

Socket APIs for ioPlatform

Version 0.9beta



© 2011 WIZnet Co., Inc. All Rights Reserved.

For more information, visit our website at <http://www.wiznet.co.kr>

Table of Contents

1. Introduction	4
2. Socket APIs List	4
3. Detail Information	5
3.1 Structure	5
3.1.1 wiz_NetInfo	5
3.2 Common	6
3.2.1 wizInit	6
3.2.2 wizSWReset	6
3.2.3 wizMemInit	7
3.2.4 SetNetInfo	8
3.2.5 GetNetInfo	8
3.2.6 SetSocketOption	9
3.3 TCP	10
3.3.1 TCPServerOpen	10
3.3.2 TCPClientOpen	10
3.3.3 TCPSend	11
3.3.4 TCPReSend	12
3.3.5 TCPRecv	13
3.3.6 TCPClose	14
3.3.7 GetTCPsocketStatus	15
3.4 UDP	16
3.4.1 UDPOpen	16
3.4.2 UDPSend	16
3.4.3 UDPRecv	17
3.4.4 UDPClose	19

3.4.5	GetUDPSocketStatus	19
4.	APPENDIX.....	21
4.1	Return Values	21

1. Introduction

2. Socket APIs List

Type	Name	Description
<u>Structure</u>	<u>wiz_NetInfo</u>	네트워크의 정보를 저장하기 위해 사용한다.
<u>Common</u>	<u>wizInit</u>	W5200을 초기화 한다.
	<u>wizSWReset</u>	W5200을 소프트웨어 리셋한다.
	<u>wizMemInit</u>	소켓의 TX, RX Buffer 크기를 할당한다.
	<u>SetNetInfo</u>	MAC, IP, Subnet, Gateway 등을 설정한다.
	<u>GetNetInfo</u>	MAC, IP, Subnet, Gateway 등의 값을 읽는다.
	<u>SetSocketOption</u>	소켓의 옵션을 설정한다.
<u>TCP</u>	<u>TCPServerOpen</u>	TCP Server 소켓을 열고 클라이언트의 접속을 기다린다.
	<u>TCPClientOpen</u>	TCP Client 소켓을 열고 서버에 접속을 시도한다.
	<u>TCPSend</u>	TCP 패킷을 전송한다.
	<u>TCPSend</u>	TCP 패킷을 재전송한다.
	<u>TCPrecv</u>	TCP 패킷을 수신한다.
	<u>TCPClose</u>	TCP 소켓을 닫는다.
	<u>GetTCPStatus</u>	TCP 소켓의 상태를 읽는다.
<u>UDP</u>	<u>UDPOpen</u>	UDP 소켓을 연다.
	<u>UDPSend</u>	UDP 데이터그램을 전송한다.
	<u>UDPrecv</u>	UDP 데이터그램을 수신한다.
	<u>UDPClose</u>	UDP 소켓을 닫는다.
	<u>GetUDPStatus</u>	UDP 소켓의 상태를 읽는다.

3. Detail Information

3.1 Structure

3.1.1 wiz_NetInfo

wiz_NetInfo 구조체는 네트워크의 정보를 저장하기 위해 사용한다.

```
typedef struct _wiz_NetInfo
{
    uint8 Mac[6];
    uint8 IP[4];
    uint8 Subnet[4];
    uint8 Gateway[4];
    uint8 DNSServerIP[4];
    uint8 DHCPEnable;
} wiz_NetInfo;
```

Member Variables

Mac

MAC Address를 위한 변수

IP

IP Address를 위한 변수

Subnet

Subnet Mask를 위한 변수

Gateway

Gateway IP Address를 위한 변수

DNSServerIP

[Optional] DNS Server IP Address를 위한 변수로 DNS 관련 동작을 위해 사용

DHCPEnable

[Optional] DHCP 사용여부에 대한 변수로 DHCP 관련 동작을 위해 사용

3.2 Common

3.2.1 wizInit

wizInit 함수는 W5200을 초기화한다.

```
void wizInit(  
);
```

Parameters

없음

Remarks

wizInit 함수는 W5200사용전에 반드시 한번 호출하여야 한다. W5200을 초기화 하고 각 소켓의 TX, RX Buffer의 크기를 모두 2KB로 설정한다.

