/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Vision lwip\_v1.4.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

◆LWIP默认包含opts.h作为系统默认配置，不过通过添加lwipopts.h文件并包含在opts.h头文件之前就可以对lwip进行用户裁剪。

◆本工程未使用操作系统，采用的是lwip RAW接口。

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*lwipopts.h文件配置\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**#define** **NO\_SYS**                          1/\*1--未使用操作系统\*/

**#define** **NO\_SYS\_NO\_TIMERS**        1

**#define** **LWIP\_SOCKET**                     0/\*0--*Disable Socket API*\*/

**#define** **LWIP\_NETCONN**                   0/\*0--*Disable Netconn API*\*/

**#define** **LWIP\_RAW**                         1/\*1--采用RAW编程接口\*/

*/\* ---------- Memory options ---------- \*/*

*/\* MEM\_ALIGNMENT: should be set to the alignment of the CPU for which*

*lwIP is compiled. 4 byte alignment -> define MEM\_ALIGNMENT to 4, 2*

*byte alignment -> define MEM\_ALIGNMENT to 2. \*/*

**#define MEM\_ALIGNMENT 4**

*/\* MEM\_SIZE: the size of the heap memory. If the application will send*

*a lot of data that needs to be copied, this should be set high. \*/*

**#define MEM\_SIZE (10\*1024)** *//*用于mem分配使用(比如分配发送/接收数据缓存区)

*/\* MEMP\_NUM\_PBUF: the number of memp struct pbufs. If the application*

*sends a lot of data out of ROM (or other static memory), this*

*should be set high. \*/*

*//#define MEMP\_NUM\_PBUF 100*

*/\* MEMP\_NUM\_UDP\_PCB: the number of UDP protocol control blocks. One*

*per active UDP "connection". \*/*

**#define** **MEMP\_NUM\_UDP\_PCB** 2

*/\* MEMP\_NUM\_TCP\_PCB: the number of simulatenously active TCP*

*connections. \*/*

**#define MEMP\_NUM\_TCP\_PCB 6**

*/\* MEMP\_NUM\_TCP\_PCB\_LISTEN: the number of listening TCP*

*connections. \*/*

**#define** **MEMP\_NUM\_TCP\_PCB\_LISTEN** 1

*/\* MEMP\_NUM\_TCP\_SEG: the number of simultaneously queued TCP*

*segments. \*/*

**#define MEMP\_NUM\_TCP\_SEG 40**

*/\* MEMP\_NUM\_SYS\_TIMEOUT: the number of simulateously active*

*timeouts. \*/*

**#define MEMP\_NUM\_SYS\_TIMEOUT 10**

*/\* ---------- Pbuf options ---------- \*/*

*/\* PBUF\_POOL\_SIZE: the number of buffers in the pbuf pool. \*/*

**#define PBUF\_POOL\_SIZE 8**//用于memp分配使用，pbuf 池buffer个数

*/\* PBUF\_POOL\_BUFSIZE: the size of each pbuf in the pbuf pool. \*/*

**#define PBUF\_POOL\_BUFSIZE (1500 - 40)***/\* TCP\_MSS = (Ethernet MTU - IP header size - TCP header size) \*/*

**#define** **LWIP\_DNS** 1

*/\* ---------- TCP options ---------- \*/*

**#define LWIP\_TCP 1**

**#define TCP\_TTL 255**

*/\* Controls if TCP should queue segments that arrive out of*

*order. Define to 0 if your device is low on memory. \*/*

**#define TCP\_QUEUE\_OOSEQ 1**

*/\* TCP Maximum segment size. \*/*

**#define TCP\_MSS (1500 - 40)***/\* TCP\_MSS = (Ethernet MTU - IP header size - TCP header size) \*/*

*/\* TCP sender buffer space (bytes). \*/*

**#define TCP\_SND\_BUF (4\*TCP\_MSS)**

*/\* TCP\_SND\_QUEUELEN: TCP sender buffer space (pbufs). This must be at least*

*as much as (2 \* TCP\_SND\_BUF/TCP\_MSS) for things to work. \*/*

**#define TCP\_SND\_QUEUELEN (4\* TCP\_SND\_BUF/TCP\_MSS)**

*/\* TCP receive window. \*/*

**#define TCP\_WND (2\*TCP\_MSS)**

*/\* ---------- ICMP options ---------- \*/*

**#define LWIP\_ICMP 1**

**#define LWIP\_ARP 1** *//值1使能*

*/\* ---------- DHCP options ---------- \*/*

*/\* Define LWIP\_DHCP to 1 if you want DHCP configuration of*

*interfaces. DHCP is not implemented in lwIP 0.5.1, however, so*

*turning this on does currently not work. \*/*

**#define LWIP\_DHCP 1**

*/\* ---------- UDP options ---------- \*/*

**#define** **LWIP\_UDP** 1

**#define** **UDP\_TTL** 255

*/\* ---------- Statistics options ---------- \*/*

**#define LWIP\_STATS 0**

**#define LWIP\_PROVIDE\_ERRNO 1**

*/\* ---------- link callback options ---------- \*/*

*/\* LWIP\_NETIF\_LINK\_CALLBACK==1: Support a callback function from an interface*

