-IF/DSP-IF/TDM
DMA		✓ Yes
TX FIFO size		8 x 32-bit
Maximum frame size per data line		16 words

Table 23. SAI features*...continued*

Feature	KE1xF	MCX E247
Working in low-power mode		OFF

9.8.1 Considerations for migrating to MCX E24x

MCX E247 has a significantly enhanced Serial Audio Interface (SAI) peripheral. Unlike the KE1xF, which lacks an SAI module, the MCX E247 includes two SAI modules. SAI0 has four communication channels and SAI1 has one channel, offering flexible audio interface options for industrial and multimedia applications. It supports word sizes configurable from 8 to 32 bits and it can synchronize over various interface protocols, such as I2S, AC97, codec-IF, DSP-IF, and TDM. The SAI includes DMA support and an 8×32-bit FIFO per a transmit channel, allowing robust buffering and low CPU load during streaming. The maximum frame size per a data line is 16 words. While the SAI is not currently functional in low-power modes, its full DMA-based throughput and configurable frame synchronization make it a strong addition to the MCX E247. Update your audio driver initialization and buffer-management logic to align with the deeper FIFO and wider protocol support.

10 Pin management and characteristics

The KE1xF and MCX E24x families support 64 LQFP and 100 LQFP packages with similar pinouts and signal assignments. This ensures a smooth hardware migration path with no need for PCB redesign when upgrading between equivalent LQFP packages.

However, while the pinout remains similar, some peripheral functions (for example, QSPI and Ethernet) may differ between the two families. Certain pins may be multiplexed differently or reserved for peripherals not present on KE1xF. For this reason, a detailed pinout comparison is shown in Table 24 and Table 25 to help evaluate function-level compatibility in your specific use case.

Note: *The ALT1 column is omitted, as it is identical on both devices.*

Table 24. LQFP100 pinout comparison

100 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
1	PTE16	DISABLED	-	LPUART1_RTS[1]	LPSPI2_SIN[1]	FTM2_CH7	FTM4_FLT0[1]	FXIO_D3	TRGMUX_OUT7
2	PTE15	DISABLED	-	LPUART1_CTS[1]	LPSPI2_SCK[1]	FTM2_CH6	FTM4_FLT1[1]	FXIO_D2	TRGMUX_OUT6
3	PTD1	ADC2_SE1[2] DISABLED[1]	ADC2_SE1[2]	FTM0_CH3	LPSPI1_SIN	FTM2_CH1	SAI0_MCLK[1]	FXIO_D1	TRGMUX_OUT2
4	PTD0	ADC2_SE0[2] DISABLED[1]	ADC2_SE0[2]	FTM0_CH2	LPSPI1_SCK	FTM2_CH0	ETM_TRACE_D0[1]	FXIO_D0	TRGMUX_OUT1
5	PTE11	ADC2_SE13[2] DISABLED[1]	ADC2_SE13[2]	PWT_IN1[2] LPSPI2_PCS0[1]	LPTMR0_ALT1	FTM2_CH5	-	FXIO_D5	TRGMUX_OUT5
6	PTE10	ADC2_SE12[2] DISABLED[1]	ADC2_SE12[2]	CLKOUT	LPSPI2_PCS1[1]	FTM2_CH4	-	FXIO_D4	TRGMUX_OUT4
7	PTE13	DISABLED	-	FTM4_CH5[1]	LPSPI2_PCS2[1]	FTM2_FLT0	-	-	-
8	PTE5	DISABLED	-	TCLK2	FTM2_QD_PHA	FTM2_CH3	CAN0_TX	FXIO_D7	EWM_IN
9	PTE4	DISABLED	-	BUSOUT[2] ETM_TRACE_D1[1]	FTM2_QD_PHB	FTM2_CH2	CAN0_RX	FXIO_D6	EWM_OUT_b
10	VDD	VDD	VDD	-	-	-	-	-	-
11	VDDA	VDDA	VDDA	-	-	-	-	-	-
12	VREFH	VREFH	VREFH	-	-	-	-	-	-
13	VREFL	VREFL	VREFL	-	-	-	-	-	-
14	VSS	VSS	VSS	-	-	-	-	-	-

