

Keil C51 Bugfix Report

Modifications, Enhancements and Corrections in Keil C51 Compiler Version 7.00

Contents:

.....	
Part 1: C51 Version 7.00	2
Part 2: C51 Version 7.01	11

C51 Version 7.00 contains support for Philips 80C51MX, SmartMX, Dallas 390, far memory and LX51 OMF2/O2 support. Several corrections are relevant only for specific options or architectures and do not apply to generic 8051 targets.

Modifications and Corrections in C51 Compiler from Version 7.00 to Version 7.01.
Note C51 Version 7.00 Beta 6 is identical with C51 Version 7.00.

Part 1: C51 Version 7.00

☒ Corrected: General Protection ErrorPF

☒ Improvement: DeadC.C1: Dead code Optimization improvement:

Sample: C51 DeadC.C1 CODE

☒ Corrected: Missing MOV A,Rn when using SFR ACC to pass #pragma asm/endasm parameters
#pragma src

```
sfr ACC = 0xE0;

unsigned char compare_maske(unsigned char maske) {
    ACC = maske;    // Diese Zeile wird nicht übersetzt
#pragma asm
    NOP
#pragma endasm
    return(1);
}
```

☒ Improvement: PDATA High address implemented (MX+D390+O2)

```
char pdata b;
char *pb;
char pdata *b1;
char xdata *xp;

void main (void) {
    pb = b1;
    pb = &b;
    xp = b1;
    xp = &b;
}
```

☒ Corrected: Potential General Protection Error in C preprocessor step

☒ Improvement: 'far _at_' added

```
char    xdata xbul [1024] _at_ 0x8000;
```

```
char const far bull [1024] _at_ 0x800100;

char  c1, c2;
void main (void) {
    c2 = xbul[1023];
    c1 = bull[0];
    c2 = bull[1023];
}
```

☒ **Corrected:** LocSym debug info incomplete with O2

```
int z;
int tab[5];

void main (void) {
    z = 5;
    {
        short bull;
        bull = tab[z];
        z    = bull;
    }
}
```

☒ **Corrected:** Incorrect initialization (SRC is Ok)

```
#pragma O2 VARBANKING
#include <absacc.h>
char far * code arr[] = {
    &FVAR(char, 0x11234),
    ((char far *) (0x10000) + 0x11234L),
    ((char far *) (0x10000 + 0x15678L)),
    ((char *) (0x10000 + 0x15678L)),
    &FVAR(char, 0x20000),
    ((char far *) (0x20000) + 0x10000L),
    &FVAR(char, 0x30000),
    ((char far *) (0x30000) + 0x10000L),
    &FVAR(char, 0x40000),
    ((char far *) (0x40000) + 0x10000L),
    ((char far *) (0x50000) + 0x10000L),
    ((char far *) (0x60000) + 0x10000L),
    ((char far *) (0x70000) + 0x10000L),
};
CORR: Asm251.C (x_cons2 /16.5.2002/)
```

☒ **Corrected + Improvment:** 80C51MX + ROM(HUGE) Indirect Calls with Parameters might be incorrect for Indirect Calls with Parameters improved on other platforms

