AN14798

System Oscillator Usage for MCX A series

Rev. 2.0 — 24 September 2025 Application note

Document information

Information	Content
Keywords	AN14798, MCX A series, System Oscillator, SOSC, crystal
Abstract	This application note provides System Oscillator usage guide for MCX A series, describes how to choose the suitable crystal oscillator, how to design the PCB, meanwhile, some tests and code examples are provided for reference.

1 Introduction

The 50 MHz crystal oscillator module (named as System Oscillator, SOSC) in MCX A series is designed to handle off-chip crystals that have a frequency of 8 MHz~50 MHz. The output of the crystal oscillator can be used as a clock source for the System PLL (SPLL) clock (if available) and can also be selected as the clock source for system clocks.

This document contains an overview of the crystal model and operating principles. It also provides instructions on how to choose the suitable crystal and external components, including some PCB design guidelines. Finally, the document provides the SOSC configuration steps and test code. It is recommended that you follow the crystal specification and crystal PCB layout guidelines in this document to ensure the accuracy and stability of the system.

2 Crystal oscillator basics

A crystal oscillator is an electronic device that uses the piezoelectric effect of a quartz crystal to generate a high-precision oscillation frequency. The crystal oscillator is widely used in embedded systems because of its stability and ease of use.

NXP MCX A series MCUs are equipped with a 50 MHz crystal oscillator module (SOSC), which supports 8 MHz~50 MHz crystal and uses a Pierce oscillator structure.

2.1 Crystal model

The equivalent electrical model of the crystal can be simplified as follows:

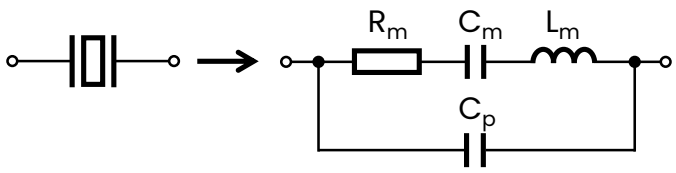

Figure 1. Crystal model

The crystal model is based on the following components:

- R_m: Motional Resistance (approximately equal to Equivalent Series Resistance known as ESR)
- C_m: Motional Capacitance
- L_m: Motional Inductance
- C_p: Shunt Capacitance

As shown in Figure 1, in the series branch, when the inductance (L_m) and capacitance (C_m) reach a balance, the L_m , and C_m can be seen as offset each other. The frequency at this point is called the series resonant frequency:

$$f_s = \frac{1}{2\pi\sqrt{L_m C_m}} \quad (1)$$

The shunt capacitance C_p can also achieve a balance with the series branch. Under this condition, the series branch behaves as inductance. The frequency at this point is called the parallel resonant frequency:

$$f_p = \frac{1}{2\pi\sqrt{L_m \frac{C_m C_p}{C_m + C_p}}} = f_s\sqrt{1+\frac{C_m}{C_p}} \quad (2)$$

For fundamental-mode crystal oscillator, it works between the f_s and f_p, also known as the parallel-resonant area, as shown in Figure 2, in this area, the reactance of the crystal is inductive.

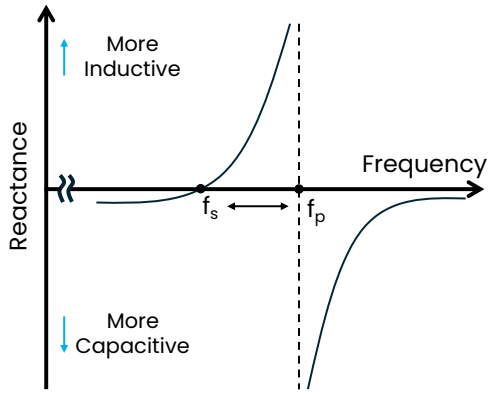

Figure 2. Reactance-frequency curve of crystal oscillator

When considering the load capacitance (C_L), which is given in the manufacturer's data sheet, the model can be updated as shown in Figure 3, and the frequency of the system is:

$$f_r = \frac{1}{2\pi\sqrt{L_m \frac{C_m(C_p+C_L)}{C_m+C_p+C_L}}} = f_s\sqrt{1+\frac{C_m}{C_p+C_L}} \quad (3)$$

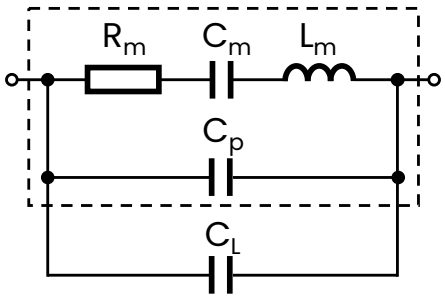

Figure 3. Crystal add load capacitance

By adjusting the load capacitance, the frequency of the system can be tuned.

