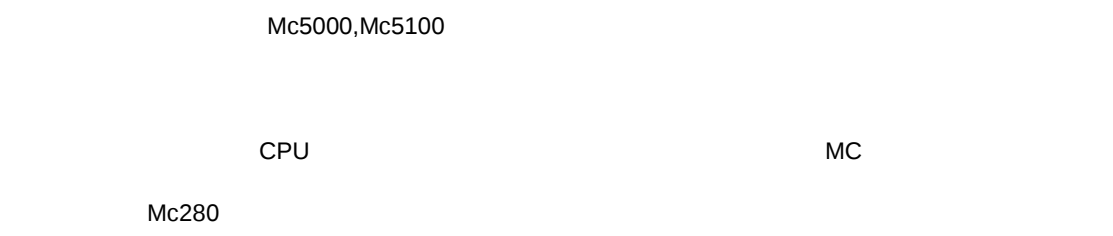

	MGT-MODE-3-003		
	V1.02		
			2024-12-12

1

1.1

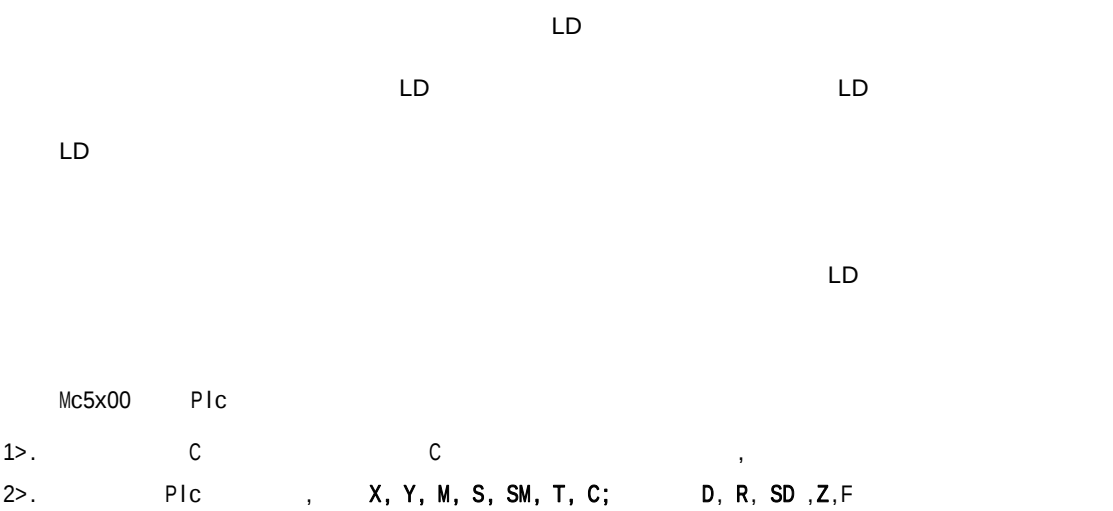


1.2

	Plc()
Mc5000, Mc5100	
MEGreator	MC5000 MU
	MEGreator

2

2.1



X , D

3> . , 64K

4> . ANSI C , C

5> . C ;

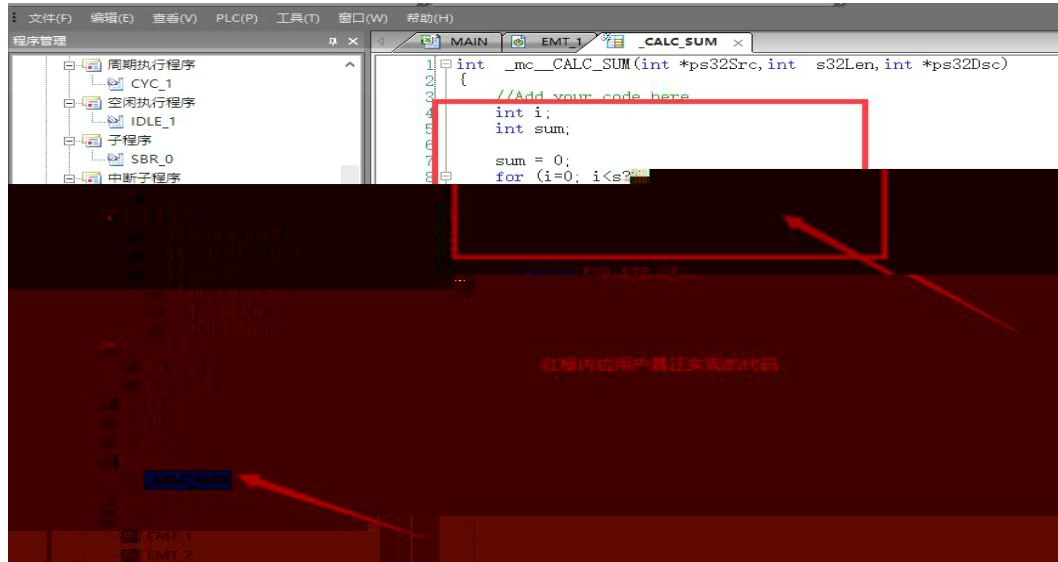
6> .

