



	MGT-MDE-3-003		
	V1.02		
			2024-12-12

---

# 1

## 1.1

## 1.2

	Pl c( )
M:5000, M:5100	
M:Greator	M:5000 M
	M:Greator

# 2

## 2.1

- M:5x00 Pl c
- 1> C C ,
- 2> Pl c , X, Y, M S, SM T, C D, R S p

---

X , D

#

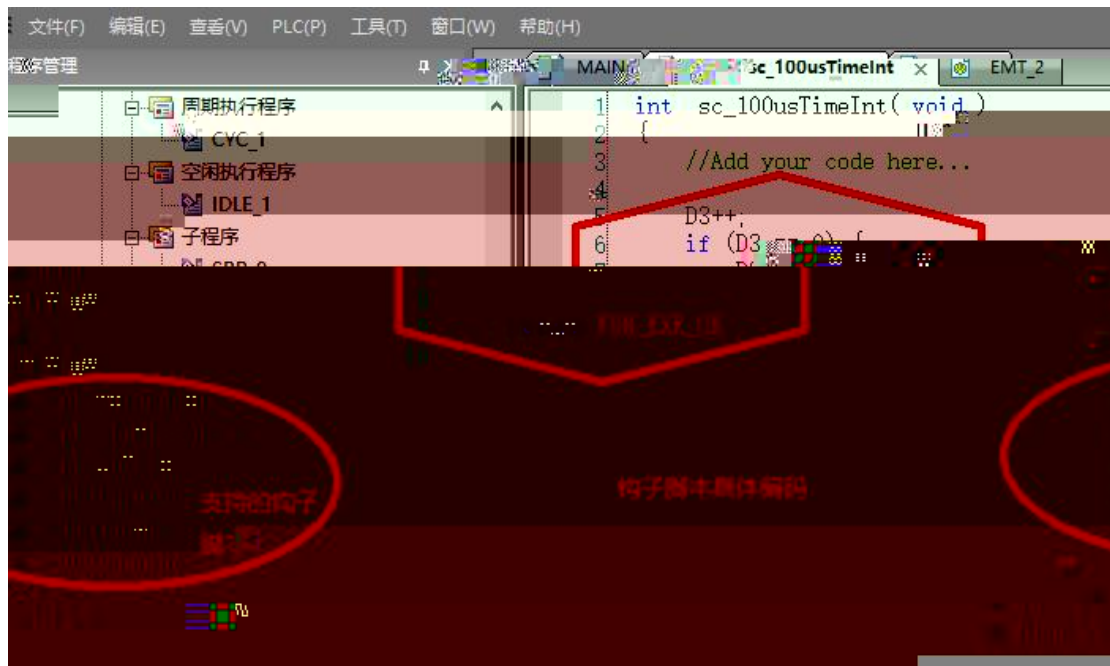


D

<http://www.megmeet.com/>

3

32



3

## 2.3

C

pl c

export_modul e. h	Pl c
user_common. c	
user_common. h	

# 3

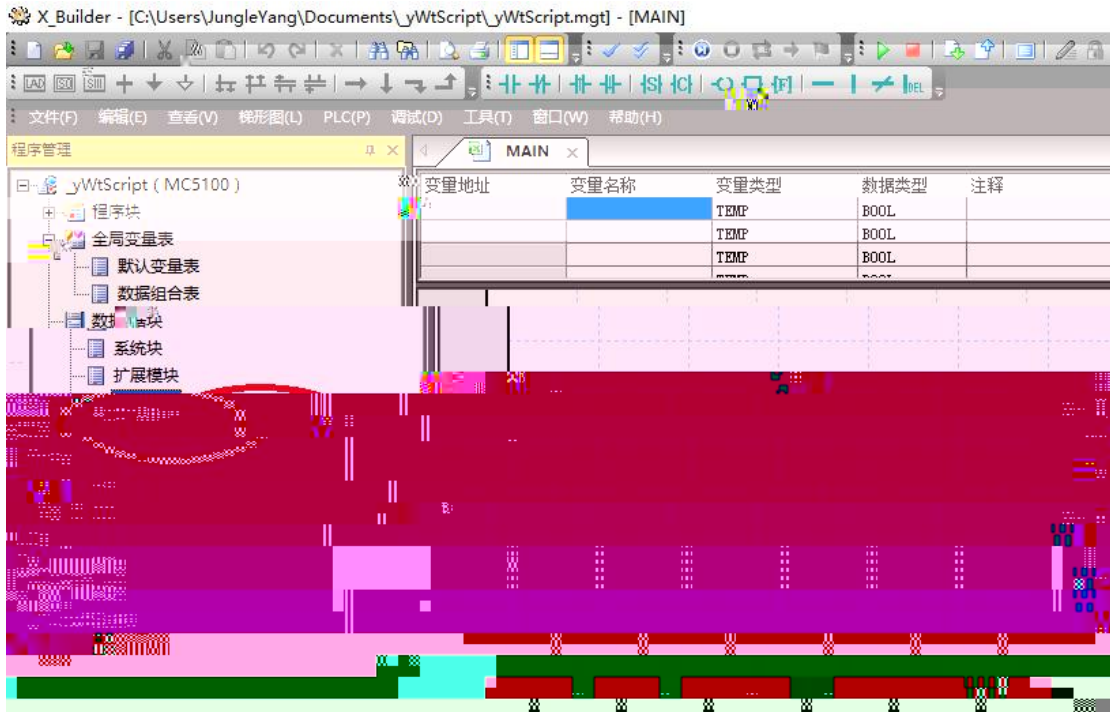
xBuild

C

D996( ) D1000 20000 D998( )

## 3.1

" " , " " " "