Note: Library Functions ICALL2 and ICALLX2 introduced

☒ **Corrected:** PUSH/POP order for DPTR register reversed in following code

```
#pragma LARGE OT(7, SPEED)

typedef unsigned char BYTE;
typedef unsigned int  WORD;

typedef unsigned char LOGDST;
typedef unsigned char LOGSRC;
typedef unsigned char SALVO;
```

```
typedef unsigned char CMDDEF;
typedef unsigned char JOYSTICK;
typedef unsigned char XPOINTDEST;
```

```
#define MAXSWITCHES 20
```

```
// switch definition
```

```
typedef struct swtype_tag {
    BYTE t;                                // enumerated type
    union {
        WORD gdata;                       // generic data from download
        LOGDST *dldp;                     //
        SALVO *salvoptr;                  //
        CMDDEF *cmdptr;                   //
        WORD stat_interval;               //
        WORD bkawaymask;                  //
    }arg1;                                // argument one of download
    union arg2_tag {
        WORD gdata;                       // generic data from download
        LOGSRC *sldp;                     //
        LOGDST *dldp;                     //
    }arg2;                                // argument two of download
} SWTYPE;
```

```
// list of lists
```

```
typedef struct lol_tag {
    SWTYPE st[MAXSWITCHES];              // switch definitions
    WORD mycrc;                          // the crc I calculated following a download
    BYTE valid_levels;                   // which levels this panel can change
    char *comment;                       // downloaded comment field
    WORD comment_length;                 // length of downloaded comment
    char *name;                          // downloaded name field
    WORD name_length;                    // length of downloaded name
    WORD logical_crc;                    // downloaded constant
    WORD map_style;                      // downloaded constant
    LOGSRC *sd;                          // pointer
    LOGDST *dd;                          // pointer
    CMDDEF *cd;                          // pointer
    SALVO *vd;                           // pointer
    JOYSTICK *jd;                        // pointer
    JOYSTICK *jSrc;                      // pointer
    XPOINTDEST *xptrd;
} LOL;
```

```
char *p = "BBBB";
```

```
extern xyparse (char *p, char *s, int *, int *, int *, int *, int *);
int tmp1 = 1, tmp2 = 2, tmp3 = 3, tmp4 = 4, tmp5 = 5;
LOL the_st;
LOL *lol = &the_st;
```

```
void main (void) {
    char c;
```

```
    switch (c = *p++) {
        case 'B':                        /* Button function definition */
// xyparse(p, "55554", &tmp1, &tmp2, &tmp3, &tmp4, &tmp5);

        if (tmp1 == 0 && tmp2 < 2)        /* misc. group */
            ;
        else if (tmp1 == 1 && tmp2 < 32)    /* destination group */
            tmp2 += 2;
        else if (tmp1 == 2 && tmp2 < 32)    /* source group */
            tmp2 += 34;
        else
```

```

    return;

    lol->st[tmp2].t = (BYTE) tmp3;    // OK
    lol->st[tmp2].arg1.gdata = tmp4;   // reverses bytes on store !!!!!!!
    lol->st[tmp2].arg2.gdata = tmp5;   // OK
    break;
}
}

```

☒ Corrected: Code Rearrange with Float Compare might be wrong

```

void main (void) {
    float          pf_high;
    float          pf_less;
    unsigned char   i;
    static float    FloatValue[1];

    for (i=0; i<1; i++) {
        pf_high = 4.67;
        pf_less = 2.5;
        if (pf_high > pf_less) { // wrong code re-arrange after float compare
            FloatValue[i] = 1.0; // OK
        }
        else {
            FloatValue[i] = 0.0; // ERROR
        }
    }

    while (1);
}

```

☒ Corrected: Fatal Error: Cannot Optimize Function on following Code

```

#define SUCCESS 0
#define CONNECTED 1
#define INVALID_STATE 2

typedef unsigned char  UNS8_t;
typedef unsigned short UNS16_t;
typedef unsigned long  UNS32_t;

typedef UNS16_t      Result_t;
typedef UNS32_t      Timer_t;

typedef struct SMsg {
    UNS16_t      size;
    UNS8_t       *pData;
} Msg_t;

typedef struct SMsgStack {
    UNS8_t       Fcs;
    struct SMsgStack *pNext;
    Msg_t        pMsg [4];
} MsgStack_t;

typedef struct PORT {
    UNS8_t       data1;
    UNS8_t       data2;
    UNS8_t       state;
    UNS16_t      size;
}

```

```

} groundPort_t;

UNS16_t verifyIdentifier ( UNS16_t );
UNS16_t sendData(UNS16_t,MsgStack_t *);

extern groundPort_t * groundPort;

Result_t write (UNS16_t identifier,MsgStack_t *stack)  {
    Result_t rv = SUCCESS;

    UNS16_t    size  = stack->pMsg[3].size;           // Size of data
    UNS8_t     *temp = stack->pMsg[3].pData;

    while (size != 0  && rv == SUCCESS)  {
        if ((groundPort+identifier)->state == CONNECTED)  {
            if (size > (groundPort+identifier)->size)  {
                stack->pMsg[3].size = (groundPort+identifier)->size;
                size -= (groundPort+identifier)->size;
            }
        }
        else  {
            rv = INVALID_STATE;
        }
    }
    return ( rv );
}