Table 24. LQFP100 pinout comparison*...continued*

100 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
15	PTB7	EXTAL	EXTAL	LPI2C0_SCL	-	-	-	-	-
16	PTB6	XTAL	XTAL	LPI2C0_SDA	-	-	-	-	-
17	PTE14	ACMP2_IN3[2] DISABLED[1]	ACMP2_IN3[2]	FTM0_FLT1	-	FTM2_FLT1	-	-	-
18	PTE3	DISABLED	-	FTM0_FLT0	LPUART2_RTS	FTM2_FLT0	-	TRGMUX_IN6	ACMP2_OUT[2] CMP0_OUT[1]
19	PTE12	DISABLED	-	FTM0_FLT3	LPUART2_TX	FTM5_FLT0[1]	-	-	-
20	PTD17	DISABLED	-	FTM0_FLT2	LPUART2_RX	FTM5_FLT1[1]	-	-	-
21	PTD16	ACMP2_IN0[2] DISABLED[1]	ACMP2_IN0[2]	FTM0_CH1	ETM_TRACE_D2[1]	LPSPI0_SIN[1]	CMP0_RRT[1]	ETM_TRACE_CLKOUT[1]	-
22	PTD15	ACMP2_IN1[2] DISABLED[1]	ACMP2_IN1[2]	FTM0_CH0	ETM_TRACE_D3[1]	LPSPI0_SCK[1]	ENET_TMR2[1]	-	-
23	PTE9	ACMP2_IN2/DAC0_OUT[2] DISABLED[1]	ACMP2_IN2/DAC0_OUT[2]	FTM0_CH7	LPUART2_CTS	-	ENET_TMR3[1]	-	-
24	PTD14	DISABLED	-	FTM2_CH5	LPUART1_TX[1]	-	ENET_TMR0[1]	-	CLKOUT
25	PTD13	DISABLED	-	FTM2_CH4	LPUART1_RX[1]	-	ENET_TMR1[1]	-	RTC_CLKOUT
26	PTE8	ACMP0_IN3[2] CMP0_IN3[1]	ACMP0_IN3[2] CMP0_IN3[1]	FTM0_CH6	-	-	MII_RMII_MDC[1]	-	-
27	PTB5	DISABLED	-	FTM0_CH5	LPSPI0_PCS1	LPSPI0_PCS0[1]	CLKOUT[1]	TRGMUX_IN0	ACMP1_OUT[2] MII_RMII_MDC[1]
28	PTB4	ACMP1_IN2[2] DISABLED[1]	ACMP1_IN2[2]	FTM0_CH4	LPSPI0_SOUT	-	MII_RMII_MDIO[1]	TRGMUX_IN1	-
29	PTC3	ACMP0_IN4/ EXTAL32[2]/ ADC0_SE11/ CMP0_IN4[1]	ACMP0_IN4/ EXTAL32[2]/ ADC0_SE11/ CMP0_IN4[1]	FTM0_CH3	CAN0_TX	LPUART0_TX[1]	MII_TX_ER[1]	-	-
30	PTC2	ACMP0_IN5/XTAL32[2]/ ADC0_SE10/ CMP0_IN5[1]	ACMP0_IN5/XTAL32[2]/ ADC0_SE10/ CMP0_IN5[1]	FTM0_CH2	CAN0_RX	LPUART0_RX[1]	MII_RMII_TXD[0][1]	ETM_TRACE_CLKOUT[1]	-
31	PTD7	DISABLED[2] CMP0_IN6[1]	CMP0_IN6[1]	LPUART2_TX	-	FTM2_FLT3	MII_RMII_TXD[1][1]	ETM_TRACE_D0[1]	-
32	PTD6	DISABLED[2] CMP0_IN7[1]	CMP0_IN7[1]	LPUART2_RX	-	FTM2_FLT2	[1]MII_TXD2	-	-
33	PTD5	DISABLED	-	FTM2_CH3	LPTMR0_ALT2	FTM2_FLT1	PWT_IN2[2] MII_TXD3[1]	TRGMUX_IN7	-
34	PTD12	DISABLED	-	FTM2_CH2	LPI2C1_HREQ	ETM_TRACE_D1[1]	MII_RMII_TX_EN[1]	LPUART2_RTS	-
35	PTD11	DISABLED	-	FTM2_CH1	FTM2_QD_PHA	ETM_TRACE_D2[1]	MII_RMII_TX_CLK[1]	LPUART2_CTS	-
36	PTD10	DISABLED	-	FTM2_CH0	FTM2_QD_PHB	ETM_TRACE_D3[1]	MII_RX_CLK[1]	CLKOUT[1]	-
37	VSS	VSS	VSS	-	-	-	-	-	-
38	VDD	VDD	VDD	-	-	-	-	-	-
39	PTC1	ACMP1_IN3[2]/ ADC0_SE9	ACMP1_IN3[2]/ ADC0_SE9	FTM0_CH1	LPSPI2_SOUT[1]	MII_RMII_RXD[1][1]	MII_RMII_RXD[0][1]	FTM1_CH7	-
40	PTC0	ACMP1_IN4[2]/ ADC0_SE8	ACMP1_IN4[2]/ ADC0_SE8	FTM0_CH0	LPSPI2_SIN[1]	MII_RMII_RXD[1][1]	MII_RMII_RXD[0][1]	FTM1_CH6	-
41	PTD9	ACMP1_IN5[2] DISABLED[1]	ACMP1_IN5[2]	LPI2C1_SCL	FXIO_D0[1]	FTM2_FLT3	MII_RXD2[1]	FTM1_CH5	-
42	PTD8	DISABLED	-	LPI2C1_SDA	MII_RXD3[1]	FTM2_FLT2	FXIO_D1[1]	FTM1_CH4	-