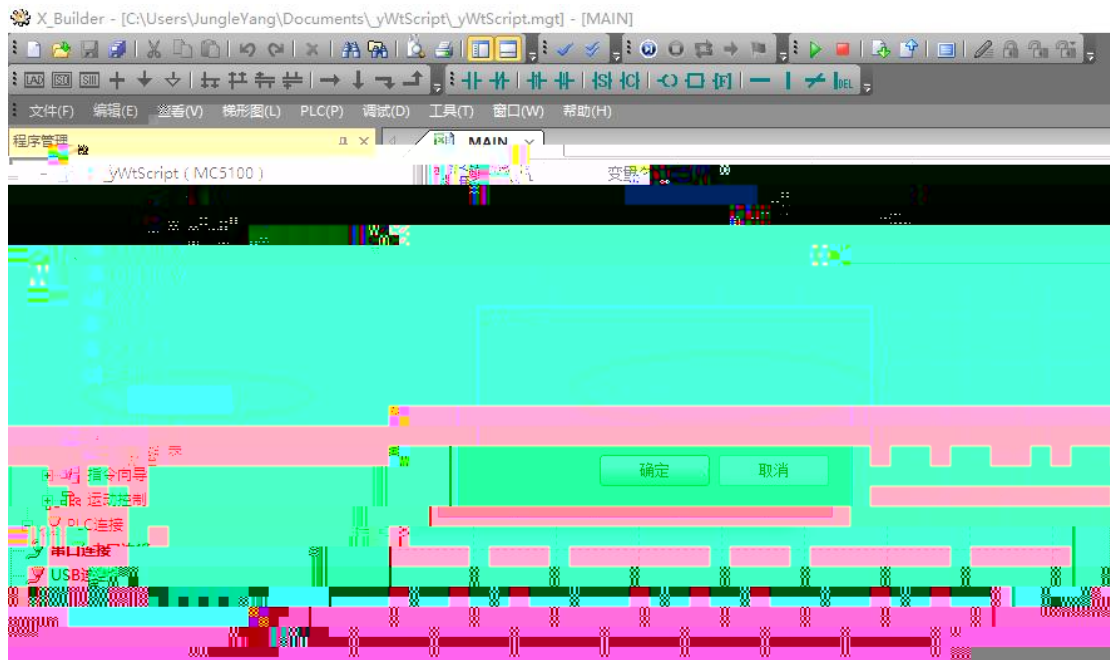
---

3. 6

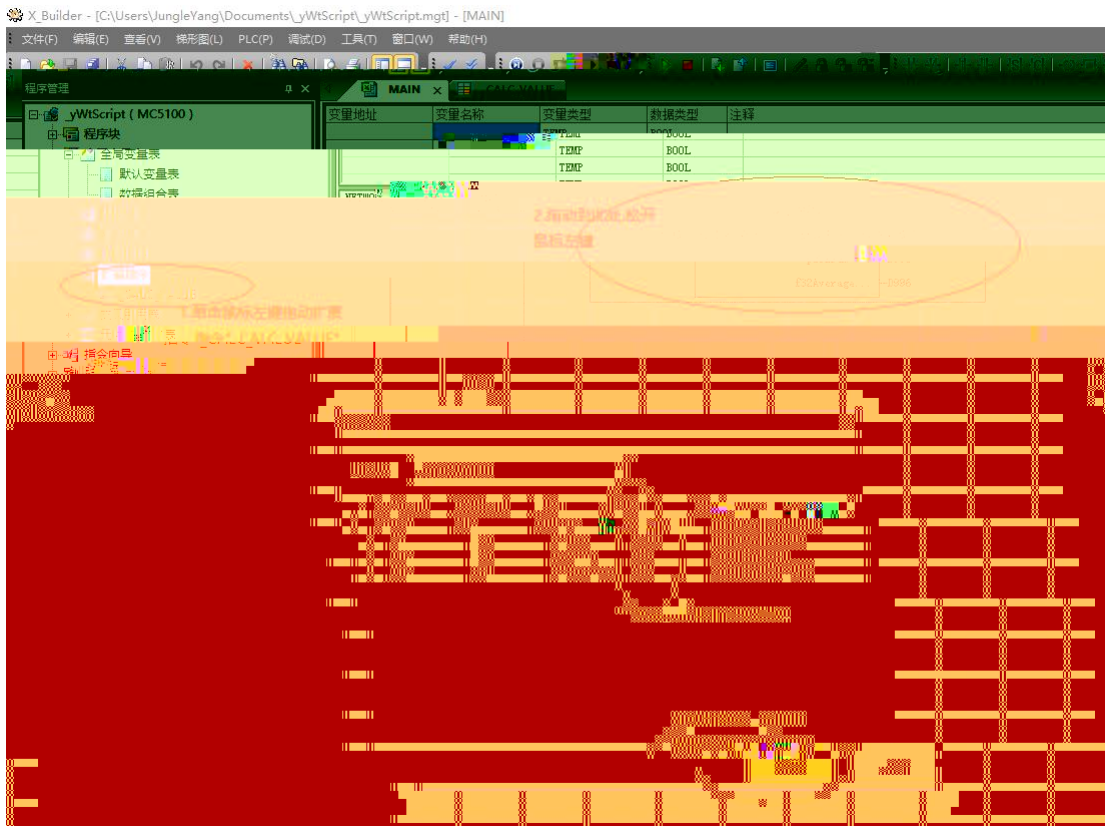
6

### 3.3

”\_CALC\_VALUE”