2.2

1>

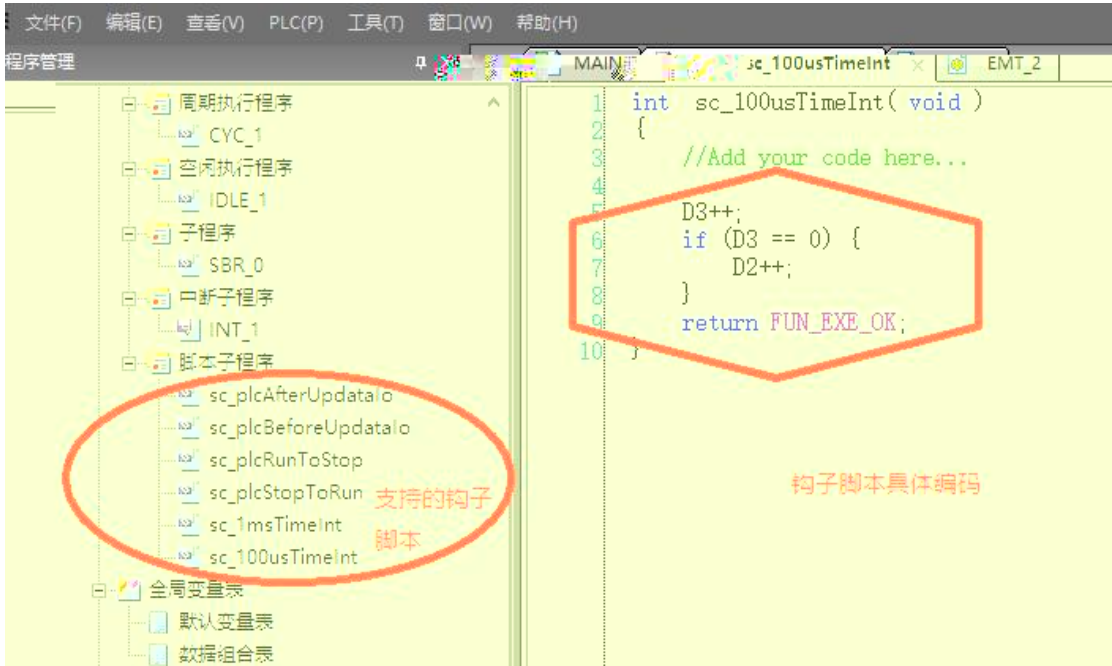


1



2

1>



2.3

LD C
plc

export_module.h	Plc
user_common.c	
user_common.h	

3

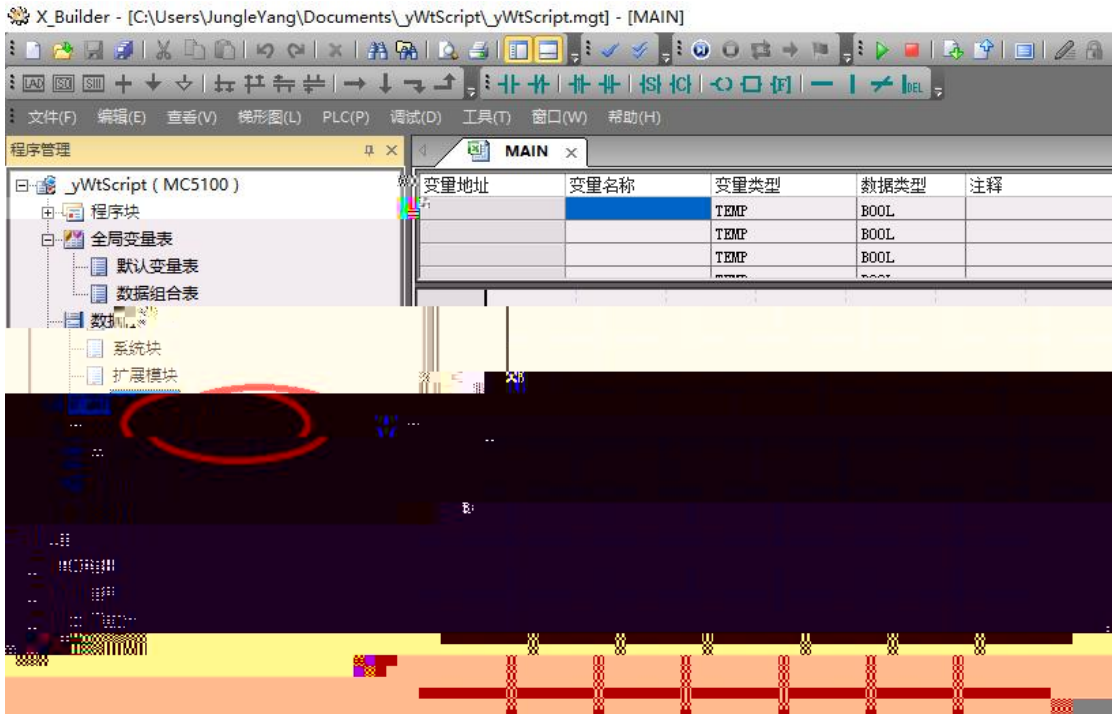
xBuild C

D1000 20000 D998()

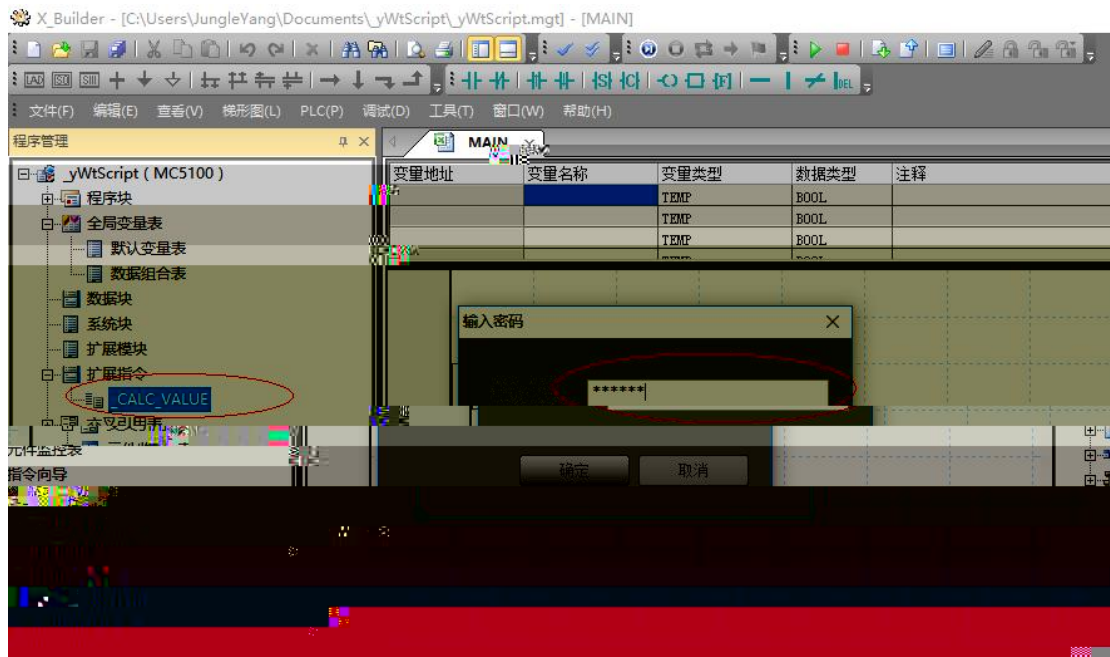
D996()