Table 24. LQFP100 pinout comparison...*continued*

100 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
43	PTC17	ADC0_SE15	ADC0_SE15	FTM1_FLT3	CAN2_TX[1]	LPI2C1_SCLS	MII_RMII_RX_DV[1]	-	-
44	PTC16	ADC0_SE14	ADC0_SE14	FTM1_FLT2	CAN2_RX[1]	LPI2C1_SDAS	MII_RMII_RX_ER[1]	-	-
45	PTC15	ACMP2_IN4[2]/ ADC0_SE13	ACMP2_IN4[2]/ ADC0_SE13	FTM1_CH3	LPSPI2_SCK[1]	MII_CRS[1]	-	TRGMUX_IN8[1]	-
46	PTC14	ACMP2_IN5[2]/ ADC0_SE12	ACMP2_IN5[2]/ ADC0_SE12	FTM1_CH2	LPSPI2_PCS0[1]	MII_COL[1]	-	TRGMUX_IN9[1]	-
47	PTB3	ADC0_SE7	ADC0_SE7	FTM1_CH1	LPSPI0_SIN	FTM1_QD_PHA	-	TRGMUX_IN2	-
48	PTB2	ADC0_SE6	ADC0_SE6	FTM1_CH0	LPSPI0_SCK	FTM1_QD_PHB	-	TRGMUX_IN3	-
49	PTC13	DISABLED	-	FTM3_CH7	FTM2_CH7	LPUART2_RTS[1]	-	-	-
50	PTC12	DISABLED	-	FTM3_CH6	FTM2_CH6	LPUART2_CTS[1]	-	-	-
51	PTC11	DISABLED	-	FTM3_CH5	FTM4_CH2[1]	-	-	TRGMUX_IN10[1]	-
52	PTC10	DISABLED	-	FTM3_CH4	-	-	-	TRGMUX_IN11[1]	-
53	PTB1	ADC0_SE5/ ADC1_SE15[1]	ADC0_SE5/ ADC1_SE15[1]	LPUART0_TX	LPSPI0_SOUT	TCLK0	CAN0_TX[1]	FTM4_CH5[1]	-
54	PTB0	ADC0_SE4/ ADC1_SE14[1]	ADC0_SE4/ ADC1_SE14[1]	LPUART0_RX	LPSPI0_PCS0	LPTMR0_ALT3	PWT_IN3[2] CAN0_RX[1]	FTM4_CH6[1]	-
55	PTC9	ADC2_SE15[2] DISABLED[1]	ADC2_SE15[2]	LPUART1_TX	FTM1_FLT1	FTM5_CH0[1]	-	LPUART0_RTS	-
56	PTC8	ADC2_SE14[2] DISABLED[1]	ADC2_SE14[2]	LPUART1_RX	FTM1_FLT0	FTM5_CH1[1]	-	LPUART0_CTS	-
57	PTA7	ACMP1_IN1[2]/ ADC0_SE3	ACMP1_IN1[2]/ ADC0_SE3	FTM0_FLT2	FTM5_CH3[1]	RTC_CLKIN	-	LPUART1_RTS	-
58	PTA6	ACMP1_IN0[2]/ ADC0_SE2	ACMP1_IN0[2]/ ADC0_SE2	FTM0_FLT1	LPSPI1_PCS1	FTM5_CH5[1]	-	LPUART1_CTS	-
59	PTE7	ADC2_SE2[2]/ACMP2_IN6[2] DISABLED[1]	ADC2_SE2[2]/ACMP2_IN6[2]	FTM0_CH7	FTM3_FLT0	-	-	-	-
60	VSS	VSS	VSS	-	-	-	-	-	-
61	VDD	VDD	VDD	-	-	-	-	-	-
62	PTA17	DISABLED	-	FTM0_CH6	FTM3_FLT0	EWM_OUT_b	FTM5_FLT0[1]	-	-
63	PTB17	ADC2_SE3[2] DISABLED[1]	ADC2_SE3[2]	FTM0_CH5	LPSPI1_PCS3	FTM5_FLT1[1]	-	-	-
64	PTB16	ADC1_SE15	ADC1_SE15	FTM0_CH4	LPSPI1_SOUT	-	-	-	-
65	PTB15	ADC1_SE14	ADC1_SE14	FTM0_CH3	LPSPI1_SIN	-	-	-	-
66	PTB14	ADC1_SE9/ ADC0_SE9[1]	ADC1_SE9/ ADC0_SE9[1]	FTM0_CH2	LPSPI1_SCK	-	-	-	-
67	PTB13	ADC1_SE8/ ADC0_SE8[1]	ADC1_SE8/ ADC0_SE8[1]	FTM0_CH1	FTM3_FLT1	CAN2_TX[1]	FTM6_FLT0[1]	-	-
68	PTB12	ADC1_SE7	ADC1_SE7	FTM0_CH0	FTM3_FLT2	CAN2_RX[1]	FTM6_FLT1[1]	-	-
69	PTD4	ACMP1_IN6[2]/ ADC1_SE6	ACMP1_IN6[2]/ ADC1_SE6	FTM0_FLT3	FTM3_FLT3	-	-	-	-
70	PTD3	NMI_b[2] ADC1_SE3[1]	ADC1_SE3	FTM3_CH5	LPSPI1_PCS0	FXIO_D5	FXIO_D7[1]	TRGMUX_IN4	NMI_b
71	PTD2	ADC1_SE2	ADC1_SE2	FTM3_CH4	LPSPI1_SOUT	FXIO_D4	FXIO_D6[1]	TRGMUX_IN5	-
72	PTA3	ADC1_SE1	ADC1_SE1	FTM3_CH1	LPI2C0_SCL	EWM_IN	FXIO_D5[1]	LPUART0_TX	-
73	PTA2	ADC1_SE0	ADC1_SE0	FTM3_CH0	LPI2C0_SDA	EWM_OUT_b	FXIO_D4[1]	LPUART0_RX	-