```

☒ Corrected: STRING(XDATA) does not place the strings into xdata ROM

```

#pragma O2 STRING (XDATA) XCROM

#include <stdio.h>

void main (void)  {
    printf ("Hello");           // string goes to ?XC?... segment (OK), but
                                // segment has wrong memory class
                                // should have XCONST!
}

```

☒ Corrected: Philips 80C51MX generates internal error on esfr = data_var assignment

```

#include <REG51M.H>
char sss;

void main (void)  {
    sss = S1BUF;
    S1BUF = 5;
    S1BUF = ACC;
    S1BUF = B;
    DPL = S1BUF;
    S1BUF = sss;
    S1BUF |= B;
}

```

☒ Corrected: Incorrect code (MOV C,R7) generated

```

#pragma LARGE
char level, choice, n;
void (*const pdata ff[5])(void);
void main (void)  {
    ff [(!level) ? 5 : choice] ();
}

```

☑ Corrected: Internal error: label not found given

```
#pragma LARGE

void Func2 (char * pucMessage) reentrant;
char Func1 (void) {
    char * pucMessage;
    Func2 (pucMessage);

    return(1);
}

void Func2 (char * pucMessage) reentrant {
    char * pucFoundTLV;
    *pucMessage *= 1;

    Func2 (pucFoundTLV);
}
```

☑ Corrected: Internal Error (label not found)

☑ Corrected: _at_ gives error

```
int code notOk _at_ (0xC000 - 4); // gives invalid at specifier
int c1;
void main (void) {
    c1 = notOk;
    while (1);
}
```

☑ Corrected: Constant '0xFFFFFFFFu' is truncated to unsigned int

```
int i;
void main (void) {
    if((( 0xFFFFFFFFlu & 0x00FF0000l ) >> 16 ) == 0x000000FFl) { // OK.
        i = 0;
    }
    if((( 0xFFFFFFFFu & 0x00FF0000u ) >> 16 ) == 0x000000FFu) { // wrong.
        i = 2;
    }
}
```

☑ Improvement: 'NAME' directive added**☑ Corrected: Philips 80C51MX: Fixup change: F2_dir to F2_low for 'idir'**

```
#pragma MXP

int test (void) {
    char xdata * idata xp;

    xp++;
    xp = 0;
}
```

☑ Corrected: Pointer sub of 2 far pointers: cast to 'int' removed

```
#pragma VARBANKING 02
struct f;

struct f {
    struct f far * next;
    char x[10];
};

struct f far *c;
struct f far *cur;

char far *c1;
char far *c2;

unsigned long l;

void main (void) {
    l = (unsigned long) (c1 - c2);
    l = (unsigned long) ((unsigned char far*)c - (unsigned char far*)cur);
}
```

☑ Corrected: OMF2: no debug info for sfr/sbit is created

```
sfr P1 = 0x90;
sbit bit1 = P1^4;
sbit bit2 = 0xA8 ^ 6;
```

```
void main(void) {
    bit test;

    P1 = 0x55;
    while(1) {
        test = ~test;
        bit1 = test;
    }
}
```

☑ Corrected: External index in lcall/ljmp fixup wrong for 'extern alien f()'

Sample: C51 Tests\Alien.C1 code db

☑ Corrected: Internal Error

```
extern unsigned char const far * const far puc_Free_List_Start[0x03];
unsigned int ui_MM_Read_3_Fold (unsigned int const far * pucValue);
unsigned char const far * puc_MM_Find_End (unsigned char const far * pucBlock);

bit b_MM_Insert_Block (unsigned char const far * pucBlock) {
    unsigned char const far * pucLocalBlock;
    {
        pucLocalBlock = puc_MM_Find_End (((void *) (0xC00000L |
            (unsigned int) (void *) ui_MM_Read_3_Fold ((unsigned int*) puc_Free_List_Start))));
    }
    return 0;
}
```

☑ Corrected: String too long error

```
struct TICON {
    unsigned char PosX;
    unsigned char PosY;
    unsigned char SizeX;
    unsigned char SizeY;
    unsigned char *PixelData;
```



```
"\0x00\0x00\0x08\0x00\0x00\0x08\0x00\0x00\0x00\0x00\0x08\0x00\0x00\0x08\0x00\0x00"\
"Peter" };
```