3.1

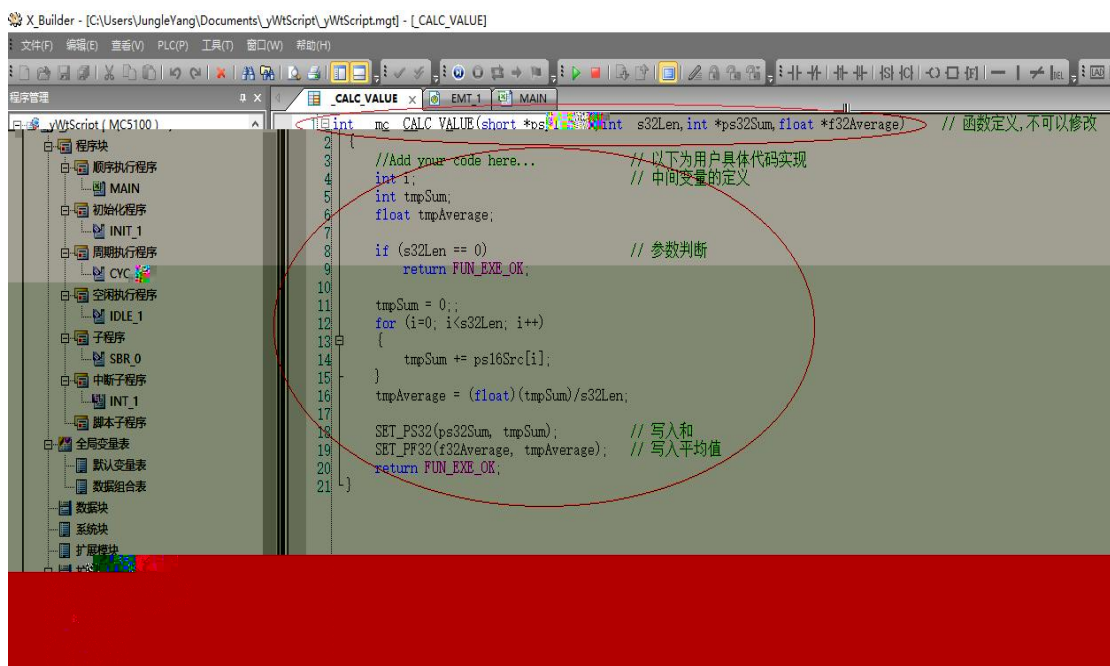
“ ” “ ” “ ”