Table 24. LQFP100 pinout comparison...*continued*

100 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
74	PTB11	ADC2_SE8[2] DISABLED[1]	ADC2_SE8[2]	FTM3_CH3	LPI2C0_ HREQ	-	-	-	-
75	PTB10	ADC2_SE9[2] DISABLED[1]	ADC2_SE9[2]	FTM3_CH2	LPI2C0_ SDAS	SAI1_ MCLK[1]	-	-	-
76	PTB9	ADC2_SE10[2] DISABLED[1]	ADC2_SE10[2]	FTM3_CH1	LPI2C0_ SCLS	SAI1_D0[1]	-	-	-
77	PTB8	ADC2_SE11[2] DISABLED[1]	ADC2_SE11[2]	FTM3_CH0	-	SAI1_ BCLK[1]	-	-	-
78	PTA1	ACMP0_IN1[2]/ ADC0_SE1/ CMP0_IN1[1]	ACMP0_IN1[2]/ ADC0_SE1/ CMP0_IN1[1]	FTM1_CH1	LPI2C0_ SDAS	FXIO_D3	FTM1_QD_ PHA	LPUART0_RTS	TRGMUX_OUT0
79	PTA0	ACMP0_IN0[2]/ ADC0_SE0/ CMP0_IN0[1]	ACMP0_IN0[2]/ ADC0_SE0/ CMP0_IN0[1]	FTM2_CH1	LPI2C0_ SCLS	FXIO_D2	FTM2_QD_ PHA	LPUART0_CTS	TRGMUX_OUT3
80	PTC7	ADC1_SE5	ADC1_SE5	LPUART1_ TX	CAN1_TX	FTM3_CH3	-	FTM1_QD_ PHA[1]	-
81	PTC6	ADC1_SE4	ADC1_SE4	LPUART1_ RX	CAN1_RX	FTM3_CH2	-	FTM1_QD_ PHB[1]	-
82	PTA16	ADC1_SE13	ADC1_SE13	FTM1_CH3	LPSPI1_ PCS2	-	-	-	-
83	PTA15	ADC1_SE12	ADC1_SE12	FTM1_CH2	LPSPI0_ PCS3	LPSPI2_ PCS3[1]	FTM7_ FLT0[1]	-	-
84	PTE6	ACMP0_IN6[2]/ ADC1_SE11	ACMP0_IN6[2]/ ADC1_SE11	LPSPI0_ PCS2	FTM7_ FLT1[1]	FTM3_CH7	-	LPUART1_RTS	-
85	PTE2	ADC1_SE10	ADC1_SE10	LPSPI0_ SOUT	LPTMR0_ ALT3	FTM3_CH6	PWT_IN3[2]	LPUART1_CTS	SAI1_SYNC[1]
86	VSS	VSS	VSS	-	-	-	-	-	-
87	VDD	VDD	VDD	-	-	-	-	-	-
88	PTA14	DISABLED	-	FTM0_FLT0	FTM3_FLT1	EWM_IN	-	FTM1_FLT0	BUSOUT[2] SAI0_D3[1]
89	PTA13	ADC2_SE4[2] DISABLED[1]	ADC2_SE4[2]	FTM1_CH7	CAN1_TX	LPI2C1_ SCLS	-	FTM2_QD_ PHA[1]	SAI0_D0[1]
90	PTA12	ADC2_SE5[2] DISABLED[1]	ADC2_SE5[2]	FTM1_CH6	CAN1_RX	LPI2C1_ SDAS	-	FTM2_QD_ PHB[1]	SAI0_BCLK[1]
91	PTA11	DISABLED	-	FTM1_CH5	LPUART0_ RX[2]	FXIO_D1	CMP0_ RRT[1]	SAI0_SYNC[1]	-
92	PTA10	JTAG_TDO/noetm_ TRACE_SWO	-	FTM1_CH4	LPUART0_ TX[2]	FXIO_D0	-	-	JTAG_TDO/noetm_ TRACE_SWO
93	PTE1	ADC2_SE6[2] DISABLED[1]	ADC2_SE6[2]	LPSPI0_ SIN	LPI2C0_ HREQ	LPI2C1_SCL	LPSPI1_ PCS0[1]	FTM1_FLT1	SAI0_D1[1]
94	PTE0	ADC2_SE7[2] DISABLED[1]	ADC2_SE7[2]	LPSPI0_ SCK	TCLK1	LPI2C1_ SDA	LPSPI1_ SOUT[1]	FTM1_FLT2	SAI0_D2[1]
95	PTC5	JTAG_TDI	-	FTM2_CH0	RTC_ CLKOUT	LPI2C1_ HREQ	-	FTM2_QD_ PHB	JTAG_TDI
96	PTC4	JTAG_TCLK/SWD_CLK	ACMP0_IN2[2] CMP0_IN2[1]	FTM1_CH0	RTC_ CLKOUT	-	EWM_IN	FTM1_QD_ PHB	JTAG_TCLK/SWD_ CLK
97	PTA5	RESET_b	-	-	TCLK1	-	-	JTAG_TRST_ b[2]	RESET_b
98	PTA4	JTAG_TMS/SWD_DIO	-	-	-	ACMP0_ OUT[2] CMP0_ OUT[1]	EWM_OUT_ b	-	JTAG_TMS/SWD_ DIO
99	PTA9	DISABLED	-	LPUART2_ TX[1]	LPSPI2_ PCS0[1]	FXIO_D7	FTM3_FLT2	FTM1_FLT3	FTM4_FLT0[1]
100	PTA8	DISABLED	-	LPUART2_ RX[1]	LPSPI2_ SOUT[1]	FXIO_D6	FTM3_FLT3	FTM4_FLT1[1]	-