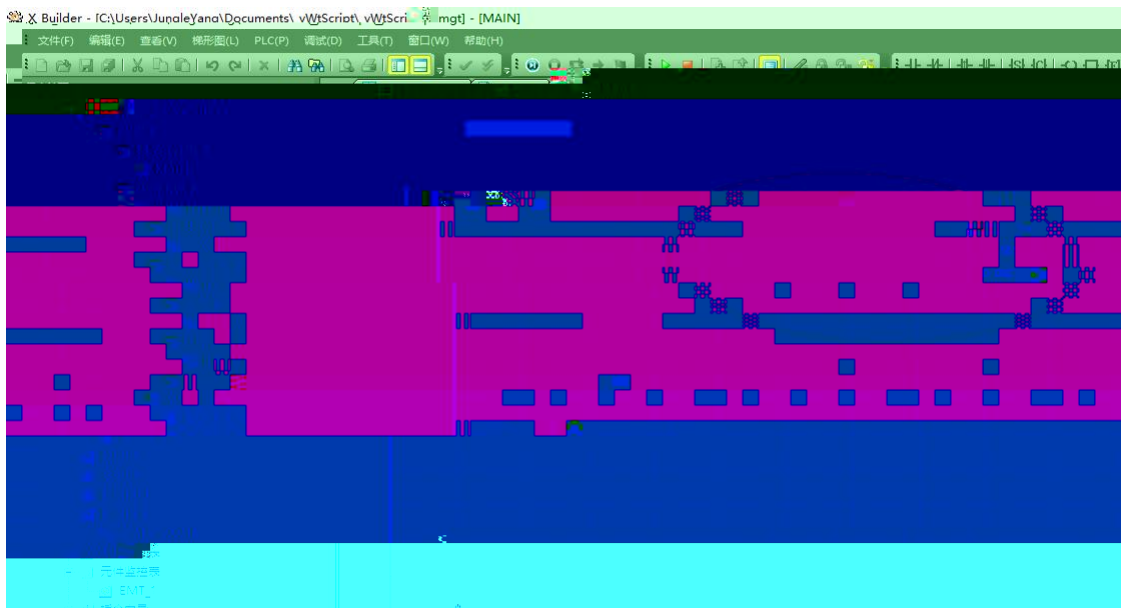


### 3.4



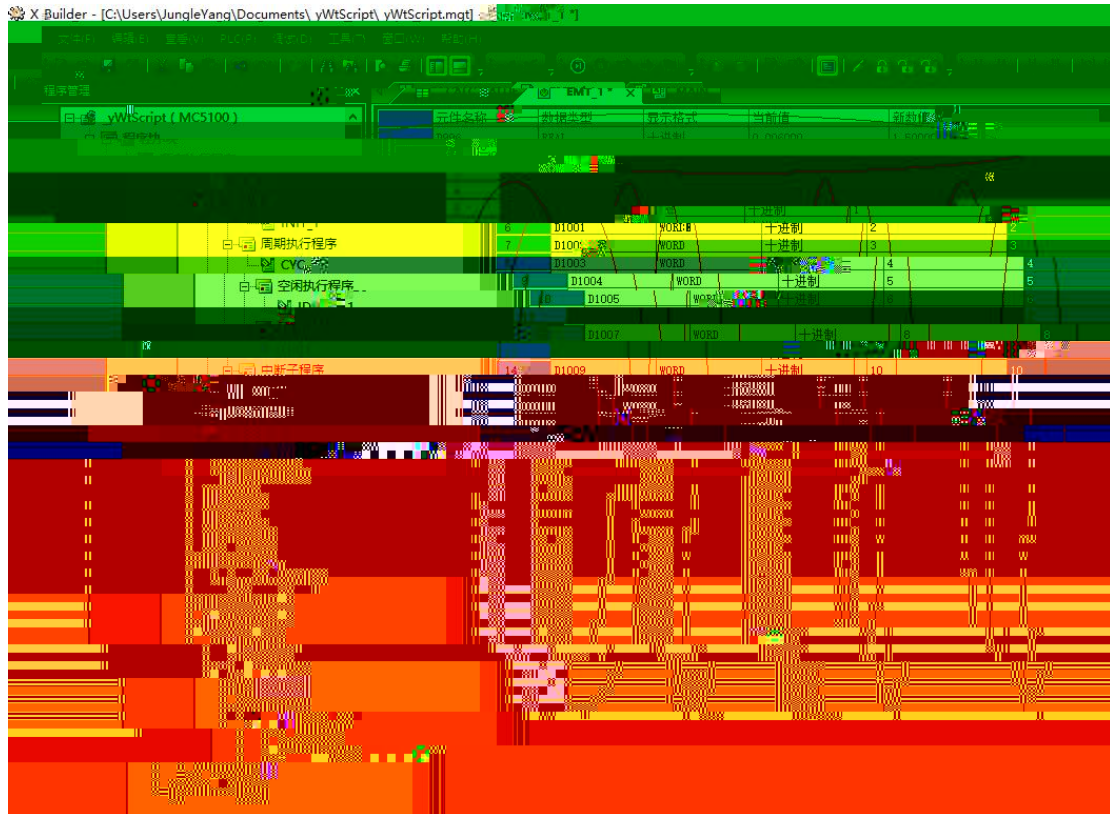
8

M10 M11 On ”\_CALC\_VALUE” \_CALC\_VALUE  
 D1000 20000 D998( ) D996(  
 )



9

## 3.5



10

## 4

### 4.1

#### 4.1.1 X

X

```

int _mc_BitOp()
{
    //Add your code here...
    if (X4)
    {
        Y10 = 1;
    }
}

```

进制编号

## 4.1.2 Y

	Y
	PLC
	Y0 ~ Y7777
	Bit
	8 , 'Y'

```

int _mc_BitOp()
{
    //Add your code here...
    if (X4)
    {
        Y10 = 1;
    }
}

```

Y位元件输入，编号位8进制

## 4.1.3 SM

SM  
PLC ST T

## 4.1.4 S

	S
	PLC S
	S0 ~ S4095
	Bit
	10 , 'S'

```
int _mc_BitOp()
{
    //Add your code here...
    if ( X4 )
    {
        S[100]=1;
        S[1000]=1;
    }
}
```