☒ **Corrected: General Protection Error (GPF) on large test file**

Testfile: C51TEST\V6BUGS\E233U.C

☒ **Corrected: OPTIMIZE(8): possible wrong call when entry code is reduced more than once**

Testfile: C51TEST\V6BUGS\E236U.C

☒ **Corrected: Duplicate sbit symbols (OMF2 - PUBDEF, PUBSFR)**

sbit 'FinTimer1' for example is duplicate
Testfile: C51TEST\V6BUGS\E233U.C1

☒ **Corrected: Internal error: undefined label**

c51 E405.C1 SM DEBUG CODE OE

☒ **Corrected: PUBSFR info shows unused sbit symbols**

```
#include <math.h>
#include <stdio.h>
#include <reg517.h>
#include <stdlib.h>

//sfr  S0CON  = 0x98;
//sbit  TB80   = S0CON ^ 4;  // 0x9B;

float pow (float x, float y) {
//unsigned int i;
  int i;  // 23.4.2001

  TB80 = 1;
  if (x == 0.0) {
    if (y > 0) return (0.0);
lab:
    *((unsigned long *) &y) = 0xffffffff;
    return (y);
  }
  i = 0;
  if (x < 0.0) {
    i = y;
    if (y-i != 0.0) goto lab;
    x = -x;
  }
  y = exp (log (x) * y);
  if (i & 1) y = -y;
  return (y);
}
```

Modifications and Corrections in C51 Compiler from Version 7.01 to Version 7.02.

Part 2: C51 Version 7.01

☒ Corrected: Browse-Info read/write/addr may be incorrect

```
char szB [10];
unsigned char ndx;

extern void TestB (char *p); // {
// p;
//}

void main (void) {
    szB [ndx] = ndx;      // should be WRITE
    szB [ndx - 1] = 0;    // ditto
    TestB (szB);          // should be ADDROF
    TestB (&szB [ndx]);  // should be ADDROF
}
```

☒ Improvement: `_at_`: Enhancement, constant expression support

```
#pragma MXP

#define CODE_END_ADR 0xc04000L
enum b1 {
    FIRST = 1,
    SECOND,
    THIRD
};

long far ulR _at_ CODE_END_ADR - SECOND;

void main (void) {
    ulR--;
}
```

☒ Corrected: Undefined tag error given

```
typedef struct a b;
//struct a {
//  int x,y,z;
//};
const b* p;          // gives error: undefined tag
```

☒ Corrected: Syntax error given

```
char XXX;
char CCC;

void main (void) {
    XXX ?
        CCC = 8
    :
        CCC = 5;    // error here given
}
```

☑ Improvement: Warning text may cause misunderstandings

```
#define VAL 10
int i;

void test1 (void) {
    i = VAL;
}

//#undef VAL
#define VAL 20    // warning: macro redefined given

void test2 (void) {
    i = VAL;      // but still old value used
}
// New text for Warning 317: attempt to redefine macro '<name>'
```

☑ Corrected: Invalid mspace Warning given

```
#pragma MXP

#pragma SAVE
#pragma REGPARMS
extern void *memset (void *s, char val, int n);
#pragma RESTORE

#define E2PROM    const far
#define EEPROM_START_PHYS 0xC00000L           // physical start of EEPROM
#define EEPROM_START_ADR  EEPROM_START_PHYS + 0x20 // start of file system
#define EEPROM_END_ADR    0xC0FFFFL
#define IM_ERASE_EEPROM_DEFAULT  0xFF

void main (void) {
    memset((unsigned char const far*)EEPROM_START_ADR, IM_ERASE_EEPROM_DEFAULT,
           EEPROM_END_ADR-EEPROM_START_ADR+1);
    memset((unsigned char far*)EEPROM_START_ADR, IM_ERASE_EEPROM_DEFAULT,
           EEPROM_END_ADR-EEPROM_START_ADR+1);
}
```

☑ Corrected: Local bit: 'data' should be ignored

```
// memory type 'data' should be ignored on 'bit' variables
#pragma LARGE

sfr SBUF = 0x99;
sbit RB8 = 0x9A;

void EIB_SerialInterrupt() interrupt 2 using 1 {
    unsigned char data nChar;           // Holds the received char
    bit          data bParity;         // 'data' causes wrong SI

    nChar = SBUF;
    bParity = RB8;
}
```