3.3

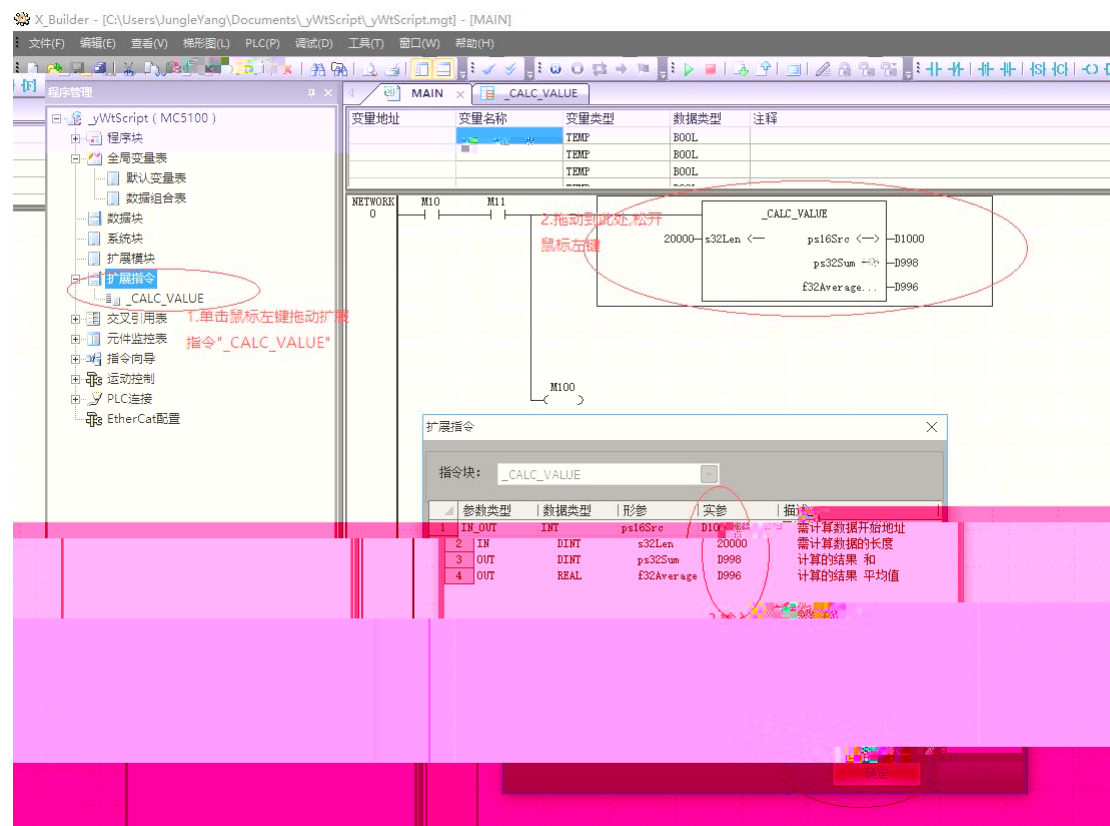


6

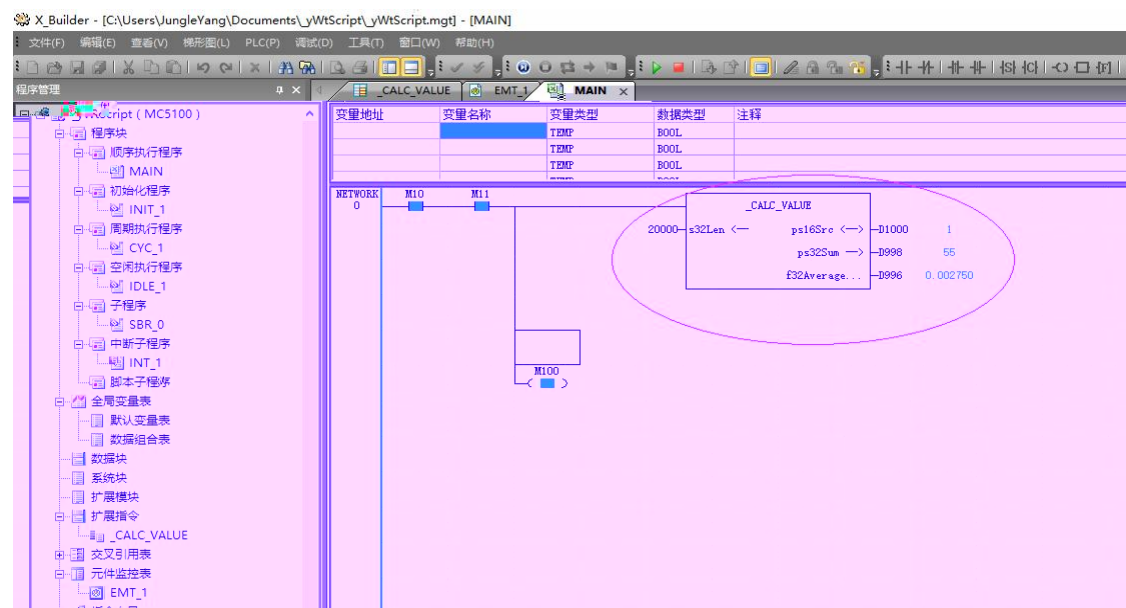


7

3.4

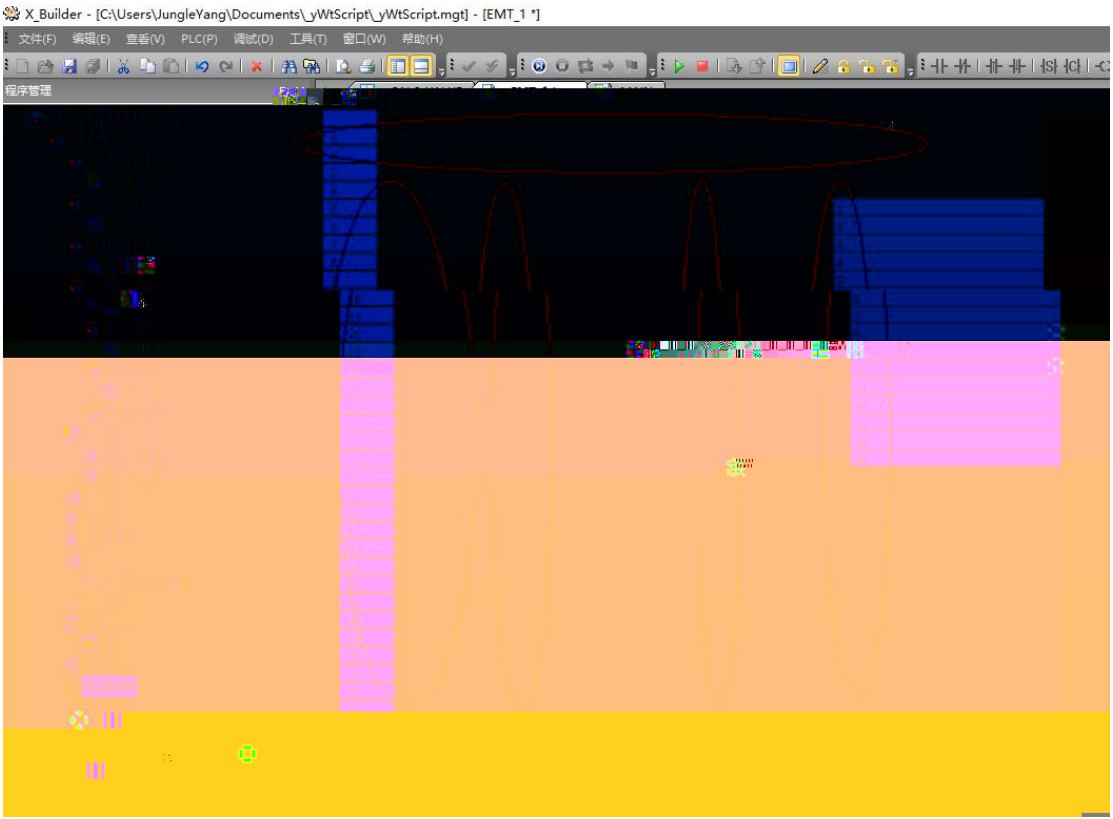


8



9

3.5



4

4.1

4.1.1 X


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

8进制编号

X[10]