*\* whenever the link changes (i.e., link down)*

*\*/*

**#define LWIP\_NETIF\_LINK\_CALLBACK 1**

*/\*---------- Checksum options ----------\*/*  
*/\* The STM32F4x7 allows computing and verifying the IP, UDP, TCP and ICMP checksums by hardware:  
 - To use this feature let the following define uncommented.  
 - To disable it and process by CPU comment the  the checksum.\*/*  
**#define** **CHECKSUM\_BY\_HARDWARE**   
**#ifdef** **CHECKSUM\_BY\_HARDWARE**  
  */\* CHECKSUM\_GEN\_IP==0: Generate checksums by hardware for outgoing IP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_IP**                 0  
  */\* CHECKSUM\_GEN\_UDP==0: Generate checksums by hardware for outgoing UDP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_UDP**                0  
  */\* CHECKSUM\_GEN\_TCP==0: Generate checksums by hardware for outgoing TCP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_TCP**                0   
  */\* CHECKSUM\_CHECK\_IP==0: Check checksums by hardware for incoming IP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_IP**               0  
  */\* CHECKSUM\_CHECK\_UDP==0: Check checksums by hardware for incoming UDP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_UDP**              0  
  */\* CHECKSUM\_CHECK\_TCP==0: Check checksums by hardware for incoming TCP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_TCP**              0  
  */\* CHECKSUM\_CHECK\_ICMP==0: Check checksums by hardware for incoming ICMP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_ICMP**               0  
**#else**  
  */\* CHECKSUM\_GEN\_IP==1: Generate checksums in software for outgoing IP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_IP**                 1  
  */\* CHECKSUM\_GEN\_UDP==1: Generate checksums in software for outgoing UDP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_UDP**                1  
  */\* CHECKSUM\_GEN\_TCP==1: Generate checksums in software for outgoing TCP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_TCP**                1  
  */\* CHECKSUM\_CHECK\_IP==1: Check checksums in software for incoming IP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_IP**               1  
  */\* CHECKSUM\_CHECK\_UDP==1: Check checksums in software for incoming UDP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_UDP**              1  
  */\* CHECKSUM\_CHECK\_TCP==1: Check checksums in software for incoming TCP packets.\*/*  
  **#define** **CHECKSUM\_CHECK\_TCP**              1  
  */\* CHECKSUM\_CHECK\_ICMP==1: Check checksums by hardware for incoming ICMP packets.\*/*  
  **#define** **CHECKSUM\_GEN\_ICMP**               1  
**#endif**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*netconf.h文件配置\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

*/\* 远端IP地址和端口 ,设置ip = 0.0.0.0时，表示接收所有目的ip地址指向本机的远端ip数据\*/ //实际使用dns域名解析目的ip，未使用*  
**#define** **DEST\_IP\_ADDR0**               172  
**#define** **DEST\_IP\_ADDR1**               16  
**#define** **DEST\_IP\_ADDR2**                 1  
**#define** **DEST\_IP\_ADDR3**               201  
  
  
**#define** **UDP\_DEST\_PORT**              50000//  
  
**#define** **UDP\_SRC\_PORT**               50000//端口动态分配时不使用

*/\*   
MAC地址：网卡地址 ，设备唯一标识符，不同的设备配置不同的mac地址；  
前3字节由芯片厂商定义，后3字节由用户自定义  
\*/ //实际使用蓝牙mac地址，未使用*  
**#define** **MAC\_ADDR0**                     2  
**#define** **MAC\_ADDR1**                     0  
**#define** **MAC\_ADDR2**                     0  
**#define** **MAC\_ADDR3**                     0  
**#define** **MAC\_ADDR4**                     0  
**#define** **MAC\_ADDR5**                     1  
  
*/\*静态IP地址 \*///DHCP使能时不使用*  
**#define** **IP\_ADDR0**                    172  
**#define** **IP\_ADDR1**                    16  
**#define** **IP\_ADDR2**                      1  
**#define** **IP\_ADDR3**                    200  
  
*/\* 子网掩码 \*///DHCP使能时不使用*  
**#define** **NETMASK\_ADDR0**               255  
**#define** **NETMASK\_ADDR1**               255  
**#define** **NETMASK\_ADDR2**               255  
**#define** **NETMASK\_ADDR3**                 0  
  
*/\* 网关 \*///DHCP使能时不使用*  
**#define** **GW\_ADDR0**                    172  
**#define** **GW\_ADDR1**                    16  
**#define** **GW\_ADDR2**                      1  
**#define** **GW\_ADDR3**                      1  
  
*/\* 检测PHY链路状态的实际间隔(单位：ms) \*/*  
**#ifndef** **LINK\_TIMER\_INTERVAL**  
**#define** **LINK\_TIMER\_INTERVAL**        1000  
**#endif**  
  
*/\* MII and RMII mode selection \*\*\*\*\*\*\*\*\*\*\*/*  
**#define** **RMII\_MODE**    
*//#define MII\_MODE*  
*/\* 在MII模式时，使能MCO引脚输出25MHz脉冲 \*/*  
**#ifdef**  **MII\_MODE**  
 **#define** **PHY\_CLOCK\_MCO**  
**#endif**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*初始化接口\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

   ETH\_BSP\_Config();    
    LwIP\_Init();

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/