S元件支持Sxx及S[xx]输入, 编号为十进制。

## 4.1.5 T

	T
	PLC T
	T0 ~ T4095
	Bit
	10 , 'T'

```
int _mc_BitOp()
{
    //Add your code here...
    if ( T10 )
    {
    }
}
```

T元件位变量, 支持Txx及T[xx]输入, 十进制编号。

## 4.1.6 C

	C
	PLC C
	C0 ~ C4095
	Bit
	10 , 'C'

```
int _mc_BitOp()
{
    //Add your code here...
    if ( C10 )
    {
        C100 = 10;
    }
}
```

C元件输入格式支持Cxx及C[xx]十进制编号。

## 4.1.7 M

	C
	PLC M
	M0 ~ M65535
	Bit
	10 , 'M'

```
int _mc_BitOp()
```

```
{
  //Add your code here...
```



## 4.1.8 SD

	SD
	PLC SD
	SD0 ~ SD4095
	signed short
	10 , 'SD'

```
int _mc_BitOp()
```

```
{
  //Add your code here...
```



## 4.1.9 Z

	Z
	PLC Z
	Z0 ~ Z4095
	signed short
	10 , 'Z'

```
int _mc_BitOp()
```

```
{
  //Add your code here...
```

```
if ( X4 )
```

```
{
  [0...4095] = 100;
```

## 4. 1. 10 D

	D
	PLC D
	D0 ~ D65535
	signed short
	10 , 'D'

```
int _mc_BitOp()
{
    //Add your code here...
    R[4095] = 100;
}
```

D字单元直接读写支持Dxx及D[xx]

## 4. 1. 11 R

	R
	PLC R
	R0 ~ R65535
	signed short
	10 , 'R'

```
int _mc_BitOp()
{
    //Add your code here...
    if ( X4 )
    {
        R[4095] = 100;
    }
}
```

R字变量支持Rxx及R[xx]直接读写操作

## 4. 2

MC5000

PLC

32

### 4. 2. 1

	int GET_DD(unsigned short stNum)
	" D"
	stNum D
	int,

--	--

```

int _mc_BitDown()
{
    //Add your code here...
    if ( X4 )
    {
        long tmp;
        tmp = GET_DD(1000);
    }
}

```

读取D1000长整型数据到tmp

### 4.2.2

	void SET_DD(unsigned short stNum int val)
	" D"
	stNum: D
	val :

```

int _mc_BitUp()
{
    //Add your code here...
    if ( X4 )
    {
        long tmp;
        tmp = GET_DD(1000);
        SET_DD(100, tmp);
    }
}

```

将tmp值写入到长整数D150中

### 4.2.3

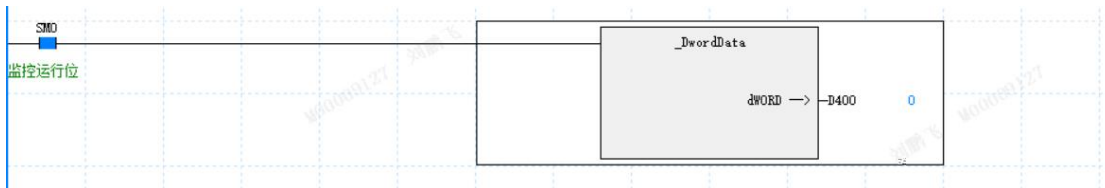
	int GET_MultiDD(int stNum int len, int *ps32Dsc)
	" D"
	stNum: D
	Len :
	ps32Dsc:
	0 ,

#### 4.2.4

### 4.2.4

	int SET_MultiDD(int stNum int len, int *ps32Src)
	" D"
	stNum: D
	Len :

	ps32Dsc:	D
	0	,



```

1          GET_Miti DD(int stNum int len, int *ps32Dsc)
SET_Miti DD(int stNum int len, int *ps32Src)          D
              dWORD    D400          D400          10

```

```

2          int *ps32Dsc int *ps32Src

```

#### 4.2.5 5N

## 4.2.6

	void SET_FD(unsigned short stNum, float val)
	" D"
	stNum: D
	val :

```
int _mc_DwordData(OUT int32 *dWORD)
{
    //Add your code here...
    uchar utmp;
    int32 *dtmp;
    float fTmp, *pfTmp;
    fTmp = GET_FD(600);
    SET_FD(700, fTmp);
}
```

读取d600内浮点数存入变量fTmp中

将浮点数fTmp的值写入d700中

## 4.2.7

	int GET_MultiFD(int stNum, int len, float *pf32Dsc)
	" D"
	stNum: D
	Len :
	ps32Dsc:
	0 ,