X[XX]

4.1.2 Y


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

Y位元件输入，编号位8进制

Y[15]

Y[XX]

4.1.3 SM


```
int _mc__BitOp()  
{  
    //Add your code here...  
    if ( X4 )  
    {  
        SM[40]=1;  
        SM[41]=1;  
    }  
}
```

SM位元件支持SMxx及SM[xx]

SM

4.1.4 S


```
int __mc__BitOp()  
{  
    //Add your code here...  
    if ( X4 )  
    {  
        S[100]=1;  
        S101 = 1;  
    }  
}
```

S位元件支持Sxx及S[xx]输入，编号为十进制

4.1.5 T


```
int __mc__BitOp()  
{  
    //Add your code here...  
    if (T10)  
    {  
        D[102] =20;  
    }  
}
```

T元件位变量，支持1xx及1[xx]输入，十进制编号

4.1.6 C


```
int __mc__BitOp()  
{  
    //Add your code here...  
    C[100] = 1;  
}
```

C位元件输入格式支持Cxx及C[xx],十进制编号

4.1.7 M


```
int __mc__BitOp()  
{  
    //Add your code here...
```

M 值函数支持 $\sin$ 及 $\cos$ 函数，输入十进制值即可。

4.1.8 SD


```
int main()
{
    
$$m_{avg} = \frac{R_i + \Omega R_o}{2\pi r_o}$$

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

```
if ( X4 )
{
    D110 = SD101;
    D[111] = SD[102];
}
```

32 SD

16 SD

32

4.1.9 Z


```
int __mc__BitOp()  
{  
    //Add your code here...
```

```
if ( X4 )
{
    7字元件支持 7xx 及 7[xx] 读写
}
```

4.1.10 D


```
int _mc__BitOp()  
{  
    //Add your code here...  
    if ( X4 )  
    {  
        SD101;  
        SD [102];  
    }  
}
```

D字单元直接读写支持Dxx及D[xx]

4.1.11 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

4.2.1 D

	int GET_DD(unsigned short stNum)
	“ ”
	stNum:

--	--

```

int _mc_BitOp()
{
    //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

D

	void SET_DD(unsigned short stNum, int val)
	“ ”

```

int _mc_BitUp()
{
    //Add your code here...

    if ( X4 )
    {
        long tmp;
        tmp = GET_DD(1000);
        SET_DD(150, tmp);
    }
}

```

将tmp值写入到长整数D150中

4.2.3

D

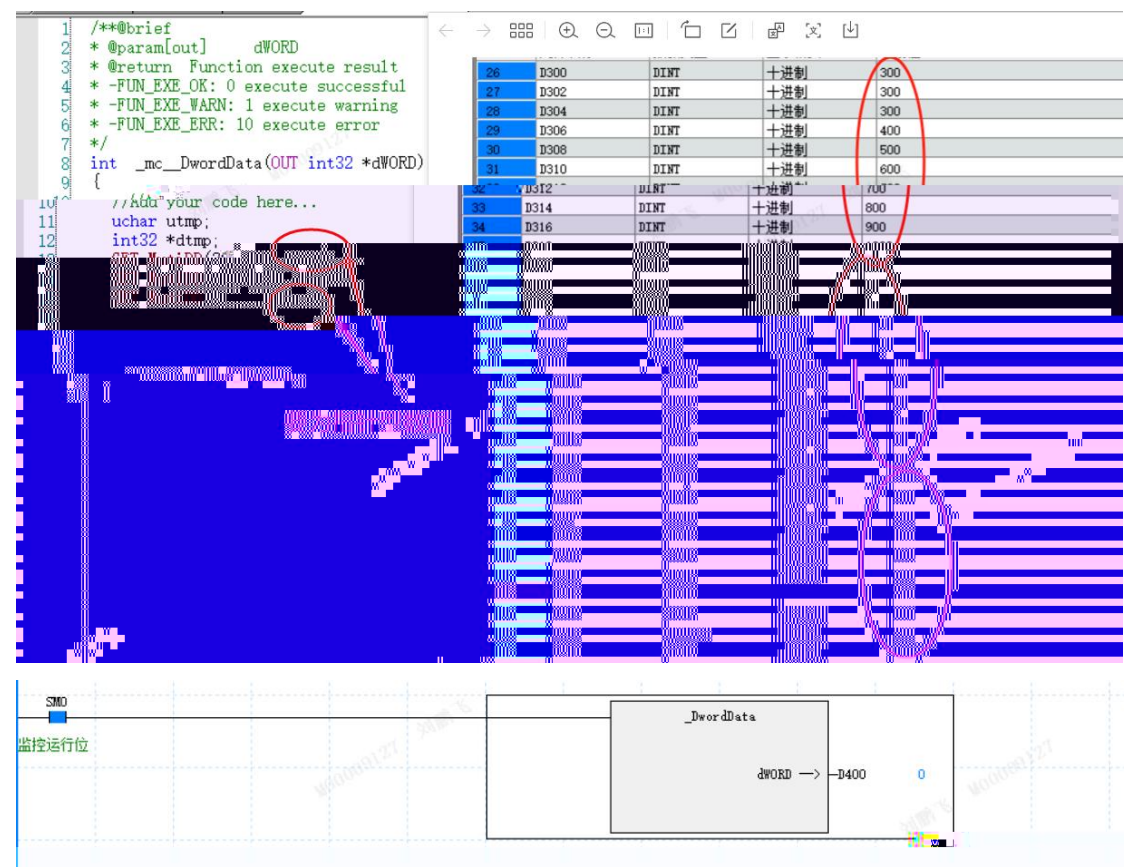
	int GET_MutiDD(int stNum, int len, int *ps32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.4

