```
// UART functions for SDCC by Chin-Shiuh Shieh on 2008-06-15
// uart_init() should be called before using UART
// No syntax error handling
// Comment un-used functions to reduce code size

// UART function prototypes
void uart_init(void);   // Set UART Baud Rate to 9600 bps
void uart_putc(char c); // Put Character to UART
char uart_getc(void);   // Get Character from UART
char uart_getce(void);  // Get Character from UART with Echo, Require uart_putc()
void uart_puts(char *s);// Put String to UART, Require uart_putc()
void uart_gets(char *s);// Get String from UART, Require uart_getce()
void i2s(int i,char *s);// Convert Integer to String
int s2i(char *s);       // Convert String to Integer
void uart_puti(int i);  // Put Integer to UART, Require i2s(),uart_puts()
int uart_geti();        // Get Integer from UART, Require uart_gets(),s2i()

void uart_init(void)    // Set UART Baud Rate to 9600 bps
{
PCON&=0x7F;             // Clear SMOD of PCON, No Double Baud Rate
TMOD&=0xAF;TMOD|=0x20;  // Set Timer1 to Mode 2 (8-bit auto reload) for Baud Rate
Generation
TH1=0xFD;               // Set Baud Rate to 9600 bps for 11.0592M Hz
SM0=0;SM1=1;            // Set UART to Mode 1 (8-bit UART)
REN=1;                  // Set REN of SCON to Enable UART Receive
TR1=1;                  // Set TR1 of TCON to Start Timer1
TI=0;RI=0;              // Set TI/RI of SCON to Get Ready to Send/Receive
}
void uart_putc(char c)  // Put Character to UART
{
SBUF=c;
while(TI==0);TI=0;
}
char uart_getc(void)    // Get Character from UART
{
while(RI==0);RI=0;
return SBUF;
}
char uart_getce(void)   // Get Character from UART with Echo, Require uart_putc()
{
while(RI==0);RI=0;
uart_putc(SBUF);
return SBUF;
}
void uart_puts(char *s) // Put String to UART, Require uart_putc()
{
while(*s!=0){uart_putc(*s);s++;}
}
void uart_gets(char *s) // Get String from UART, Require uart_getce()
{
while(((*s)=uart_getce())!=13)s++;
*s=0;
}
void i2s(int i,char *s) // Convert Integer to String
{
char sign;short len;char *p;
sign='+';len=0;p=s;
if(i<0){sign='-';i=-i;}
do{*s=(i%10)+'0';s++;len++;i/=10;}while(i!=0);
if(sign=='-'){*s='-';s++;len++;}
```

```
for(i=0;i<len/2;i++){p[len]=p[i];p[i]=p[len-1-i];p[len-1-i]=p[len];}
p[len]=0;
}
int s2i(char *s)        // Convert String to Integer
{
int i=0;char sign='+';
if(*s=='-'){sign='-';s++;}
while(*s!=0){i=i*10+(*s-'0');s++;}
if(sign=='-')i=-i;
return i;
}
void uart_puti(int i)   // Put Integer to UART, Require i2s(),uart_puts()
{
char s[7];
i2s(i,s);uart_puts(s);
}
int uart_geti()         // Get Integer from UART, Require uart_gets(),s2i()
{
char s[7];
uart_gets(s);
return s2i(s);
}
```