Note: *Due to the high Q value of the crystal, the area between f_s and f_p is small. For the crystal oscillator to operate at the nominal frequency, it is necessary to ensure that the actual load capacitance value is around the data given in the data sheet. For load capacitance calculation, refer to Section 3.4*

2.2 Pierce crystal oscillator

Pierce crystal oscillator is a widely used parallel resonant topology, which is also used in the NXP MCX A series. The structure of the Pierce oscillator is shown in Figure 4 that consists of two parts:

- The active part, which is inside the MCU, mainly the inverter amplifier (g_m) and the feedback resistor (R_F).
- The passive part, which is connected externally, consists of a crystal oscillator (Y_1) and two load capacitors (C_{L1} and C_{L2}).

And EXTAL represents the crystal input pin on the MCU. XTAL represents the crystal output pin on the MCU.

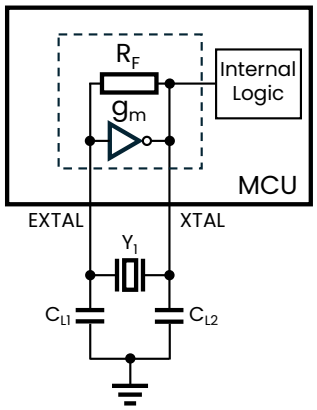

Figure 4. Pierce Crystal Oscillator

To make the crystal oscillate, follow the Barkhausen stability criterion:

- Amplifier Gain ≥ 1

$$|\beta A_v| \geq 1 \tag{4}$$

- Total phase shift across crystal = n * 360°

$$\angle \beta A_v = 0°, \ \pm 360° \ ... \tag{5}$$

Note: *This is a necessary condition for crystal oscillate.*

For the first condition, the inverter amplifier provides the gain ≥ 1.

For the second condition, the crystal and the load capacitors provide a 180° phase shift, and the inverter amplifier provides the other 180° phase shift, totally, the phase shift is 360°.

For a suitable oscillator circuit, through repeated amplification, the environmental noise can be amplified into stable oscillation signals.

3 Crystal selection

This section provides details on how to choose a suitable crystal oscillator based on key parameters.

The most important parameters that must be considered are:

- Oscillation mode
- Frequency
- Frequency tolerance
- Load capacitance
- Equivalent Series Resistance (ESR)

3.1 Oscillation mode

For passive crystal oscillator, there are two oscillation modes: fundamental frequency oscillation mode and overtone oscillation mode.

It is recommended to operate the crystal oscillator in the fundamental frequency mode because the crystal oscillator in overtone oscillation mode requires an additional frequency selection circuit.

3.2 Frequency

Frequency is the main parameter for selecting the crystal. For MCX A series MCU, 8 MHz~50 MHz are supported; choose the crystal within this frequency range and determine the final frequency based on the application.

3.3 Frequency tolerance

Usually, parts per million (ppm) is used as the unit of frequency tolerance, and the ppm can be calculated as:

$$ppm = \frac{f_r - f_0}{f_0} * 10^6 \quad (6)$$

Where:

- f_r: real frequency measured
- f_0: frequency given by the crystal manufacturer

For frequency tolerance, it is the frequency accuracy under nominal conditions.

Take the 10 MHz crystal as an example, if the frequency tolerance is ±20 ppm, the frequency of it is 10 MHz ± 200 Hz.

At the same time, the frequency stability over temperature characteristics is provided in the data sheet, pmm is also used as the unit. It represents the frequency change of the crystal within the operating temperature range.

3.4 Load capacitance

Load capacitance is an external parameter of the crystal used to tune the operating frequency of the crystal. The load capacitance provided by the crystal manufacturer is to make the crystal to operate at the nominal frequency.

To meet the load resistance requirements, ensure that the total capacitance in the crystal circuit matches the target load capacitance.

Figure 5 shows the commonly used crystal circuit, C_{L1} and C_{L2} are equal value capacitors.

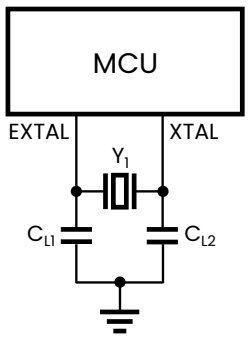

Figure 5. Crystal circuit

C_{L1} and C_{L2} are not the load capacitance provided by the manufacturer without considering parasitic capacitance. Assuming $C_{L1} = C_{L2}$, the below formula can be used to calculate the approximate load capacitance:

$$C_L = \frac{C_{L1} * C_{L2}}{C_{L1} + C_{L2}} = \frac{C_{L1}}{2} = \frac{C_{L2}}{2} \quad (7)$$

However, it is under ideal conditions. When all parasitic capacitances in the circuit are considered, the model can be simplified to Figure 6. All the capacitances in the box constitute the actual load capacitance. The design goal is to make the load capacitance the same as the data provided by the manufacturer.