4.2.8

## 4.2.8



#### 4. 2. 12

	int SET_MitiDR(int stNum, int len, int *ps32Src)
	" R"
	stNum: R
	Len :
	ps32Dsc: R
	0 ,

#### 4. 2. 13

	float GET_FR(unsigned short stNum)
	" R"
	stNum R
	float,

#### 4. 2. 14

	void SET_FR(unsigned short stNum, float val)
	" R"
	stNum: R
	val :

#### 4. 2. 15

int GET\_MitiFR(int stNum, int len, float \*pf32

---

	0 ,

#### 4. 2. 16

	int SET_MultiFR(int stNum, int len, float *pf32Src)
	" R"
	stNum: R
	Len :
	ps32Dsc: R
	0 ,

#### 4. 2. 17

```

int GET_DF(int stNum)
    " F"
stNum F

int,

```

	stNum: F
	Len :
	ps32Dsc:
	0 ,

4.2.34 F0~F9

## 4. 2. 20

	int SET_MitIDF(int stNum, int len, int *ps32Src)
	" F"
	stNum: F
	Len :
	ps32Dsc: F
	0 ,

4.2.34 F0~F9

## 4. 2. 21

	float GET_FF(unsigned short stNum)
	" F"
	stNum F
	float,

4.2.34 F0~F9

## 4. 2. 22

	void SET_FF(unsigned short stNum, float val)
	" F"
	stNum: F
	val :

4.2.34 F0~F9

#### 4. 2. 23

	int SET_Mitiff(int stNum, int len, float *pf32Src)
	" F" 4
	stNum: F
	Len :
	ps32Dsc:
	0 ,

4.2.34 F0~F9

#### 4. 2. 24

	int SET_Mitiff(int stNum, int len, float *pf32Src)
	" F" 4
	stNum: F
	Len :
	ps32Dsc: F
	0 ,

4.2.34 F0~F9

#### 4. 2. 25

```

int GET_DFO(int stNum)
int GET_DF1(int stNum)
int GET_DF2(int stNum)
int GET_DF3(int stNum)
int GET_DF4(int stNum)
int GET_DF5(int stNum)
int GET_DF6(int stNum)
int GET_DF7(int stNum)
int GET_DF8(int stNum)
int GET_DF9(int

```

---

## 4. 2. 26

	void SET_DF0(int stNum, int val)
	void SET_DF1(int stNum, int val)
	void SET_DF2(int stNum, int val)
	void SET_DF3(int stNum, int val)
	void SET_DF4(int stNum, int val)
	void SET_DF5(int stNum, int val)
	void SET_DF6(int stNum, int val)
	void SET_DF7(int stNum, int val)
	void SET_DF8(int stNum, int val)
	void SET_DF9(int stNum, int val)
	" Fx"
	stNum: Fx
	val :

4.2.34 F0~F9

## 4. 2. 27

int "

## 4. 2. 28

	int SET_MultiDF0(int stNum, int len, int *ps32Src)
	int SET_MultiDF1(int stNum, int len, int *ps32Src)
	int SET_MultiDF2(int stNum, int len, int *ps32Src)
	int SET_MultiDF3(int stNum, int len, int *ps32Src)
	int SET_MultiDF4(int stNum, int len, int *ps32Src)
	int SET_MultiDF5(int stNum, int len, int *ps32Src)
	int SET_MultiDF6(int stNum, int len, int *ps32Src)
	int SET_MultiDF7(int stNum, int len, int *ps32Src)
	int SET_MultiDF8(int stNum, int len, int *ps32Src)
	int SET_MultiDF9(int stNum, int len, int *ps32Src)
	" F"
	stNum:            F
	Len :
	ps32Dsc:                    F
	0 ,

4.2.34 F0~F9

## 4. 2. 29

	float GET_FF0(unsigned short stNum)
	float GET_FF1(unsigned short stNum)
	float GET_FF2(unsigned short stNum)
	float GET_FF3(unsigned short stNum)
	float GET_FF4(unsigned short stNum)
	float GET_FF5(unsigned short stNum)
	float GET_FF6(unsigned short stNum)
	float GET_FF7(unsigned short stNum)
	float GET_FF8(unsigned short stNum)
	float GET_FF9(unsigned short stNum)
	" Fx"
	stNum    Fx
	float,

