# M4 DSP Matrix Functions

NuMicro® 32 位系列微控制器範例代碼介紹

## 文件資訊

| | |
|---|---|
| 代碼簡述 | 本範例代碼使用M4內核DSP計算矩陣運算 |
| BSP 版本 | M480 Series BSP CMSIS V3.04.000 |
| 開發平台 | NuMaker-PFM-M487 Ver 3.0 |

*The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.*

*Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.*

*All data and specifications are subject to change without notice.*

*For additional information or questions, please contact: Nuvoton Technology Corporation.*

www.nuvoton.com

# 1 功能介紹

## 1.1 簡介

CMSIS DSP函式庫有許多關於矩陣的運算包含：

1. 矩陣加法運算
2. 矩陣減法運算
3. 矩陣乘法運算
4. 逆矩陣
5. 矩陣縮放
6. 轉置矩陣

在此範例程式使用CMSIS DSP函式庫進行轉置矩陣、矩陣乘法和反矩陣運算，用戶可以直接使用這些函式，來實現自己的數學方程式運算。程式內比較了有無使用DSP計算時間的差異，並計算效率提升比率。

## 1.2 原理

使用者須先用arm_matrix_instance_f32宣告矩陣變數，再來呼叫函式arm_mat_init_f32()對使用到的矩陣初始化設定，定義矩陣直行與橫列個數以及矩陣數值。完成設定後即可對各矩陣進行運算，其程式設定如下表。

| arm_mat_init_f32(arm_matrix_instance_f32 *S, uint16_t nRows, uint16_t nColumns, float32_t *pData) | | |
|---|---|---|
| **參數：** | *S | [in, out] 指向所宣告的浮點數矩陣結構 |
| | nRows | [in] 矩陣直行個數 |
| | nColumns | [in] 矩陣橫列個數 |
| | *pData | [in] 矩陣數據 |
| **回傳值：** | | 無 |

| arm_mat_add_f32(const arm_matrix_instance_f32 *pSrcA, const arm_matrix_instance_f32 *pSrcB, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| **參數：** | *pSrcA | [in] 輸入被加數矩陣 |
| | *pSrcB | [in] 輸入加數矩陣 |
| | *pDst | [out] 輸出結果矩陣 |

| | |
|---|---|
| 回傳值： | 成功顯示ARM_MATH_SUCCESS, 矩陣不相合則顯示 ARM_MATH_SIZE_MISMATCH |

| arm_mat_sub_f32(const arm_matrix_instance_f32 *pSrcA, const arm_matrix_instance_f32 *pSrcB, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| 參數： | *pSrcA | [in] 輸入被減數矩陣 |
| | *pSrcB | [in] 輸入減數矩陣 |
| | *pDst | [out] 輸出結果矩陣 |
| 回傳值： | | 成功顯示ARM_MATH_SUCCESS, 矩陣不相合則顯示 ARM_MATH_SIZE_MISMATCH |

| arm_mat_mult_f32(const arm_matrix_instance_f32 *pSrcA, const arm_matrix_instance_f32 *pSrcB, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| 參數： | *pSrcA | [in] 輸入被乘數矩陣 |
| | *pSrcB | [in] 輸入乘數矩陣 |
| | *pDst | [out] 輸出結果矩陣 |
| 回傳值： | | 成功顯示ARM_MATH_SUCCESS, 矩陣不相合則顯示 ARM_MATH_SIZE_MISMATCH |

| arm_mat_inverse_f32(const arm_matrix_instance_f32 *pSrc, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| 參數： | *pSrc | [in] 輸入欲計算矩陣 |
| | *pDst | [out] 輸出逆矩陣 |
| 回傳值： | | 成功顯示ARM_MATH_SUCCESS, 矩陣不為方陣則顯示ARM_MATH_SIZE_MISMATCH, 若矩陣不可逆顯示ARM_MATH_SINGULAR |

| arm_mat_scale_f32(const arm_matrix_instance_f32 *pSrc, float32_t scale, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| 參數： | *pSrc | [in] 輸入欲計算矩陣 |
| | scale | [in] 縮放參數 |
| | *pDst | [out] 輸出結果矩陣 |
| 回傳值： | | 成功顯示ARM_MATH_SUCCESS |

| arm_mat_trans_f32(const arm_matrix_instance_f32 *pSrc, arm_matrix_instance_f32 *pDst) | | |
|---|---|---|
| 參數： | *pSrc | [in] 輸入欲計算矩陣 |
| | *pDst | [out] 輸出轉置後矩陣 |
| 回傳值： | | 成功顯示ARM_MATH_SUCCESS |

## 1.3 執行結果

執行後會打印出以下資訊

```
+----------------------------------------+
|     DSP Matrix functions Sample Code   |
+----------------------------------------+

Calculating time with DSP instruction is 0.040667 ms

Calculating time without DSP instruction is 0.243083 ms

Efficiency increase rate is 5.98
```