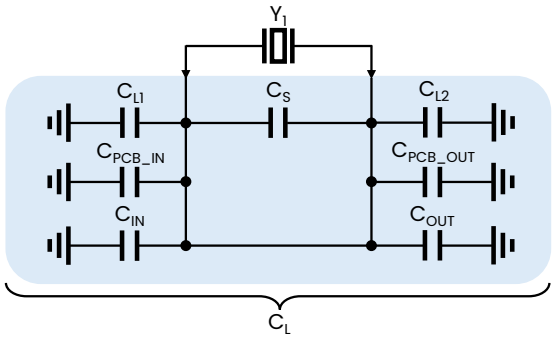

Figure 6. Crystal Model

Where:

- C_L: nominal load capacitance given by the crystal manufacturer
- C_{L1} and C_{L2}: external load capacitors
- C_{PCB_IN} and C_{PCB_OUT}: parasitic capacitance of the input and output PCB traces
- C_S: stray capacitance of the PCB
- C_{IN} and C_{OUT}: parasitic capacitance of the MCU EXTAL and XTAL pins

Then, the total capacitance of the crystal input to the ground is

$$C_{L1'} = C_{L1} + C_{PCB\_IN} + C_{IN} \quad (8)$$

The total capacitance of the crystal output to the ground is

$$C_{L2'} = C_{L2} + C_{PCB\_OUT} + C_{OUT} \quad (9)$$

Therefore, the load capacitance can be expressed as

$$C_L = \frac{(C_{L1'})*(C_{L2'})}{(C_{L1'})+(C_{L2'})} + C_S \quad (10)$$

Assuming that the input and output parameters are symmetric:

$$C_L = \frac{C_{L1'}}{2} + C_S = \frac{C_{L2'}}{2} + C_S \quad (11)$$

C_{L1} and C_{L2} can be calculated as:

$$C_{L1} = C_{L2} = 2*(C_L - C_S) - C_{IN} - C_{PCB\_IN} \quad (12)$$

For example,

- C_L = 10 pF (provided by crystal data sheet)
- C_S is around 1 pF (estimated value, measured on board)
- C_{IN} and C_{OUT} are around 2 pF (provided by MCU data sheet)
- C_{PCB_IN} and C_{PCB_OUT} are around 1 pF (estimated value, measured on board)

Note: *Good PCB layout is important, otherwise the capacitance value will be larger.*

In this case:

$$C_{L1} = C_{L2} = 2*(10-1) - 2 - 1 = 15\text{pF} \quad (13)$$

When choosing the capacitors, find a suitable value around the calculated value.

For details on how to measure these parameters on the PCB, refer to *Crystal Oscillator Design Guide* (document AN14518).

3.5 ESR and negative resistance

The ESR (R_m) and Negative resistance (R_N) are important parameters that must be considered in crystal selection that affects the reliable start of the oscillator and the oscillation stability.

Figure 7 shows the complex plane representation of the three-point oscillator in the Pierce Crystal Oscillator, on the left side of the vertical line in the figure is the oscillation area.

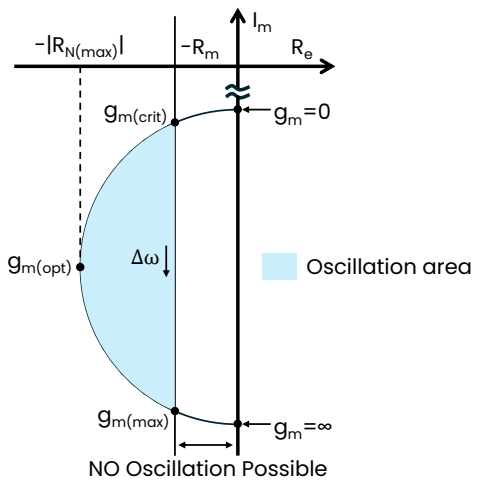

Figure 7. Complex plane representation of the three-point oscillator

The theoretical condition for a crystal oscillator is:

$$|R_N| = R_m \quad (14)$$

To ensure the stability of the system, there must be enough margin here, greater than 3~5 times. If a margin greater than 5 times is required, then

$$|R_N| > 5*R_m \quad (15)$$

The attachment provides the simulation results of the maximum ESR follow formula (15).

Note: *These simulation results can guide users to select the appropriate ESR, but there may be some differences between the simulation results and the actual situation.*

To measure the approximate negative resistance on the board, follow the below steps:

- Rework the board and connect a variable resistance R_T to the oscillation circuit, as shown in Figure 8.
- Set R_T to 0 Ω, observing the oscillation waveform of the crystal oscillator, see Section 5 for details.

- Gradually increase the value of R_T until the crystal oscillation stops.
- Gradually reduce the value of R_T until the crystal oscillator starts oscillating again after the board is repowered, then record the resistance value $R_{T(max)}$ at this time.
- Ensure

$$\frac{R_{T(max)}+ESR}{ESR} > 5 \quad (16)$$

Note: *The parasitic parameters of R_T affect the oscillation circuit. Use an SMD resistor for verification to minimize the impact.*

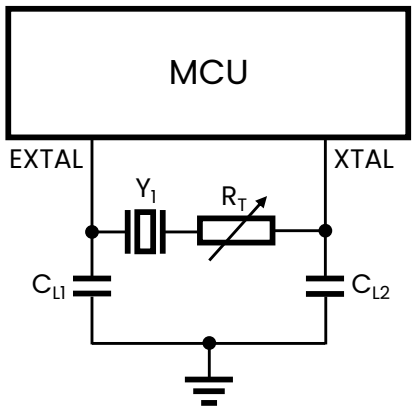

Figure 8. Negative resistance test

If the test above indicates that the margin is insufficient, the following operations may help improve it:

- Choose a crystal oscillator with lower ESR
- Change the value of C_{L1} and C_{L2} . Refer to the attached table for the impact of C_L on the trend of ESR changes.
  Note: *Change C_{L1} and C_{L2} can also affect the accuracy of the frequency.*
- Optimize the PCB layout to reduce the C_S. Lower C_S leads to higher negative resistance.