☑ Improvement: Warning added

```
bit func (void);
int x;
void main (void) {
    if (func) x = 1;           // should give a Warning
    if (x = 3) x = 2;         // ditto.
```

}

☑ Corrected: Function-pointer may be incorrect with OT(8) or higher

```
extern int sprintf (char *, const char *, ...);

extern char Output[100];
typedef union ValueU {
    float Vfloat;
    char Vchar;
} ValueT;

typedef struct FunctionPointersS {
    void(*Init)(void);
    void(*Run)(ValueT*);
} FunctionPointerT;

extern void Init2(void);
extern void Sub2(ValueT *);

extern FunctionPointerT FunctionPointer2;

FunctionPointerT /* code */      // also when table is in code
FunctionPointer2 = {
    Init2,      Sub2
};

void Init2(void) {
    ValueT Value2;
    Value2.Vfloat = 1.; Sub2(&Value2);
    Value2.Vfloat = 2.; Sub2(&Value2);
    Value2.Vfloat = 3.; Sub2(&Value2);
    Value2.Vfloat = 4.; Sub2(&Value2);
    Value2.Vfloat = 5.; Sub2(&Value2);
    Value2.Vfloat = 6.; Sub2(&Value2);
    Value2.Vfloat = 7.; Sub2(&Value2);
}

void Sub2 (ValueT * V2) {
    sprintf(Output, "%f", V2->Vfloat);
}
```

☑ Corrected: PUBDEF incorrect for 'far' spaced data

```
#pragma MXP

#define CODE_END_ADR 0xc04000L
#define XDATA_ADR    0x001234L
#define CODE_ADR     0x005678L

enum b1 {
    FIRST = 1,
    SECOND,
    THIRD
};

int far    ulR    _at_ CODE_END_ADR - SECOND;
int xdata ulX    _at_ XDATA_ADR;
int code  ulC    _at_ CODE_ADR;

void main (void) {
    ulR--;
    --ulX;
    ulR = ulC;
}
```

☑ Corrected: General Protection Error while initializing

member init 'x.p = (p1 - p2) + CONST' gives GPF

☑ Improvement: Different enumeration types/values: no warning given

```

#if 0
    #pragma warning "BullShit Warning"
    #pragma message "BullShit Message"
#endif

enum STATE{ON = 0, OFF, BULL=10 };

enum STATE e_CF (enum STATE x) {
    return (x);
}
enum STATE e_C2 (enum STATE x, enum STATE y) {
    x = y;
    return (x);
}

void main() {
    char err;
    enum STATE state, st2;

    state = e_C2 (st2, 44); // state);
    state = e_C2 (10, OFF);
    state = e_C2 (8, BULL);
    state = e_C2 (8, 20);

    state = e_CF (st2);
    state = e_CF (10);
    state = e_CF (8);

    st2    = BULL;
    err    = ON;           // OK
    st2    = ON + 2;
    state  = 0x35;         // Warum wird hier kein Fehler ausgegeben !!
    state  = BULL;
    state  = 0x0A;         // equivalent
    state  = 0x33;
    state  = st2;

    state  = (enum STATE)0x35;

    //state = (    STATE)0x35;    // C++ Mode !!!
}

```

☑ Improvement: #pragma message added

```

#pragma warning "user warning text"
#pragma message "user message text"

```

☑ Corrected: mSpace in typedef ignored for array and pointer types

```

#pragma 02

typedef unsigned char xdata      xJmpBuf[4];
typedef unsigned char pdata      P_UCHAR;
typedef unsigned char far * xdata P_PXUC;
typedef unsigned char idata      ID_UC8;

```

```

xJumpBuf xdata xB3;    // Ok.

#if 0 // these must generate errors:
  xJumpBuf idata xB1;    // negative-test-case
  P_PXUC  pdata p2;    // negative-test-case
  ID_UC8  data v2;    // negative-test-case
#endif

xJumpBuf      xB2;
P_UCHAR      puc1;
P_PXUC       p1;
ID_UC8       v1;

void main (void) {
  xB2[1] = v1;
  *p1 = v1;
}

```

☒ Corrected: Optimizer Problem with loop-rotation

```

//Problem: '<= 4' is translated as '!= 5' (strength reduction)
//          in a loop with decrement this causes an underflow and incorrect loop
//          termination.

extern void _nop_(void);
void main(void) {
  unsigned char i;

  for ( i=3 ; i <= 4 ; i-- ) {
    _nop_();
  }
}