# 2 代碼介紹

使用CMSIS DSP函數庫進行轉置矩陣、矩陣乘法和反矩陣運算：

```
/* Initialize A Matrix Instance with numRows, numCols and data array(A_f32) */
arm_mat_init_f32(&A, numRows, numCols, A_f32);

/* Initialize Matrix Instance AT with numRows, numCols and data array(AT_f32) */
arm_mat_init_f32(&AT, numRows, numCols, AT_f32);

/* calculation of A transpose with DSP instruction */
arm_mat_trans_f32(&A, &AT);

/* Initialize ATMA Matrix Instance with numRows, numCols and data array(ATMA_f32) */
arm_mat_init_f32(&ATMA, numRows, numCols, ATMA_f32);

/* calculation of AT Multiply with A with DSP instruction */
arm_mat_mult_f32(&AT, &A, &ATMA);

/* Initialize ATMAI Matrix Instance with numRows, numCols and data array(ATMAI_f32) */
arm_mat_init_f32(&ATMAI, numRows, numCols, ATMAI_f32);

/* calculation of Inverse ((Transpose(A) * A) with DSP instruction */
arm_mat_inverse_f32(&A, &ATMAI);
```

接著使用CPU進行相同的計算：

```
/* Matrix Inverse (number of rows) */
void MatrixInverse(int n)
{
    uint32_t i,j,k;
    double t;
    for(i=0; i<n; ++i) {
        for(j=0; j<n; ++j) {
            A1_f32[i][n+j] = 0;
        }
    }
    for(i=0; i<n; ++i) {
        A1_f32[i][n+i] = 1;
    }
    for(i=0; i<n; ++i) {
```

```
        if(A1_f32[i][i] == 0)
            for(j=i+1; j<n; ++j)
                if(A1_f32[j][i] != 0) {
                    for(k=i; k<n*2; ++k)
                        swap(&A1_f32[i][k], &A1_f32[j][k]);
                    break;
                }
        /* if Matrix inverse not exist*/
        if(A1_f32[i][i] == 0) break;

        t = A1_f32[i][i];
        for(k=i; k<n*2; ++k) {
            A1_f32[i][k] /= t;
        }
        for(j=0; j<n; ++j) {
            if(i != j && A1_f32[j][i] != 0) {
                double t = A1_f32[j][i];
                for(k=i; k<n*2; ++k) {
                    A1_f32[j][k] -= A1_f32[i][k] * t;
                }
            }
        }
    }
}
/* Matrix Transpose (number of rows, number of columns) */
void MatrixTranspose(int m, int n)
{
    uint32_t i, j;
    for(i=0; i<n; i++) {
        for(j=0; j<m; j++) {
            resultT[i][j] = A1_f32[j][i];
        }
    }
}
/* Matrix Multiplication (number of rows, number of columns) */
void MatrixMult(int m, int n, int p)
{
    uint32_t i, j, k;
    for(i=0; i<m; i++) {
        for(j=0; j<p; j++) {
            resultM[i][j] = 0;
```

```
            for(k=0; k<n; k++) {
                resultM[i][j] = resultM[i][j] + resultT[i][k] * A1_f32[k][j];
            }
        }
    }
}
```

把計數器換成時間，其中計數器時鐘源為HXT 12MHz：

```
/* Calculate the time, timer clock source is 12M, unit is ms */
DSPCalTime = (DSPCalTime/12000000) * 1000;
CalTime = (CalTime/12000000)* 1000;
```

# 3 軟體與硬體環境

## ● 軟體環境

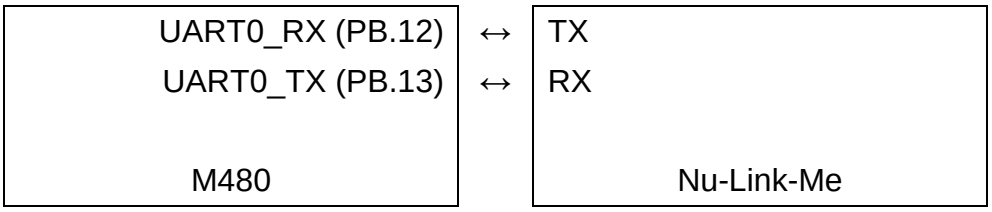
- ■ BSP 版本
  - ◆ M480 Series BSP CMSIS V3.04.000
- ■ IDE 版本
  - ◆ Keil uVersion 5.26

## ● 硬體環境

- ■ 電路元件
  - ◆ NuMaker-PFM-M487 or other M480 Development Board
- ■ 示意圖

M480 的 UART0_RX(PB.12)、UART0_TX(PB.13)連接至 Nu-Link Me，打印訊息。

設置終端機的 COM Port 與 Baud，COM Port 的編號可在裝置管理員中找到「NuBridge Virtual Com Port (COMX)」，Baud 設置為 115200。



# 4 目錄資訊

- EC_M480_DSP_Matrix_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

# 5 如何執行範例程式

1. 根據目錄資訊章節進入 ExampleCode 路徑中的 KEIL 資料夾，雙擊 DSP_Matrix.uvproj
2. 進入編譯模式介面
   a. 編譯
   b. 下載代碼至記憶體
   c. 進入 / 離開除錯模式
3. 進入除錯模式介面
   a. 執行代碼

# 6 修訂紀錄

| Date | Revision | Description |
|---|---|---|
| Jun. 25, 2019 | 1.00 | 1. 初始發布. |

## Important Notice

**Nuvoton 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, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.**

**All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.**

*Please note that all data and specifications are subject to change without notice.*
*All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.*