3.6 Other parameters

There are other parameters that must be considered. There are different requirements for these parameters in different applications.

- Startup time: it defines the time between the crystal being turned on and reaching a certain oscillation amplitude.
- Drive level: it defines the power dissipation of the crystal.
- Footprint: it affects the crystal size and price.

3.7 Selection example

Here is an example for crystal selection:

1. Select a crystal of the desired frequency and footprint, make sure that the frequency is 8 MHz ~ 50 MHz and works in fundamental frequency oscillation mode.
2. Download the data sheet of selected crystal and evaluate relevant parameters. Below is a table from a crystal data sheet:

Table 1. Crystal specifications

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions/Remarks
Nominal frequency	f_nom	-	16.000000	-	MHz	Fundamental
Frequency tolerance	f_tol	-10	-	+10	$X10^{-6}$	@+25°C
Frequency Stability over temperature	f_tem	-12	-	+12	$X10^{-6}$	-20°C to +75°C
Operating temperature	T_use	-20	-	+75	°C	
Level of drive	DL	10	-	100	µW	
Load capacitance	CL	-	9	-	pF	
Motional resistance(ESR)	R1	-	-	80	Ω	
Motional capacitance	C1	-	1.88	-	fF	
Motional inductance	L1	-	52.65	-	mH	
Shunt capacitance	C0	-	0.84	-	pF	
Frequency aging	f_age	-1	-	+1	$x10^{-6}$ /year	@+25°C, First year

As shown above, it contains many of the parameters mentioned above, the frequency of the crystal is 16 MHz (Fundamental), which meets the requirements mentioned in step 1.

3. Frequency tolerance, frequency stability over temperature, operating temperature depend on the needs of the current application and can be checked according to the actual conditions.
4. Besides frequency, a critical parameter in the selection is ESR, the maximum value here is 80 Ω (shown in Table 1), and the attachment can be used to determine whether the selected crystal is suitable.

- Frequency, load capacitance, and shunt capacitance are the main parameters that determine maximum ESR.
- In Table 1, Frequency = 16 MHz, load capacitance = 9 pF, shunt capacitance = 0.84 pF.
- Open the attachment, select sheet *Freq_16MHz*. Find the row where C_p = 1 pF and C_L = 9 pF, shown as below:

Table 2. ESR selection table

Frequency	Cp	SOSCCFG[RANGE]	CL	Max ESR
16 MHz	1 pF	00b (8-16 MHz)	9 pF	62 Ω
16 MHz	1 pF	01b (16-25 MHz)	9 pF	164 Ω

When SOSCCFG[RANGE] = 01b, the maximum ESR is 164 Ω, and the maximum ESR of the selected crystal is 80 Ω, which meets the requirements.

Note: *The maximum ESR in the table is given according to the principle of formula (15), which ensures the stability of the system. When the ESR of the selected crystal is greater than the value given in the table, the crystal usually works normally, but it is still recommended to leave enough margin based on the application.*

4 PCB design guidelines

Through the previous theoretical analysis of the crystal oscillator, it is shown that a good PCB design is important because the PCB design affects the frequency and stability of the crystal oscillator.

It is recommended to follow the following specifications for PCB design:

- Connect the crystal and external load capacitors on the PCB as close as possible to the oscillator input and output pins of the chip.
- The length of traces in the oscillation circuit should be as short as possible and must not cross other signal lines.
- Ensure that the load capacitors have a common ground plane.
- Loops must be made as small as possible to minimize the noise coupled in through the PCB and to keep parasitic effects as small as possible.
- Lay out the ground (GND) pattern around the traces and under the crystal unit.
- Do not lay out other signal lines under crystal unit for a multilayered PCB.

5 SOSC configuration and test

The module used for external crystal in MCX A series is called System Oscillator (SOSC). This section introduces the steps for SOSC configuration and provides sample code for startup time test and clock output test.