D

	int SET_MutiDD(int stNum, int len, int *ps32Src)
	“ ”
	stNum
	Len :

	ps32Dsc:	D



1

GET_MutiDD(int stNum, int len, int *ps32Dsc)

SET_MutiDD(int stNum, int len, int *ps32Src)

D

dWORD

D400

D400

10

2

int *ps32Dsc

int *ps32Src

4.2.5 D

	float GET_FD(unsigned short stNum)
	“ ”
	stNum:

4.2.6

D

	void SET_FD(unsigned short stNum, float 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

D

	int GET_MutiFD(int stNum, int len, float *pf32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.8

D

	int SET_MutiFD(int stNum, int len, float *pf32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: D

```
int _mc_DwordData(OUT int32 *dWORD)
{
    //Add your code here...
    uchar utmp;
    int32 *dtmp;
    float fTmp; *pfTmp;
    int i = 610;
    GET_MutiFD(i, 5, pfTmp);
    SET_MutiFD(i+20, 5, pfTmp);
}
```

读取从610开始的5个浮点数,存入指针变量pfTmp中

将指针变量pfTmp中连续5个浮点数存入D630开始地址

4.2.9 R

	int GET_DR(unsigned short stNum)
	“ ”
	stNum: R

4.2.10 R

	void SET_DR(unsigned short stNum, int val)
	“ ”

4.2.11 R

	int GET_MutiDR(int stNum, int len, int *ps32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.12

R

	int SET_MutiDR(int stNum, int len, int *ps32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: R

4.2.13

R

	float GET_FR(unsigned short stNum)
	“ ”
	stNum: R

4.2.14

R

	void SET_FR(unsigned short stNum, float val)
	“ ”

4.2.15

R

	int GET_MutiFR(int stNum, int len, float *pf32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.16

R

	int SET_MutiFR(int stNum, int len, float *pf32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: R

4.2.17

F

F0 ~ F9

	int GET_DF(int stNum)
	“ ”
	stNum: F

4.2.18

F

F0 ~ F9

	void SET_DF(int stNum, int val)
	“ ”

4.2.19

F

F0 ~ F9

	int GET_MutiDF(int stNum, int len, int *ps32Dsc)
	“ ”

	stNum
	Len :
	ps32Dsc:

4.2.20

F

F0 ~ F9

	int SET_MutiDF(int stNum, int len, int *ps32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: F

4.2.21

F

F0 ~ F9

	float GET_FF(unsigned short stNum)
	“ ”
	stNum: F

4.2.22

F

F0 ~ F9

	void SET_FF(unsigned short stNum, float val)
	“ ”

4.2.23

F**F0 ~ F9**

	int GET_MutiFF(int stNum, int len, float *pf32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.24

F**F0 ~ F9**

	int SET_MutiFF(int stNum, int len, float *pf32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: F

4.2.25

Fx**F0 ... F9**

	int GET_DF0(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 stNum)
	“ ”
	stNum: Fx

4.2.26

Fx**F0 ... F9**

	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)
	“ ”

4.2.27

Fx**F0 ... F9**

	int GET_MutiDF0(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF1(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF2(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF3(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF4(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF5(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF6(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF7(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF8(int stNum, int len, int *ps32Dsc)
	int GET_MutiDF9(int stNum, int len, int *ps32Dsc)
	“ ”
	stNum
	Len :
	ps32Dsc:

4.2.28

Fx**F0 ... F9**

	int SET_MutiDF0(int stNum, int len, int *ps32Src)
	int SET_MutiDF1(int stNum, int len, int *ps32Src)
	int SET_MutiDF2(int stNum, int len, int *ps32Src)
	int SET_MutiDF3(int stNum, int len, int *ps32Src)
	int SET_MutiDF4(int stNum, int len, int *ps32Src)
	int SET_MutiDF5(int stNum, int len, int *ps32Src)
	int SET_MutiDF6(int stNum, int len, int *ps32Src)
	int SET_MutiDF7(int stNum, int len, int *ps32Src)
	int SET_MutiDF8(int stNum, int len, int *ps32Src)
	int SET_MutiDF9(int stNum, int len, int *ps32Src)
	“ ”
	stNum
	Len :
	ps32Dsc: F

4.2.29

Fx**F0 ... F9**

	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)
	“ ”
	stNum: Fx

4.2.30

Fx**F0 ... F9**

	void SET_FF0(unsigned short stNum, float val)
	void SET_FF1(unsigned short stNum, float val)
	void SET_FF2(unsigned short stNum, float val)
	void SET_FF3(unsigned short stNum, float val)
	void SET_FF4(unsigned short stNum, float val)
	void SET_FF5(unsigned short stNum, float val)
	void SET_FF6(unsigned short stNum, float val)
	void SET_FF7(unsigned short stNum, float val)
	void SET_FF8(unsigned short stNum, float val)
	void SET_FF9(unsigned short stNum, float val)
	“ ”