```

☒ Improvement: Warning 'pointer truncation' added

```

// generic pointer
unsigned char * data ch_gen_ptr;

unsigned char data  data_var;
unsigned char pdata pdata_var;
unsigned char xdata xdata_var;
unsigned char code  code_var = 9;

unsigned char data result1;
unsigned int  data result2;

// hier sollte eigentlich in jeder Zeile eine Warning ausgegeben werden.
void main (void) {
  result1 = ch_gen_ptr;
  result1 = (unsigned char) ch_gen_ptr;
  result1 = &data_var;
  result1 = &pdata_var;
  result1 = &xdata_var;
  result1 = &code_var;

  result2 = ch_gen_ptr;
  result2 = &data_var;
  result2 = &pdata_var;
  result2 = &xdata_var;
  result2 = &code_var;
}

```

☑ Corrected: General Protection Fault with sample below

```

struct TstrGeraet {
    long lSegmentEndTemp[50];
};

struct TstrGeraet xdata strGeraet;

int code strZahlEingeben =
    (unsigned char*)&strGeraet.lSegmentEndTemp[10]) -
    (unsigned char*)&strGeraet);

```

☑ Corrected: Preprocessor: incorrect expr. calculation

```

#define VALOR_1      1
#define VALOR_2      36

#define PVN_WORD_OFFSET(x) (((x) - 1)/8)&~1

#if (((1) - 1)/8)&~1) - (((36) - 1)/8)&~1)
    #error bull
#endif

#if (PVN_WORD_OFFSET(VALOR_1)) - (PVN_WORD_OFFSET(VALOR_2))
    #error Un fallo
#endif

#if (PVN_WORD_OFFSET(VALOR_1)) != (PVN_WORD_OFFSET(VALOR_2))
    #error Otro fallo
#endif

int main()
{
    int v1 = PVN_WORD_OFFSET(VALOR_1);
    int v2 = PVN_WORD_OFFSET(VALOR_2);

    return v1+v2;
}

```

☑ Corrected: Typedef'd mSpace ignored for function-locals

```

typedef unsigned char uchar;
typedef struct {
    uchar ucZahl1;
    uchar ucZahl2;
} mS;
typedef mS xdata xStruct;
//xStruct stStruktur;
void vTestfunk1() small {
    xStruct stStruktur;          // uses idata, should use xdata
    stStruktur.ucZahl1 = 0x89;
    stStruktur.ucZahl2 = 0xA1;
}

```

☑ Corrected: Incorrect code

```

#pragma LARGE DB OE CODE
#pragma OT(SIZE)

#define UC unsigned char
typedef struct mapping_para{
    unsigned int index;
    unsigned char num_objects;
}t_map_para;

```

```

t_map_para USR_as_RxPDOMappingObj[5] = { {0x1600,3},
                                           {0x1601,4},
                                           {0x1602,4},
                                           {0x1603,4},
                                           {0x1604,4}};

unsigned char b_Test = 2;

void TST_Fkt (unsigned int index) {
    t_map_para * idata p_pdo_para;

    p_pdo_para = &USR_as_RxPDOMappingObj[0];
    if ((*((UC *) &(index))) == (*((UC *) &(p_pdo_para->index)))) {
        b_Test = 7;
    }
    else {
        b_Test = 3;
    }
}

void main (void){
    TST_Fkt (0x1600);
    while (1);
}

```

☒ Corrected: Incorrect code with 'MODE2'

```

#pragma MODE2
#define UBYTE unsigned char
#define BOOL bit // booléen
#define TRUE (BOOL)1 // booleen TRUE
#define FALSE (BOOL)0 // booleen FALSE

#define OFFSET 4

data UBYTE AR0 _at_ 0;
UBYTE xdata * data PtrCourant _at_ 0x20;
UBYTE xdata ObjetRecherche[4] _at_ 0x0084;

BOOL func1(void) { return TRUE;}

BOOL func2(void) {
    while (func1()) {
        if ( (ObjetRecherche[0] == *(PtrCourant+OFFSET+0))
            &&(ObjetRecherche[1] == *(PtrCourant+OFFSET+1))
            &&(ObjetRecherche[2] == *(PtrCourant+OFFSET+2)) )
            return TRUE; // SETB C is missing on OT(4) or greater
    }
    return FALSE;
}

void main(void) {
    if (func2()) AR0 = 1;
    else AR0 = 2;
}