Note: *The module also supports an external reference clock as the selected clock by setting the register SOSCCFG[EREFS] (0b for external reference clock, 1b for crystal oscillator).*

5.1 SOSC configuration steps

Below is the SOSC configuration example:

1. Write 1 to LDOCSR[LDOEN] to enable LDO
2. Write 1 to SOSCCFG[EREFS] to select SOSC source (crystal oscillator)
3. Write 0 to SOSCCFG[RANGE] to configure the SOSC range (based on the selected crystal oscillator)
4. Write 0 to SOSCCSR[LK] to unlock SOSCCSR
5. Write 1 to SOSCCSR[SOSCCM] to enable SOSC clock monitor
6. Write 1 to SOSCCSR[SOSCSTEN] to enable SOSC in Deep Sleep mode if needed
7. Write 1 to SOSCCSR[SOSCEN] to enable SOSC
8. Read SOSCCSR[SOSCVLD] until it returns 1, indicating SOSC is valid
9. Read SOSCCSR[SOSCERR] to ensure that it returns 0
10. Write 1 to RCCR[SCS] to switch the main clock to SOSC if needed
11. Read CSR[SCS] until it returns 1, indicating the switch is complete

5.2 Startup time test and code

As shown in Figure 9, a scope is used to measure the startup time.

Note: *For startup time test, an active low capacitance single-ended probe (max. 1pF) is recommended to be used for measurement because the capacitance of the probe can affect the measurement results. When measuring, it is best to measure directly close to the test point and not to use long wires to minimize the measurement uncertainties as much as possible.*

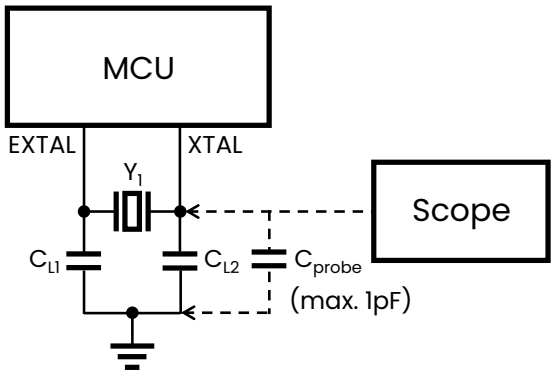

Figure 9. Startup time test

At the same time, the startup process can also be marked with a GPIO, and timed by a timer. If only the startup time is to be measured, this is a simpler approach and does not affect the circuit.

Below is the example code for startup time test on FRDM-MCXA346, P1_1 is used as the test GPIO.

```
int main(void)
{
    char ch;

    /* Init board hardware. */
    BOARD_InitHardware();

    if (SysTick_Config(SystemCoreClock / 1000000U)) // 1us interrupt set
    {
        while (1)
        {
        }
    }

    PRINTF("hello world.\r\n");

    ch = GETCHAR(); // getchar before crystal oscillator start
    app_SoscSet();
    PRINTF("SOSC WORK\r\n");
    PRINTF("Startup time: %f ms\r\n", g_count/1000.0);

    while (1)
    {
    }
}

/**
* @brief app_SoscSet  SOSC set
* @param NULL
* @return NULL
*/
void app_SoscSet(void)
{
  /* De-initializes the SCG SOSC */
  SCG0->SOSCCSR = SCG_SOSCCSR_SOSCERR_MASK;
  /* Enable LDO */
  SCG0->LDOCSR |= SCG_LDOCSR_LDOEN_MASK;
  /* Select SOSC source (crystal oscillator) and Configure SOSC range */
  SCG0->SOSCCFG = SCG_SOSCCFG_EREFS_MASK | SCG_SOSCCFG_RANGE(0);
  /* Unlock SOSCCSR */
```

```
  SCG0->SOSCCSR &= ~SCG_SOSCCSR_LK_MASK;
  /* Enable SOSC clock monitor and Enable SOSC */
  SCG0->SOSCCSR |= (SCG_SOSCCSR_SOSCCM_MASK | SCG_SOSCCSR_SOSCEN_MASK);

  GPIO_PortSet(GPIO1, 1u << 1U);  // SOSC enable signal
  g_start_flag = 1;  // Start timing
  /* Wait for SOSC clock to be valid. */
  while ((SCG0->SOSCCSR & SCG_SOSCCSR_SOSCVLD_MASK) == 0U)
  {
  }
  GPIO_PortClear(GPIO1, 1u << 1U);  // SOSC valid signal
  g_stop_flag = 1;  // Stop timing
}

/**
 * @brief  SysTick_Handler   System tick interrupt
 * @param  NULL
 * @return NULL
*/
void SysTick_Handler(void)
{
    if(g_stop_flag == 1)  // Stop timing
    {
      g_stop_flag = 0;
      g_start_flag = 0;
    }

    if (g_start_flag == 1) // Start timing
    {
       g_count ++;  // Counting result
    }
}
```

Figure 10 shows the startup time test results.