4.2.31

Fx**F0 ... F9**

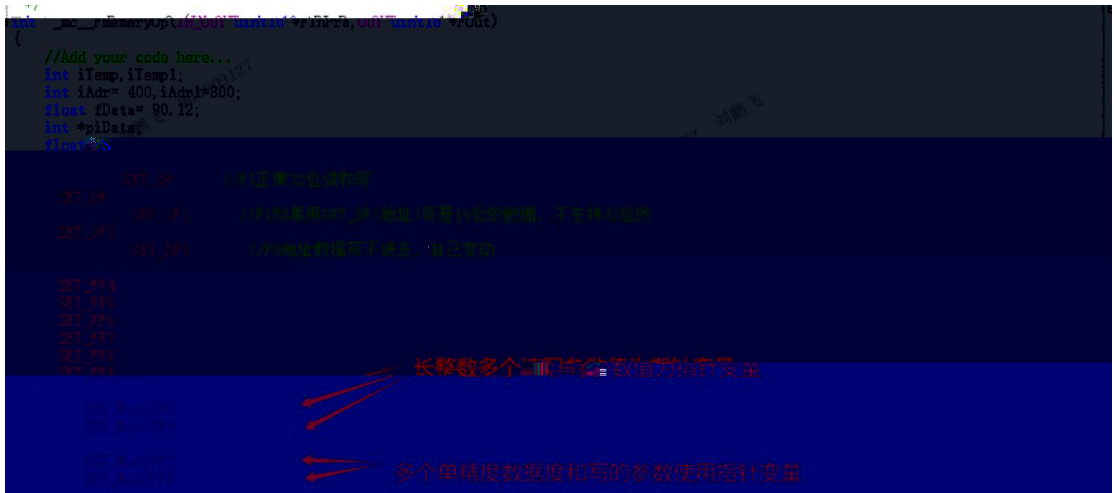
```
void GET_MutiFF0(int stNum, int len, float *pf32Dsc)
int GET_MutiFF1(int stNum, int len, float *pf32Dsc)
int GET_MutiFF2(int stNum, int len, float *pf32Dsc)
int GET_MutiFF3(int stNum, int len, float *pf32Dsc)
int GET_MutiFF4(int stNum, int len, float *pf32Dsc)
int GET_MutiFF5(int stNum, int len, float *pf32Dsc)
int GET_MutiFF6(int stNum, int len, float *pf32Dsc)
int GET_MutiFF7(int stNum, int len, float *pf32Dsc)
int GET_MutiFF8(int stNum, int len, float *pf32Dsc)
int GET_MutiFF9(int stNum, int len, float *pf32Dsc)
“ ”
```

4.2.34

Fx

F0 ... F9

	void SET_F0(int stNum, signed short val)
	void SET_F1(int stNum, signed short val)
	void SET_F2(int stNum, signed short val)
	void SET_F3(int stNum, signed short val)
	void SET_F4(int stNum, signed short val)
	void SET_F5(int stNum, signed short val)
	void SET_F6(int stNum, signed short val)
	void SET_F7(int stNum, signed short val)
	void SET_F8(int stNum, signed short val)
	void SET_F9(int stNum, signed short val)
	“ ”



4.2.35

int

()

	void SET_PS32(int *ps32Dsc, int s32Src)
	s32Src ps32Dsc
	s32Src:
	ps32Dsc:



4.2.38 unsigned int ()

	void SET_PU32(unsigned int *pu32Dsc, unsigned int u32Src)
	u32Src pu32Dsc
	u32Src:
	pu32Dsc:

4.2.39 unsigned int ()

	unsigned int GET_PU32(unsigned int *pu32Src)
	pu32Src:

4.2.40 unsigned int ()