[1] Pin assignment differs on MCX E24x
[2] Pin assignment differs on KE1xF

Table 25. LQFP64 pinout comparison

64 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
10[1]	VREFL/ VSS[1]	VREFL/VSS[1]	VREFL/VSS[1]	-	-	-	-	-	-
1	PTD1	ADC2_SE1[1] DISABLED[2]	ADC2_SE1[1]	FTM0_CH3	LPSPI1_ SIN	FTM2_CH1	-	FXIO_D1	TRGMUX_OUT2
2	PTD0	ADC2_SE0[1] DISABLED[2]	ADC2_SE0[1]	FTM0_CH2	LPSPI1_ SCK	FTM2_CH0	-	FXIO_D0	TRGMUX_OUT1
3	PTE11	ADC2_SE13[1] DISABLED[2]	ADC2_SE13[1]	PWT_IN1[1] LPSPI2_ PCS0[2]	LPTMR0_ ALT1	FTM2_CH5	-	FXIO_D5	TRGMUX_OUT5
4	PTE10	ADC2_SE12[1] DISABLED[2]	ADC2_SE12[1]	CLKOUT	LPSPI2_ PCS1[2]	FTM2_CH4	-	FXIO_D4	TRGMUX_OUT4
5	PTE5	DISABLED	-	TCLK2	FTM2_QD_ PHA	FTM2_CH3	CAN0_TX	FXIO_D7	EWM_IN
6	PTE4	DISABLED	-	BUSOUT[1]	FTM2_QD_ PHB	FTM2_CH2	CAN0_RX	FXIO_D6	EWM_OUT_b
7	VDD	VDD	VDD	-	-	-	-	-	-
8	VDDA	VDDA	VDDA	-	-	-	-	-	-
9	VREFH	VREFH	VREFH	-	-	-	-	-	-
10[2]	VSS	VSS	VSS	-	-	-	-		-
11	PTB7	EXTAL	EXTAL	LPI2C0_ SCL	-	-	-	-	-
12	PTB6	XTAL	XTAL	LPI2C0_ SDA	-	-	-	-	-
13	PTE3	DISABLED	-	FTM0_FLT0	LPUART2_ RTS	FTM2_FLT0	-	TRGMUX_IN6	ACMP2_OUT[1] CMP0_OUT[2]
14	PTD16	ACMP2_IN0[1] DISABLED[2]	ACMP2_IN0[1]	FTM0_CH1	-	LPSPI0_ SIN[2]	CMP0_ RRT[2]	-	-
15	PTD15	ACMP2_IN1[1] DISABLED[2]	ACMP2_IN1[1]	FTM0_CH0	-	LPSPI0_ SCK[2]	-	-	-
16	PTE9	ACMP2_IN2[1]/DAC0_ OUT[1] DISABLED[2]	ACMP2_IN2[1]/DAC0_ OUT[1]	FTM0_CH7	LPUART2_ CTS	-	-	-	-
17	PTE8	ACMP0_IN3[1] CMP0_IN3[2]	ACMP0_IN3[1] CMP0_IN3[2]	FTM0_CH6	-	-	-	-	-
18	PTB5	DISABLED	-	FTM0_CH5	LPSPI0_ PCS1	LPSPI0_ PCS0[2]	CLKOUT[2]	TRGMUX_IN0	ACMP1_OUT[1]
19	PTB4	ACMP1_IN2[1] DISABLED[2]	ACMP1_IN2[1]	FTM0_CH4	LPSPI0_ SOUT	-	-	TRGMUX_IN1	-
20	PTC3	ACMP0_IN4[1]/ EXTAL32[1]/ ADC0_SE11/ CMP0_IN4[2]	ACMP0_IN4[1]/ EXTAL32[1]/ ADC0_SE11/ CMP0_IN4[2]	FTM0_CH3	CAN0_TX	LPUART0_ TX[2]	-	-	-
21	PTC2	ACMP0_IN5[1]/ XTAL32[1]/ ADC0_SE10/ CMP0_IN5[2]	ACMP0_IN5[1]/ XTAL32[1]/ ADC0_SE10/ CMP0_IN5[2]	FTM0_CH2	CAN0_RX	LPUART0_ RX[2]	-	-	-
22	PTD7	DISABLED[1] CMP0_IN6[2]	CMP0_IN6[2]	LPUART2_ TX	-	FTM2_FLT3	-	-	-
23	PTD6	DISABLED[1] CMP0_IN7[2]	CMP0_IN7[2]	LPUART2_ RX	-	FTM2_FLT2	-	-	-
24	PTD5	DISABLED	-	FTM2_CH3	LPTMR0_ ALT2	FTM2_FLT1	PWT_IN2[1] MII_TXD3[2]	TRGMUX_IN7	-
40[2]	VSS	VSS	VSS	-	-	-	-	-	-
25	PTC1	ACMP1_IN3[1]/ ADC0_SE9	ACMP1_IN3[1]/ ADC0_SE9	FTM0_CH1	LPSPI2_ SOUT[2]	-	-	FTM1_CH7	-
26	PTC0	ACMP1_IN4[1]/ ADC0_SE8	ACMP1_IN4[1]/ ADC0_SE8	FTM0_CH0	LPSPI2_ SIN[2]	-	-	FTM1_CH6	-
27	PTC17	ADC0_SE15	ADC0_SE15	FTM1_FLT3	CAN2_TX[2]	LPI2C1_ SCLS[1]	-	-	-
28	PTC16	ADC0_SE14	ADC0_SE14	FTM1_FLT2	CAN2_RX[2]	LPI2C1_ SDAS[1]	-	-	-