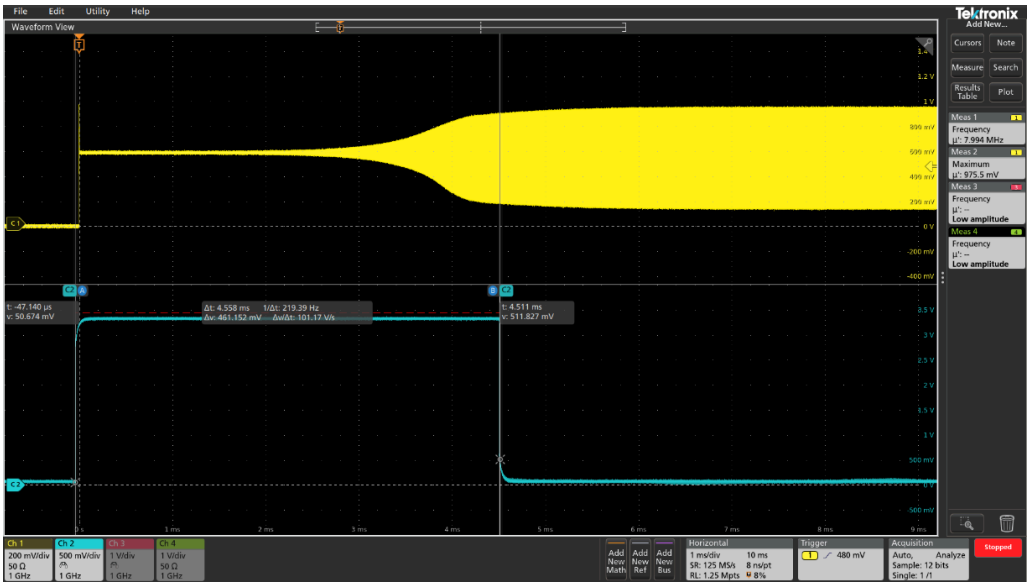

Figure 10. Startup time test results

Where:

- CH1 is the waveform from MCU XTAL pin

- CH2 is the waveform from P1_1 (test GPIO)

P1_1 is low before SOSC is enabled. It jumps from low to high when SOSC is enabled and jumps from high to low when the clock signal is valid. The startup time shown in this figure is 4.558 ms.

The waveform at the moment of crystal oscillator startup is shown in Figure 11, after enabling the register (SOSCCSR[SOSCEN]) and a short delay, the output voltage of XTAL pin changes indicating that the crystal oscillator entered the startup state.

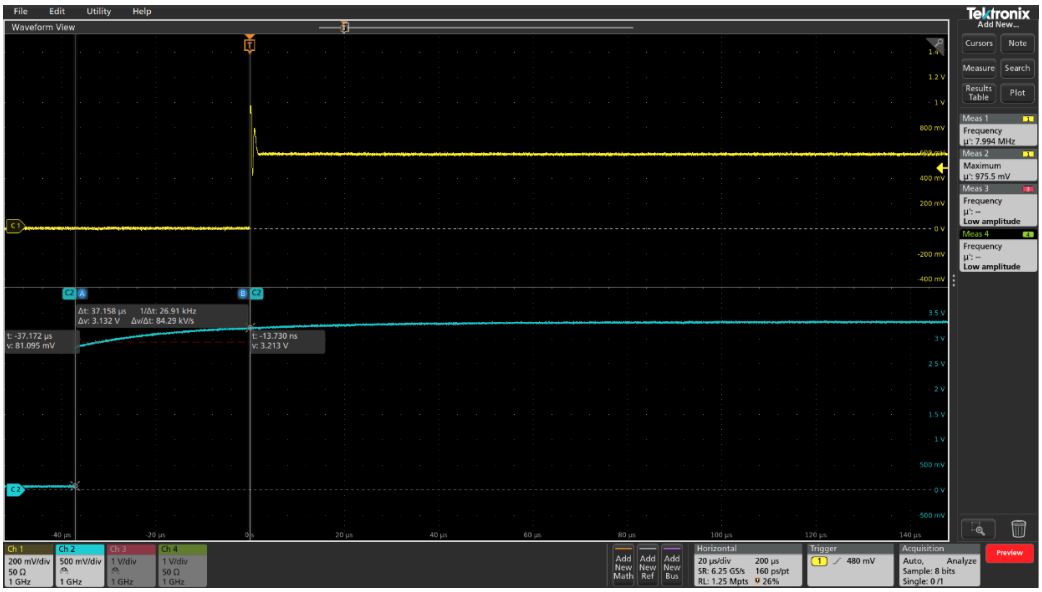

Figure 11. The moment of startup

After startup, the crystal oscillator enters a stable oscillation state as shown in Figure 12.