Return Values

없음

Error Codes

없음

See Also

[wizSWReset](#), [wizMemInit](#)

3.2.2 wizSWReset

wizSWReset 함수는 W5200을 소프트웨어 리셋 한다.

```
void TCPServerOpen(  
);
```

Parameters

없음

Remarks

소프트웨어 리셋을 통해 W5200의 내부 레지스터를 초기화 시킨다.

Return Values

없음

Error Codes

없음

See Also

[wizInit](#)

3.2.3 wizMemInit

wizMemInit 함수는 각 소켓의 TX, RX Buffer의 크기를 설정한다.

```
void wizMemInit(
    uint8 * tx_size,
    uint8 * rx_size
);
```

Parameters

tx_size

[입력] TX Buffer의 크기

rx_size

[입력] RX Buffer의 크기

Remarks

W5200은 8개의 하드웨어 소켓을 제공한다. 각 소켓의 TX, RX Buffer 크기는 8바이트 배열에 다음과 같이 KB(Kilo-Bytes)단위로 값을 넣고, 배열을 wizMemInit 함수에 인자로 넣어 설정한다.

```
tx_size[8] = {2, 2, 2, 2, 2, 2, 2, 2};
rx_size[8] = {2, 2, 2, 2, 2, 2, 2, 2};
wizMemInit(tx_size, rx_size);
```

wizInit 함수를 통해 이미 모든 소켓의 TX, RX Buffer 크기가 2KB로 설정되어 있기 때문에 소켓 별로 다른 크기의 TX, RX Buffer를 사용할 사용자만 이 함수를 사용하도록 한다.

Return Values

없음

Error Codes

없음

See Also

[wizInit](#)

3.2.4 SetNetInfo

SetNetInfo 함수는 W5200의 MAC, IP, Subnet, Gateway, DNS Server IP, DHCPEnable를 설정한다.

```
int8 TCPServerOpen(
    wiz_NetInfo *netinfo
);
```

Parameters

netinfo

[입력] MAC, IP, Subnet, DNS Server IP, DHCPEnable 값을 명시하는 구조체.

Remarks

Return Values

없음

Error Codes

없음

See Also

[wiz_NetInfo](#), [wizInit](#), [GetNetInfo](#)

3.2.5 GetNetInfo

GetNetInfo 함수는 소켓을 초기화하고 클라이언트의 접속을 기다리도록 한다.

```
void GetNetinfo(
    wiz_NetInfo *netinfo
);
```

Parameters

netinfo

[출력] MAC, IP, Subnet, DNS Server IP, DHCPEnable 값을 저장할 구조체

Remarks

Return Values

없음

Error Codes

없음

See Also

[wiz_NetInfo](#), [wizInit](#), [SetNetInfo](#)

3.2.6 SetSocketOption

SetSocketOption 함수는 소켓을 초기화하고 클라이언트의 접속을 기다리도록 한다.

```
void SetSocketOption(  
    uint8 option_type,  
    uint16 option_value  
);
```

Parameters

option_type

[입력] 설정하려는 옵션의 타입을 명시(0: Retry Duration, 1: Retry Count, 2: Interrupt Mask)

option_value

[입력] 설정할 옵션 값을 명시

Remarks**Return Values**

없음

Error Codes

없음

See Also

3.3 TCP

3.3.1 TCPServerOpen

TCPServerOpen 함수는 소켓을 TCP 모드로 초기화하고 클라이언트의 접속을 기다리도록 한다.

```
int8 TCPServerOpen(
    SOCKET s,
    uint16 port
);
```

Parameters

s

[입력] 초기화할 소켓의 번호를 명시한다.

port

[입력] 클라이언트의 접속을 기다릴 포트 번호를 명시한다.

Remarks

Return Values

에러가 발생하지 않는 경우 TCPServerOpen 함수는 SUCCESS를 반환 한다. 에러가 발생한 경우에는 FAIL을 리턴한다.

Error Codes

SUCCESS	성공적으로 소켓을 초기화 하였음
FAIL	소켓 초기화를 실패 하였음

See Also

[TCPSend](#), [TCPReSend](#), [TCPRecv](#), [TCPClose](#)

3.3.2 TCPClientOpen

TCPClientOpen 함수는 소켓을 TCP 모드로 초기화하고 서버로 접속을 시도 한다.

```
int8 TCPClientOpen(
    SOCKET s,
    uint16 port,
    uint8 * destip,
```

```
uint16 destport
);
```

Parameters

s

[입력] 초기화할 소켓의 번호를 명시한다.

port

[입력] 클라이언트의 포트 번호를 명시한다.

destip

[입력] 연결을 시도할 서버의 IP주소를 명시한다.

destport

[입력] 연결을 시도할 서버의 포트 번호를 명시한다.