```

☒ Corrected: Memory Typed -> long cast might be wrong

```

#define uchar unsigned char
#define uint unsigned int
#define ulong unsigned long
#define IDATA idata
#define XDATA xdata

ulong ulTest(uchar XDATA *pucTest1, uchar XDATA *pucTest2);

```

```

void main() {
    ulong ulTemp;
    ulTemp = ulTest(0x007733, 0x006655);
    return;
}
ulong ulTest(uchar XDATA *pucTest1, uchar XDATA *pucTest2) {
    ulong ulResult;
    uint uiResult;

    ulResult = (ulong)pucTest1 - (ulong)pucTest2;    //Ergebnis: 0x00007733 ->
Schmarren
    uiResult = (uint)pucTest1 - (uint)pucTest2;      //Ergebnis: 0x10DE -> o.k.

    return ulResult;
}

```

☒ Corrected: Incorrect code

```

unsigned char test (unsigned char code **pif) {
    if( *pif != ((void *) 0))
        return(1);
    else
        return(0);
}

```

☒ Corrected: Internal Error

```

unsigned char far large_array[0xC000];
#define LENGTH_NVRAM_PAGE_SIZE    0x40
#define ADDRESS_NVRAM              ((unsigned char xdata *)0x8400)
char value;

void main(void) {
    unsigned char *blockAddress;
    blockAddress = (unsigned char xdata *)0x8204;
    if (((unsigned long)(unsigned char *) (ADDRESS_NVRAM+
        (*((unsigned short *)blockAddress)))) & 0xC0) == 0)
        return;
    value = 1;
}

```

☒ Corrected: Parse Stack Overflow Error Message

☒ Improvement: volatile attribute can be used to suppress common code

```

#pragma OT(9)

// func should be not in common code sequences. The volatile attribute
// can be used to avoid that this function is put into common code.

extern volatile void func (int z);

int x;

void main (void) {
    x = 0;
    func (1);
    func (2);
    x = 0x222;
    x = 0;
}

```

```

func (1);
func (2);
while (1);
}

```

☒ Corrected: strtol, strtoul accepts ':' as valid digit

☒ Corrected: Incorrect Expression Folding

```

typedef unsigned int word;
typedef unsigned char byte;
typedef byte _3byte[3];

void main() {
    word    xdata wVarDummy = 4;
    _3byte  xdata awVar[10];
    byte          byVar1 = 1;
    byte          byVar2 = 2;

    awVar[byVar1 + 1 - byVar2][0] = 0xEE;    //Index sollte [0][0] sein.
    for(;;);

// another example

void main (void) {

    typedef struct {
        unsigned long a;
        unsigned char b;
    } A_type;

    data unsigned char i,j,k;
    xdata A_type Test[10];

    j=2;
    for(i=0;i<10;i++)
        k=Test[4+i-j].b;    // if i<j this line generates a wrong address
    for(i=0;i<10;i++)
        k=Test[4-j+i].b;    // workaround
}

```

☒ Corrected: Problem with compound assignments += / -= on far pointer operations

```

#pragma ROM(D512K) O2

char          far *apcXAry[ 2 ];    // in XDATA
char          far * far apcHAry[ 2 ]; // in HDATA
char          far * far pcHPtr;    // in HDATA
char          far * far pcH64K;    // in HDATA
int           far i;

void main (void) {
    apcHAry[0] = 0x10fff7;
    i = 0;
    apcHAry[ i ] += 0x10L;    // result wrong
    apcHAry[ i ] = apcHAry[ i ] + 0x10L; // result OK

    pcH64K = 0x10fff7;
    pcH64K += 0x10L;    // result wrong
    pcH64K = pcH64K + 0x10L;    // result OK
}

```

```
}
```

☒ Corrected: if (p) is not identical with if(p == NULL)

```
#pragma O2
char far *p;

void main (void) {
    if (p)          p++;
    if (p == NULL)  p++;
}
```

☒ Corrected: General Protection Fault with long invocation lines (via @cmd_file)