Table 25. LQFP64 pinout comparison*...continued*

64 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
29	PTC15	ACMP2_IN4[1]/ ADC0_SE13	ACMP2_IN4[1]/ ADC0_SE13	FTM1_CH3	LPSPI2_ SCK[2]	-	-	TRGMUX_ IN8[2]	-
30	PTC14	ACMP2_IN5[1]/ ADC0_SE12	ACMP2_IN5[1]/ ADC0_SE12	FTM1_CH2	LPSPI2_ PCS0[2]	-	-	TRGMUX_ IN9[2]	-
31	PTB3	ADC0_SE7	ADC0_SE7	FTM1_CH1	LPSPI0_ SIN	FTM1_QD_ PHA	-	TRGMUX_IN2	-
32	PTB2	ADC0_SE6	ADC0_SE6	FTM1_CH0	LPSPI0_ SCK	FTM1_QD_ PHB	-	TRGMUX_IN3	-
33	PTB1	ADC0_SE5/ ADC1_SE15[2]	ADC0_SE5/ ADC1_SE15[2]	LPUART0_ TX	LPSPI0_ SOUT	TCLK0	CAN0_TX[2]	-	-
34	PTB0	ADC0_SE4/ ADC1_SE14[2]	ADC0_SE4/ ADC1_SE14[2]	LPUART0_ RX	LPSPI0_ PCS0	LPTMR0_ ALT3	PWT_IN3[1] CAN0_RX[2]	-	-
35	PTC9	ADC2_SE15[1] DISABLED[2]	ADC2_SE15[1]	LPUART1_ TX	FTM1_FLT1	-	-	LPUART0_RTS	-
36	PTC8	ADC2_SE14[1] DISABLED[2]	ADC2_SE14[1]	LPUART1_ RX	FTM1_FLT0	-	-	LPUART0_CTS	-
37	PTA7	ACMP1_IN1[1]/ ADC0_SE3	ACMP1_IN1[1]/ ADC0_SE3	FTM0_FLT2	-	RTC_CLKIN	-	LPUART1_RTS	-
38	PTA6	ACMP1_IN0[1]/ ADC0_SE2	ACMP1_IN0[1]/ ADC0_SE2	FTM0_FLT1	LPSPI1_ PCS1	-	-	LPUART1_CTS	-
39	PTE7	ADC2_SE2[1]/ACMP2_ IN6[1] DISABLED[2]	ADC2_SE2[1]/ACMP2_ IN6[1]	FTM0_CH7	FTM3_FLT0	-	-	-	-
40[1]	VSS	VSS	VSS	-	-	-	-	-	-
41	VDD	VDD	VDD	-	-	-	-	-	-
42	PTB13	ADC1_SE8/ ADC0_SE8[2]	ADC1_SE8/ ADC0_SE8[2]	FTM0_CH1	FTM3_FLT1	CAN2_TX[2]	-	-	-
43	PTB12	ADC1_SE7	ADC1_SE7	FTM0_CH0	FTM3_FLT2	CAN2_RX[2]	-	-	-
44	PTD4	ACMP1_IN6[1]/ ADC1_SE6	ACMP1_IN6[1]/ ADC1_SE6	FTM0_FLT3	FTM3_FLT3	-	-	-	-