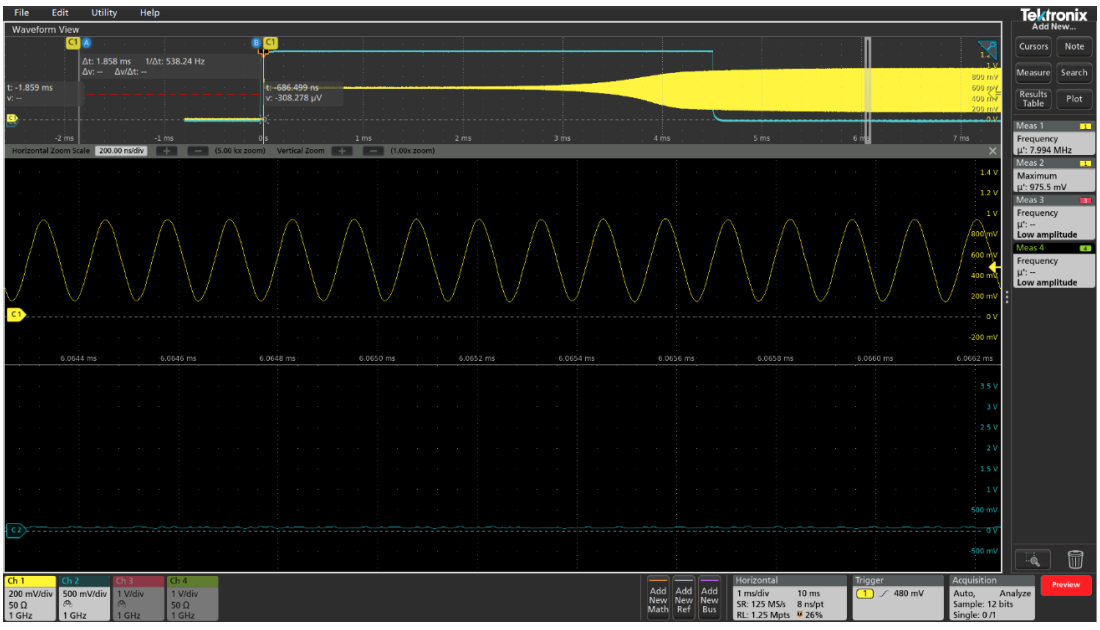

Figure 12. Stable oscillation state

The startup time can also be printed by the UART, as shown in Figure 13, the startup time recorded by timer is 4.567 ms. It is consistent with the test waveform.

Note: *Set PRINTF_FLOAT_ENABLE and PRINTF_ADVANCED_ENABLE to 1 to ensure that the printing of float variables works properly.*

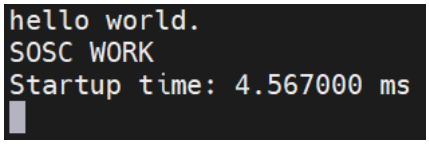

Figure 13. UART printed result with probe

5.3 Clock output test and code

The MCX A series MCU provides a clock output function, a useful function for clock observation. Not to affect the crystal oscillator circuit, observe the oscillation frequency through the clock output function instead of measuring it directly on the crystal oscillator pin. Below are the steps to enable the crystal oscillator clock output:

1. Configure the IO as a CLKOUT function (there are several IOs that can be configured as CLKOUT, refer to the reference manual for details, here take P4_2 of MCXA346 as an example).
2. Write 0 to CLKUNLOCK[UNLOCK] to unlock the clock configuration registers.
3. Write 2 to MRCC_CLKOUT_CLKSEL[MUX] to select CLK_IN as the clock source.
4. Write 7 to MRCC_CLKOUT_CLKDIV[DIV] to set the divider value to 8 (for an 8 MHz crystal oscillator, a 1 MHz clock output can be obtained here).
5. Write 1 to CLKUNLOCK[UNLOCK] to lock the clock configuration registers.

The core code for clock output configuration is as follows:

```
/**
* @brief BOARD_InitCLKOUT_Pins   Initialization of the CLKOUT pin
* @param NULL
* @return NULL
*/
void BOARD_InitCLKOUT_Pins(void)
{
    /* PORT4: Peripheral clock is enabled */
    CLOCK_EnableClock(kCLOCK_GatePORT4);
    /* PORT4 peripheral is released from reset */
    RESET_ReleasePeripheralReset(kPORT4_RST_SHIFT_RSTn);

    const port_pin_config_t port4_2_config = {
          /* Internal pull-up resistor is enabled */
          .pullSelect = kPORT_PullUp,
          /* Low internal pull resistor value is selected. */
          .pullValueSelect = kPORT_LowPullResistor,
          /* Fast slew rate is configured */
          .slewRate = kPORT_FastSlewRate,
          /* Passive input filter is disabled */
          .passiveFilterEnable = kPORT_PassiveFilterDisable,
          /* Open drain output is disabled */
          .openDrainEnable = kPORT_OpenDrainDisable,
          /* Low drive strength is configured */
          .driveStrength = kPORT_LowDriveStrength,
          /* Normal drive strength is configured */
          .driveStrength1 = kPORT_NormalDriveStrength,
          /* Pin is configured as CLKOUT */
          .mux = kPORT_MuxAlt1,
          /* Digital input enabled */
          .inputBuffer = kPORT_InputBufferEnable,
```

```
            /* Digital input is not inverted */
            .invertInput = kPORT_InputNormal,
            /* Pin Control Register fields [15:0] are not locked */
            .lockRegister = kPORT_UnlockRegister};
    /* PORT4_2 is configured as CLKOUT */
    PORT_SetPinConfig(PORT4, 2U, &port4_2_config);
}

/**
* @brief app_ClockOut clock output set, to observe clock signal
* @param NULL
* @return NULL
*/
void app_ClockOut(void)
{
    /* Unlock the register */
    SYSCON -> CLKUNLOCK = SYSCON_CLKUNLOCK_UNLOCK(0U);
    /* Select CLK_IN as the clock out source */
    MRCC0 -> MRCC_CLKOUT_CLKSEL = MRCC_MRCC_CLKOUT_CLKSEL_MUX(2U);
    /* Set the clock divider, divider value = (DIV+1) */
    MRCC0 -> MRCC_CLKOUT_CLKDIV = MRCC_MRCC_CLKOUT_CLKDIV_DIV(7U);
    /* Lock the register */
    SYSCON -> CLKUNLOCK = SYSCON_CLKUNLOCK_UNLOCK(1U);
}
```

Figure 14 shows the clock out test result in this example, it is 1.000 MHz.