	unsigned int GET_U32(unsigned int u32Src)
	u32Src:

```
#define DD *(int32 *)&D
#define FD *(float *)&D
#define D0 0
#define D1 1
#define D2 2
#define D3 3
#define D4 4
#define D5 5
#define D6 6
#define D7 7
#define D8 8
#define D9 9
#define D10 10
#define D11 11
#define D12 12
#define D13 13
#define D14 14
#define D15 15
#define D16 16
#define D17 17
#define D18 18
#define D19 19
#define D20 20
#define D21 21
#define D22 22
#define D23 23
#define D24 24
#define D25 25
#define D26 26
#define D27 27
#define D28 28
#define D29 29
#define D30 30
#define D31 31
#define D32 32
#define D33 33
#define D34 34
#define D35 35
#define D36 36
#define D37 37
#define D38 38
#define D39 39
#define D40 40
#define D41 41
#define D42 42
#define D43 43
#define D44 44
#define D45 45
#define D46 46
#define D47 47
#define D48 48
#define D49 49
#define D50 50
#define D51 51
#define D52 52
#define D53 53
#define D54 54
#define D55 55
#define D56 56
#define D57 57
#define D58 58
#define D59 59
#define D60 60
#define D61 61
#define D62 62
#define D63 63
#define D64 64
#define D65 65
#define D66 66
#define D67 67
#define D68 68
#define D69 69
#define D70 70
#define D71 71
#define D72 72
#define D73 73
#define D74 74
#define D75 75
#define D76 76
#define D77 77
#define D78 78
#define D79 79
#define D80 80
#define D81 81
#define D82 82
#define D83 83
#define D84 84
#define D85 85
#define D86 86
#define D87 87
#define D88 88
#define D89 89
#define D90 90
#define D91 91
#define D92 92
#define D93 93
#define D94 94
#define D95 95
#define D96 96
#define D97 97
#define D98 98
#define D99 99
#define D100 100
#define D101 101
#define D102 102
#define D103 103
#define D104 104
#define D105 105
#define D106 106
#define D107 107
#define D108 108
#define D109 109
#define D110 110
#define D111 111
#define D112 112
#define D113 113
#define D114 114
#define D115 115
#define D116 116
#define D117 117
#define D118 118
#define D119 119
#define D120 120
#define D121 121
#define D122 122
#define D123 123
#define D124 124
#define D125 125
#define D126 126
#define D127 127
#define D128 128
#define D129 129
#define D130 130
#define D131 131
#define D132 132
#define D133 133
#define D134 134
#define D135 135
#define D136 136
#define D137 137
#define D138 138
#define D139 139
#define D140 140
#define D141 141
#define D142 142
#define D143 143
#define D144 144
#define D145 145
#define D146 146
#define D147 147
#define D148 148
#define D149 149
#define D150 150
#define D151 151
#define D152 152
#define D153 153
#define D154 154
#define D155 155
#define D156 156
#define D157 157
#define D158 158
#define D159 159
#define D160 160
#define D161 161
#define D162 162
#define D163 163
#define D164 164
#define D165 165
#define D166 166
#define D167 167
#define D168 168
#define D169 169
#define D170 170
#define D171 171
#define D172 172
#define D173 173
#define D174 174
#define D175 175
#define D176 176
#define D177 177
#define D178 178
#define D179 179
#define D180 180
#define D181 181
#define D182 182
#define D183 183
#define D184 184
#define D185 185
#define D186 186
#define D187 187
#define D188 188
#define D189 189
#define D190 190
#define D191 191
#define D192 192
#define D193 193
#define D194 194
#define D195 195
#define D196 196
#define D197 197
#define D198 198
#define D199 199
#define D200 200
#define D201 201
#define D202 202
#define D203 203
#define D204 204
#define D205 205
#define D206 206
#define D207 207
#define D208 208
#define D209 209
#define D210 210
#define D211 211
#define D212 212
#define D213 213
#define D214 214
#define D215 215
#define D216 216
#define D217 217
#define D218 218
#define D219 219
#define D220 220
#define D221 221
#define D222 222
#define D223 223
#define D224 224
#define D225 225
#define D226 226
#define D227 227
#define D228 228
#define D229 229
#define D230 230
#define D231 231
#define D232 232
#define D233 233
#define D234 234
#define D235 235
#define D236 236
#define D237 237
#define D238 238
#define D239 239
#define D240 240
#define D241 241
#define D242 242
#define D243 243
#define D244 244
#define D245 245
#define D246 246
#define D247 247
#define D248 248
#define D249 249
#define D250 250
#define D251 251
#define D252 252
#define D253 253
#define D254 254
#define D255 255
#define D256 256
#define D257 257
#define D258 258
#define D259 259
#define D260 260
#define D261 261
#define D262 262
#define D263 263
#define D264 264
#define D265 265
#define D266 266
#define D267 267
#define D268 268
#define D269 269
#define D270 270
#define D271 271
#define D272 272
#define D273 273
#define D274 274
#define D275 275
#define D276 276
#define D277 277
#define D278 278
#define D279 279
#define D280 280
#define D281 281
#define D282 282
#define D283 283
#define D284 284
#define D285 285
#define D286 286
#define D287 287
#define D288 288
#define D289 289
#define D290 290
#define D291 291
#define D292 292
#define D293 293
#define D294 294
#define D295 295
#define D296 296
#define D297 297
#define D298 298
#define D299 299
#define D300 300
#define D301 301
#define D302 302
#define D303 303
#define D304 304
#define D305 305
#define D306 306
#define D307 307
#define D308 308
#define D309 309
#define D310 310
#define D311 311
#define D312 312
#define D313 313
#define D314 314
#define D315 315
#define D316 316
#define D317 317
#define D318 318
#define D319 319
#define D320 320
#define D321 321
#define D322 322
#define D323 323
#define D324 324
#define D325 325
#define D326 326
#define D327 327
#define D328 328
#define D329 329
#define D330 330
#define D331 331
#define D332 332
#define D333 333
#define D334 334
#define D335 335
#define D336 336
#define D337 337
#define D338 338
#define D339 339
#define D340 340
#define D341 341
#define D342 342
#define D343 343
#define D344 344
#define D345 345
#define D346 346
#define D347 347
#define D348 348
#define D349 349
#define D350 350
#define D351 351
#define D352 352
#define D353 353
#define D354 354
#define D355 355
#define D356 356
#define D357 357
#define D358 358
#define D359 359
#define D360 360
#define D361 361
#define D362 362
#define D363 363
#define D364 364
#define D365 365
#define D366 366
#define D367 367
#define D368 368
#define D369 369
#define D370 370
#define D371 371
#define D372 372
#define D373 373
#define D374 374
#define D375 375
#define D376 376
#define D377 377
#define D378 378
#define D379 379
#define D380 380
#define D381 381
#define D382 382
#define D383 383
#define D384 384
#define D385 385
#define D386 386
#define D387 387
#define D388 388
#define D389 389
#define D390 390
#define D391 391
#define D392 392
#define D393 393
#define D394 394
#define D395 395
#define D396 396
#define D397 397
#define D398 398
#define D399 399
#define D400 400
#define D401 401
#define D402 402
#define D403 403
#define D404 404
#define D405 405
#define D406 406
#define D407 407
#define D408 408
#define D409 409
#define D410 410
#define D411 411
#define D412 412
#define D413 413
#define D414 414
#define D415 415
#define D416 416
#define D417 417
#define D418 418
#define D419 419
#define D420 420
#define D421 421
#define D422 422
#define D423 423
#define D424 424
#define D425 425
#define D426 426
#define D427 427
#define D428 428
#define D
```

4.2.41 float ()

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

4.2.42 float ()

	float GET_PF32(float *pf32Src)
	pf32Src:

4.2.43 float ()

	float GET_F32(float f32Src)
	f32Src:

```
10 //
11 #define DD *(int32 *)&D
12 #define FD *(float *)&D
13 #define DR *(int32 *)&R
14 int __mc_pointerOP(IN int32 InPra, IN_OUT uint16 *IOPra, OUT uint16 *OPra)
15 {
16     //Add your code here...
17     int *Adr1,*Adr2,Adr3 = 620;
18     long dTmp;
19     float fData1,*fData2;
20     float fTemp = 568.12;
21     short sTemp = 123;
22     Adr1 = 400;
23     dTmp = GET_PU32(Adr2);
24     GET_F32(Adr1+20, dTmp);
```

dTmp = GET_PU32(Adr2);
GET_F32(Adr1+20, dTmp);

5

5.1.1 modbus crc

	unsigned short _ycrcModbus(unsigned char *data,unsigned int length)
	data: Crc
	Length: Crc

5.1.2 ccit crc

	unsigned short _ycrcCcitt(unsigned char *ptr, unsigned int len)
	data: Crc
	Length: Crc

6