45	PTD3	NMI_b[1] ADC1_SE3[2]	ADC1_SE3	FTM3_CH5	LPSPI1_ PCS0	FXIO_D5	FXIO_D7[2]	TRGMUX_IN4	NMI_b
46	PTD2	ADC1_SE2	ADC1_SE2	FTM3_CH4	LPSPI1_ SOUT	FXIO_D4	FXIO_D6[2]	TRGMUX_IN5	-
47	PTA3	ADC1_SE1	ADC1_SE1	FTM3_CH1	LPI2C0_ SCL	EWM_IN	FXIO_D5[2]	LPUART0_TX	-
48	PTA2	ADC1_SE0	ADC1_SE0	FTM3_CH0	LPI2C0_ SDA	EWM_OUT_ b	FXIO_D4[2]	LPUART0_RX	-
49	PTA1	ACMP0_IN1[1]/ ADC0_SE1/ CMP0_IN1[2]	ACMP0_IN1[1]/ ADC0_SE1/ CMP0_IN1[2]	FTM1_CH1	LPI2C0_ SDAS	FXIO_D3	FTM1_QD_ PHA	LPUART0_RTS	TRGMUX_OUT0
50	PTA0	ACMP0_IN0[1]/ ADC0_SE0/ CMP0_IN0[2]	ACMP0_IN0[1]/ ADC0_SE0/ CMP0_IN0[2]	FTM2_CH1	LPI2C0_ SCLS	FXIO_D2	FTM2_QD_ PHA	LPUART0_CTS	TRGMUX_OUT3
51	PTC7	ADC1_SE5	ADC1_SE5	LPUART1_ TX	CAN1_TX	FTM3_CH3	-	FTM1_QD_ PHA[2]	-
52	PTC6	ADC1_SE4	ADC1_SE4	LPUART1_ RX	CAN1_RX	FTM3_CH2	-	FTM1_QD_ PHB[2]	-
53	PTE6	ACMP0_IN6[1]/ ADC1_SE11	ACMP0_IN6[1]/ ADC1_SE11	LPSPI0_ PCS2	-	FTM3_CH7	-	LPUART1_RTS	-
54	PTE2	ADC1_SE10	ADC1_SE10	LPSPI0_ SOUT	LPTMR0_ ALT3	FTM3_CH6	PWT_IN3[1]	LPUART1_CTS	-
55	PTA13	ADC2_SE4[1] DISABLED[2]	ADC2_SE4[1]	FTM1_CH7	CAN1_TX	LPI2C1_ SCLS[1]	-	FTM2_QD_ PHA[2]	-
56	PTA12	ADC2_SE5[1] DISABLED[2]	ADC2_SE5[1]	FTM1_CH6	CAN1_RX	LPI2C1_ SDAS[1]	-	FTM2_QD_ PHB[2]	-
57	PTA11	DISABLED	-	FTM1_CH5	LPUART0_ RX[1]	FXIO_D1	CMP0_ RRT[2]	-	-
58	PTA10	JTAG_TDO/noetm_ TRACE_SWO	-	FTM1_CH4	LPUART0_ TX[1]	FXIO_D0	-	-	JTAG_TDO/noetm_ TRACE_SWO
59	PTE1	ADC2_SE6[1] DISABLED[2]	ADC2_SE6[1]	LPSPI0_ SIN	LPI2C0_ HREQ	LPI2C1_ SCL[1]	LPSPI1_ PCS0[2]	FTM1_FLT1	-