Note: *It is recommended to use a ground spring for measurement to minimize the loop area, otherwise high-frequency noise is coupled in.*

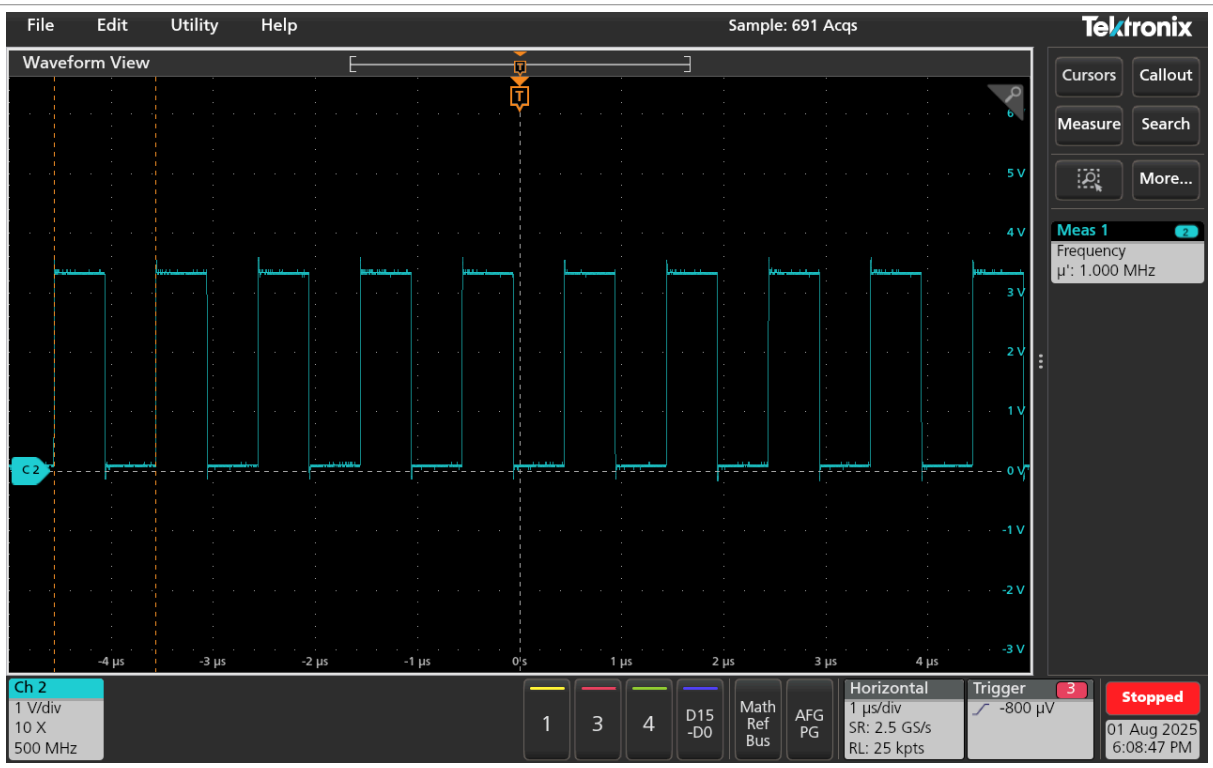

Figure 14. Clock output signal

6 Conclusion

This application note is intended to help users optimize their use of the SOSC module on the MCX A series. It includes an analysis of crystal oscillator principles, recommendations for selecting appropriate crystal oscillators, PCB design guidelines, and illustrative test examples.

7 References

1. *Using the 16 MHz Crystal Oscillator MC9328MX1, MC9328MXL, and MC9328MXS* (document AN2500)
2. *Crystal Oscillator Design Guide* (document AN14518)
3. *MCX A345/346 Reference Manual* (document MCXAP144M240F60RM)
4. *MCX A345/346 data sheet* (document MCXAP144M240F60)
5. *Design of Electrical Parameter Measurement System of Quartz Crystal Oscillator*, IEEE
6. *A Study of Low-Power Crystal Oscillator Design,* IEEE
7. *High-performance crystal oscillator circuits: theory and application, IEEE*

8 Revision history

Table 3. Revision history

Document ID	Release date	Description
AN14798 v.2.0	24 September 2025	Minor updates
AN14798 v.1.0	16 September 2025	Initial version

9 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2025 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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.

Microsoft, Azure, and ThreadX — are trademarks of the Microsoft group of companies.

Contents