4.2.34 F0~F9

## 4. 2. 30

	voi d SET_FF0(unsig ned short stNum, fl oat val )
	voi d SET_FF1(unsig ned short stNum, fl oat val )
	voi d SET_FF2(unsig ned short stNum, fl oat val )
	voi d SET_FF3(unsig ned short stNum, fl oat val )
	voi d SET_FF4(unsig ned short stNum, fl oat val )
	voi d SET_FF5(unsig ned short stNum, fl oat val )
	voi d SET_FF6(unsig ned short stNum, fl oat val )
	voi d SET_FF7(unsig ned short stNum, fl oat val )
	voi d SET_FF8(unsig ned short stNum, fl oat val )
	voi d SET_FF9(unsig ned short stNum, fl oat val )
	" Fx"
	stNum: Fx
	val :

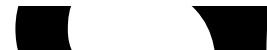
4.2.34 F0~F9

## 4. 2. 31

```

int GET_MitiFF0(int stNum, int len, float *pf32Dsc)
int GET_MitiFF1(int stNum, int len, float *pf32Dsc)
int GET_MitiFF2(int stNum, int len, float *pf32Dsc)
int GET_MitiFF3(int stNum, int len, float *pf32Dsc)
int GET_MitiFF4(int stNum, int len, float *pf32Dsc)
int GET_MitiFF5(int stNum, int len, float *pf32Dsc)
int GET_MitiFF6(int stNum, int len, float *pf32Dsc)
int GET_MitiFF7(int stNum, int len, float *pf32Dsc)
int GET_MitiFF8(int stNum, int len, float *pf32Dsc)
int GET_MitiFF9(int stNum, int len, float *pf32Dsc)
" Fx"

```







```

#define DR *(int32 *)&R
#define FD *(float *)&D
#define DR *(int32 *)&R
int _mc_pointerOP(IN int32 InPra, IN_OUT uint16 *IOPra, OUT uint16 *OPra)
{
    //Add your code here...

    int *Adr1,*Adr2,Adr3 = 620;
    long dInt1;

    float fData1,*fData2;
    float fTemp = 568.12;
    long DTemp = 654321;
    long temp;

    Adr1 = &D600;
    Adr2 = &R550;
}

```

写入目标寄存器指向的地址  
被写数据为立即数

Adr1 D600 Adr2

R550

### 4.2.36

	int GET_PS32(int *ps32Src)
	int
	ps32Src:

```

#define DR *(int32 *)&R
#define FD *(float *)&D
#define DR *(int32 *)&R
int _mc_pointerOP(IN int32 InPra, IN_OUT uint16 *IOPra, OUT uint16 *OPra)
{
    //Add your code here...

    int *Adr1,*Adr2,Adr3 = 620;
    long dInt1,*dInt2;

    float fData1,*fData2;
    float fTemp = 568.12;
    long DTemp = 654321;
    long temp;

    ...Adr1 = &D600;
    Adr2 = &R550;

    GET_PS32(Adr1,DTemp); //Adr1为指针，将长整型DTemp的值写入Adr1指向的地址（大小端调整）
    Adr1 = GET_PS32(OPra); //将指针OPra指向的地址的值读出来，赋给dInt1
    SET_PS32(Adr1+1,dInt1); //将dInt1的值经大小端调整后赋给指针（Adr1+1）指向的地址
}

```

操作数为指向地址的指针

IOPra

### 4.2.37

	int GET_S32(int s32Src)
	int 32
	s32Src:



## 4. 2. 40

	unsigned int GET_U32(unsigned int u32Src)
	Unsigned int                      32
	u32Src:
	PLC

\*2

```

#define int (*int32 *)16
#define int (*int32 *)16
// 将指针IOPra指向的地址的值读出来，赋给dInt1
// 将dInt1的值经大小端调整后赋给指针 (Adr1+
) 指向的地址
SET_PS32(Adr1+1, dInt1); // 将长整数dInt1的值经大小端调整后
dInt1 = GET_U32(Adr1); // 将长整数dInt1的值经大小端调整后
// 将经过大小端调整的长整数结果dInt1的值直接赋给R652
    
```

## 4. 2. 41

	void SET_PF32(float *pf32Dsc, float f32Src)
	f32Src                      pf32Dsc
	f32Src:
	pf32Dsc:

4.4.23

## 4. 2. 42

	float GET_PF32(float *pf32Src)
	int
	pf32Src:



---

5

5.1.1