Table 25. LQFP64 pinout comparison...*continued*

64 LQFP	Pin name	Default	ALT0	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
60	PTE0	ADC2_SE7[1] DISABLED[2]	ADC2_SE7[1]	LPSPI0_ SCK	TCLK1	LPI2C1_ SDA[1] _	LPSPI1_ SOUT[2]_	FTM1_FLT2	-
61	PTC5	JTAG_TDI	-	FTM2_CH0	RTC_ CLKOUT	LPI2C1_ HREQ[1]_	-	FTM2_QD_ PHB	JTAG_TDI
62	PTC4	JTAG_TCLK/SWD_CLK	ACMP0_IN2[1] CMP0_IN2[2]	FTM1_CH0	RTC_ CLKOUT	-	EWM_IN	FTM1_QD_ PHB	JTAG_TCLK/SWD_ CLK
63	PTA5	RESET_b	-	-	TCLK1	-	-	JTAG_TRST_ b[1]	RESET_b
64	PTA4	JTAG_TMS/SWD_DIO	-	-	-	ACMP0_ OUT[1] CMP0_ OUT[2]_	EWM_OUT_ b	-	JTAG_TMS/SWD_ DIO

[1] differs on KE1xF
[2] differs on MCX E24x

11 Software and tools

Software and tools are key enablers for both the KE1xF and MCX E24x families. A major objective is to maintain as much compatibility as possible to facilitate software reuse for customers. However, a few important differences are highlighted in Table 26.

The IDE and compiler support (including IAR, MCUXpresso, Keil, and GCC) is either already available or planned for both families.

The KE1xF and MCX E24x families are supported by the MCUXpresso SDK, which provides a consistent set of peripheral drivers, middleware components, and example projects. This shared software infrastructure greatly simplifies migration, as the majority of low-level driver interfaces and initialization flows is aligned. Developers already familiar with the KE1xF software stack will find the transition to MCX E24x straightforward, requiring minimal changes to existing application code.

11.1 Software products available or planned from NXP

Table 26. Software products available for the devices

Feature	KE1xF	MCX E24x
MCUXpresso SDK support	✓ Yes	✓ Yes
Math and motor control library	✓ Yes	✓ Yes
Security	✗ No (only basic CRC, ..)	EdgeLock Accelerator (CSEC)
Safety	Basic (ECC flash, watchdog)	Advanced (ECC SRAM + flash, error injection module, error report module)
Operating system	FreeRTOS	FreeRTOS and Zephyr
Bootloader	Classic bootloader	Secure bootloader
LIN stack	✓ Yes	✓ Yes
TCP/IP stack	✗ No	✓ Yes
FreeMASTER	✓ Yes	✓ Yes

11.2 Debugging capabilities

Debugging capabilities are a critical aspect of embedded system development, as they directly impact development efficiency and troubleshooting effectiveness. The KE1xF and MCX E24x families provide standard debugging interfaces such as JTAG and SWD, but the MCX E24x family introduces some enhancements.

These include advanced trace features, such as ETM and MTB. This section highlights the key differences to help developers take full advantage of the MCX E24x's more capable debug infrastructure.

Table 27. Debug feature comparison

Feature	KE1xF	MCX E24x
Debug interface	JTAG/SWD	JTAG/SWD
Embedded Trace Macrocell (ETM)	✗ No	✓ Yes
Data Watchpoint and Trace (DWT)	✓ Yes	✓ Yes
Instruction Trace Macrocell (ITM)	✓ Yes	✓ Yes
Trace Port Interface Unit (TPIU)	✓ Yes	✓ Yes

12 Revision history

Table 28. Revision history

Document ID	Release date	Description
AN14806 v.1.0	22 September 2025	• Initial version

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.

Contents