N32H7xx series CORDIC user guide

Introduction

This document is designed for users to quickly understand the basic functions and usage of the CORDIC peripherals in N32H7xx series microcontrollers (MCUs), in order to guide users to easily use this product.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

Content

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

1 Overview

CORDIC performs iterative addition and shift operations in hardware, replacing complex software calculations using common mathematical functions. It can perform four iterations per clock cycle, significantly saving time while providing acceptable data accuracy.

This document introduces the basic usage of the CORDIC module in the N32H7xx series microcontrollers (MCUs). For detailed such as function descriptions, parameters, precautions, etc., please refer to the EN_UM_N32H7xx Series User Manual.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

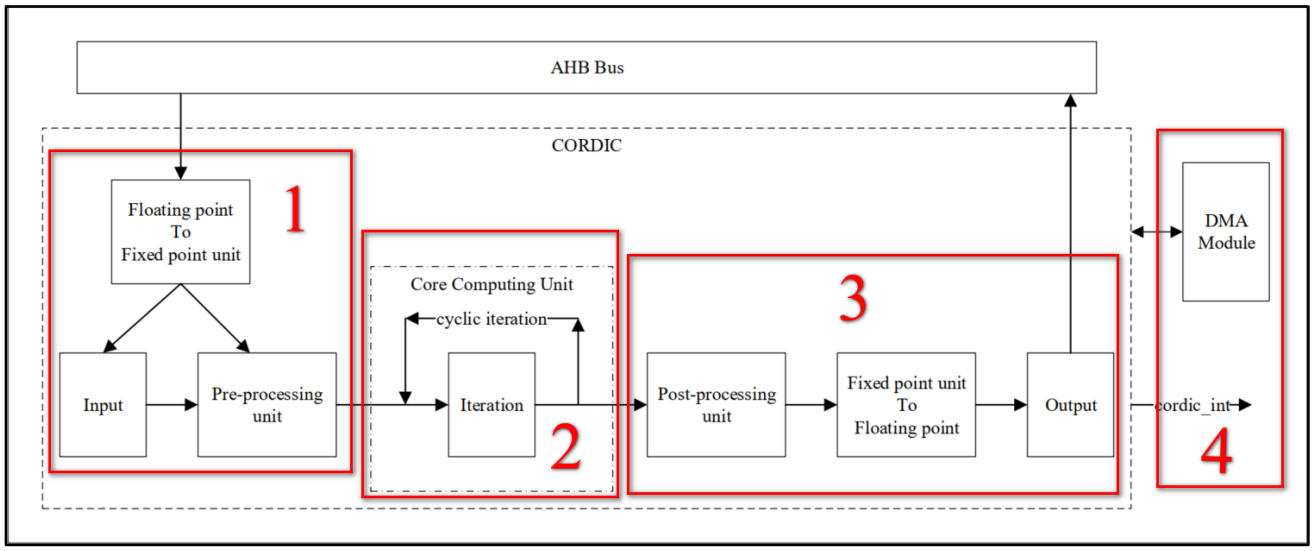
2 Introduction of CORDIC

2.1 Brief introduction

CORDIC supports the following 10 commonly used mathematical functions: sin, cosine, sinh, cosh, atan, atan2, atanh, modulus, square root, and natural logarithm.

After initialization, write the required parameters (format and data range) to the WDAT register, and CORDIC will automatically begin the conversion. Once the conversion is complete, retrieve the result directly from the RDAT register.

2.2 CORDIC Functional Block Diagram



The functions of each part are as follows:

1) Data Input: Includes an input data conversion unit and a preprocessing unit. The data conversion unit can convert input floating-point data (single-precision or half-precision) into fixed-point data (q1.31 or q1.15).
2) Calculation Core Unit: Iteratively calculates the input data to obtain the calculation result. The precision is determined by the configurable number of iterations.
3) Data Output: Includes a data post-processing unit and an output data conversion unit. The output data conversion unit can convert the output fixed-point data (q1.31 or q1.15) into floating-point data (single-precision or half-precision).
4) Interrupt and DMA: Interrupt request output and DMA read/write interface.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

3 CORDIC Configuration

3.1 CORDIC Configuration Instructions

CORDIC configuration is achieved through the operation control status register CORDIC_CTRLSTS. The main configuration items are as follows:

1) Input data format:
   - Supports both fixed-point and floating-point formats.
   - Input data width can be configured as 32-bit or 16-bit.
   - Number of times the data register (WDAT) needs to be written before each calculation can be configured as 1 or 2 times, for example:
     - Two write operation with 32-bit width: Requires a 32-bit main parameter and a 32-bit secondary parameter, written sequentially to the data register.
     - One write operation with 32-bit width: Only a 32-bit main parameter needs to be written; the secondary parameter retains its default value of 1 or the previous setting.
     - One write operation with 16-bit width: Requires a 16-bit main parameter and a 16-bit secondary parameter, combined into 32-bit data and written to the register at once, with the main parameter in the lower 16 bits.
     - Two write operation with 16-bit width: The first write is a 16-bit main parameter combined with a 16-bit secondary parameter, with the main parameter in the lower 16 bits. The second write is ignored.
2) Output Data Format:
   - Supports both fixed-point and floating-point formats.
   - Output data width can be configured as 32-bit or 16-bit.
   - Number of times the result register (RDAT) needs to be read after each calculation can be configured as 1 or 2 times, for example:
     - Two read operation with 32-bit width: The first read return a 32-bit main result, and the second read return a 32-bit secondary result.
     - One read operation with 32-bit width: Only read once, then return a 32-bit main result.
     - One read operation with 16-bit width: Only read once, then return a 32-bit result, with the main result in the lower 16 bits and the secondary result in the higher 16 bits.
     - Two read operation with 16-bit width: The first read return a 32-bits result, with the the 16-bits main result in the lower 16 bits only, and the higher 16 bits retain the previous value. Then the second read return another 32-bit result, with the main result in the lower 16 bits and the secondary result in the higher 16bits.
3) Mathematical functions:
   Choose one of 10 functions.
4) Precision:
   Number of iterations; 12, 16, 20, or 24 iterations are recommended.
5) Scaling Factor (SCALE):
   Some functions support scaling the result ($*2^{SCALE}$), but the input parameters need to be scaled down ($/2^{SCALE}$) before writing to the register.Different functions have different scaling factor requirements, and the actual input and output data ranges are also different. Please refer to the user manual for details.
6) DMA:
   DMA is supported for both reading and writing, and can be configured separately.
7) Supports two types of interrupts, sharing one interrupt entry:
   - Input parameter overflow.
   - Calculation result ready.
8) Limiting:
   Values exceeding the upper limit are replaced with the upper limit value; values below the lower limit are replaced with the lower limit value.
   - Coordinate output result limiting.
   - Phase output result limiting.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

3.2 CORDIC Initialization

The CORDIC initialization structure shows as follows:

```
typedef struct
{
    uint32_t  Function;      /* Select function */
    uint32_t  Precision;     /* Set the number of iterations */
    uint32_t  Scale;         /* Select Scaling Factor */
    uint32_t  NbWrite;       /* Select the number of CORDIC_WDAT register to write */
    uint32_t  NbRead;        /* Select the number of CORDIC_RDAT register to read */
    uint32_t  InSize;        /* Select input data width */
    uint32_t  OutSize;       /* Select output data width */
    uint32_t  InSelect;      /* Select floating-point input or fixed-point input */
    uint32_t  OutSelect;     /* Select floating-point output or fixed-point output */
    uint32_t  CodinLimit;    /* Enables or disables Coordinate Limit */
    uint32_t  PhaseLimit;    /* Enables or disables Phase Limit */
} CORDIC_InitType;
```

The introduction of the structure members are as follows:

- Function:
  Select the mathematical function to be calculated, one out of ten, e.g, CORDIC_FUNCTION_SINE.
- Precision:
  The precision of the operation, and the actual value written to the register is the number of clock cycles, which is the number of iterations divided by 4; e.g, CORDIC_PRECISION_6CYCLES, which means 24 iterations.
- Scale:
  The scaling factor, e.g, CORDIC_SCALE_2, which means a scaling factor of 2^2.
- NbWrite:
  The number of times the data register WDAT needs to be written before each calculation,
  e.g, CORDIC_NBWRITE_1, which means only write once.
- NbRead:
  The number of times the result register RDAT needs to be read after each conversion,
  e.g, CORDIC_NBREAD_1, which means only read once.
- InSize:
  Input data width, e.g, CORDIC_INSIZE_16BITS, which means 16-bits intput data width.
- OutSize:
  Output result width, e.g, CORDIC_OUTSIZE_32BITS, which means 32-bits output data width.
- InSelect:
  Input data format, e.g, CORDIC_INPUT_FIX, which means fix-point input data format.
- OutSelect:
  Output data format, e.g, CORDIC_OUTPUT_FLOAT, which means float-point output data format..
- CodinLimit:
  Coordinate output limiting, e.g, CORDIC_CODIN_LIMIT_ENABLE,
  which means the output coordinate result is limited to [-1, 1).
- PhaseLimit:
  Phase output limiting, e.g, CORDIC_PHASE_LIMIT_DISABLE,
  which means the phase output result is not limited, the result outside [-π,π) will not be correct.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

3.3 CORDIC Interrupts and DMA

The interrupt and DMA configuration of CORDIC are not included in the initialization, so additional configuration is required:

- Supports input parameter overflow interrupts and calculation result ready interrupts, sharing a single interrupt entry point, which can be enabled or disabled separately.
  The following funtion can be used:
  - void CORDIC_InterruptCmd(uint32_t Interrupt, FunctionalStatus Cmd)

- Both data input and result output support DMA, which can be configured separately.
  The following funtions can be used:
  - void CORDIC_DMAReadRequestCmd(FunctionalStatus Cmd)
  - void CORDIC_DMAWriteRequestCmd(FunctionalStatus Cmd)

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

4 CORDIC Operation Mode

CORDIC supports four operating modes as follows. The detailed descriptions of each mode can be found in the user manual:

- Zero-Overhead Mode:
  After writing parameters, the result is read directly. If the result is not ready, a bus wait state will be inserted until it is ready and the result is returned.
- DMA Mode:
  Input parameters and read results via DMA.
- Polling Mode:
  Polls the RRF flag in CORDIC_CTRLSTS. The result is read after the flag is set to 1.
- Interrupt Mode:
  Enables interrupts. When RRF is set, a calculation result ready interrupt is generated and the result is read.

Since CORDIC calculations only require approximately 3-6 clock cycles, the zero-overhead mode is the fastest:

- DMA Mode:
  After data is ready, additional overhead is required such as DMA request and data transfer.
- Polling Mode:
  Checking the RRF flag requires additional software operations.
- Interrupt Mode:
  Requires additional interrupt response and interrupt service.

In applications, users can select the appropriate operating mode according to their needs.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

5 History versions

Version	Date	Notes
V1.0.1	2025.12.5	Initial version

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg

6 Disclaimer

This document is the exclusive property of NSING TECHNOLOGIES PTE. LTD. (Hereinafter referred to as NSING). This document, and the product of NSING described herein (Hereinafter referred to as the Product) are owned by NSING under the laws and treaties of Republic of Singapore and other applicable jurisdictions worldwide. The intellectual properties of the product belong to NSING Technologies Inc. and NSING Technologies Inc. does not grant any third party any license under its patents, copyrights, trademarks, or other intellectual property rights. Names and brands of third party may be mentioned or referred thereto (if any) for identification purposes only. NSING reserves the right to make changes, corrections. enhancements, modifications, and improvements to this document at any time without notice. Please contact NSING and obtain the latest version of this document before placing orders. Although NSING has attempted to provide accurate and reliable information, NSING assumes no responsibility for the accuracy and reliability of this document. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. In no event shall NSING be liable for any direct, indirect, incidental, special, exemplary, or consequential damages arising in any way out of the use of this document or the Product. NSING Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, Insecure Usage'. Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, all types of safety devices, and other applications intended to supporter sustain life. All Insecure Usage shall be made at user's risk. User shall indemnify NSING and hold NSING harmless from and against all claims, costs, damages, and other liabilities, arising from or related to any customer's Insecure Usage Any express or implied warranty with regard to this document or the Product, including, but not limited to. The warranties of merchantability, fitness for a particular purpose and non-infringement are disclaimed to the fullest extent permitted by law. Unless otherwise explicitly permitted by NSING, anyone may not use, duplicate, modify, transcribe or otherwise distribute this document for any purposes, in whole or in part.

NSING Technologies Pte. Ltd.
Address: 20 Science Park, #03-15/16
Teleteck Park, East Wing, Singapore 117674
Email: sales@nsing.com.sg