Remarks

Return Values

에러가 발생하지 않는 경우 TCPClientOpen 함수는 SUCCESS를 반환 한다. 에러가 발생한 경우에는 FAIL을 리턴한다. SUCCESS는 서버로 접속 요청을 성공적으로 마쳤다는 의미이므로, 서버에 성공적으로 연결 되었다는 의미는 아니다.

Error Codes

SUCCESS	성공적으로 소켓을 초기화하고 서버로 접속 요청을 시도 하였음
FAIL	소켓 초기화를 실패 하였음

See Also

[TCPSend](#), [TCPReSend](#), [TCPRecv](#), [TCPClose](#)

3.3.3 TCPSend

TCPSend 함수는 접속되어 있는 상대방 소켓으로 데이터를 보내는 함수이다.

```
int16 TCPSend(
    SOCKET s,
    const uint8 * buf,
    uint16 len
```

```
);
```

Parameters

s

[입력] 접속되어 있는 소켓의 번호를 명시한다.

buf

[입력] 전송하려는 데이터를 가지고 있는 버퍼

len

[입력] 전송하려는 데이터의 길이

Remarks

데이터를 모두 전송 하지 못한 경우 TCPReSend 함수를 통해 나머지 데이터를 전송 할 수 있다.

Return Values

에러가 발생하지 않는 경우 TCPSend 함수는 전송한 데이터 길이를 반환 한다. 에러가 발생한 경우에는 아래 표와 같은 값을 리턴한다.

Error Codes

ERROR_CLOSED	소켓이 닫혀있음
ERROR_NOT_TCP_SOCKET	TCP 소켓이 아님
ERROR_FIN_WAIT	소켓 연결이 종료되었음
ERROR_NOT_ESTABLISHED	소켓이 연결되어 있지 않음

See Also

[TCPServerOpen](#), [TCPClientOpen](#), [TCPReSend](#), [TCPRecv](#), [TCPClose](#)

3.3.4 TCPReSend

TCPReSend 함수는 TCPSend 함수를 통해 데이터를 모두 전송 하지 못한 경우, 나머지 데이터를 접속된 소켓으로 보내는 함수이다.

```
int16 TCPReSend(
    SOCKET s
);
```

Parameters

s

[입력] 접속되어 있는 소켓의 번호를 명시한다.

Remarks

상대방이 데이터를 모두 전송 받지 못하는 경우 TCPReSend 함수를 다시 호출하여 나머지 데이터를 전송 할 수 있다. Window Full 상태에 도달하게 되면 ERROR_WINDOW_FULL 에러를 리턴하게 된다.

Return Values

에러가 발생하지 않는 경우 TCPReSend 함수는 전송한 데이터 길이를 반환 한다. 에러가 발생한 경우에는 아래 표와 같은 값을 리턴한다.

Error Codes

ERROR_CLOSED	소켓이 닫혀있음
ERROR_NOT_TCP_SOCKET	TCP 소켓이 아님
ERROR_FIN_WAIT	소켓 연결이 종료되었음
ERROR_NOT_ESTABLISHED	소켓이 연결되어 있지 않음
ERROR_WINDOW_FULL	Window Full로 인하여 데이터를 모두 전송 할 수 없음

See Also

[TCPServerOpen](#), [TCPClientOpen](#), [TCPSend](#), [TCPRecv](#), [TCPClose](#)

3.3.5 TCPRecv

TCPRecv 함수는 접속되어 있는 소켓으로부터 들어오는 데이터를 수신하는 함수이다.

```
int16 TCPRecv(
    SOCKET s,
    UInt8 * buf
    uint16 len
);
```

Parameters

s

[입력] 초기화할 소켓의 번호를 명시한다.

buf

[출력] 수신한 데이터를 저장할 버퍼

len

[입력] 버퍼의 길이

Remarks

Return Values

에러가 발생하지 않는 경우 TCPRecv 함수는 수신한 데이터 길이를 반환 한다. 에러가 발생한 경우에는 아래 표와 같은 값을 리턴한다.

Error Codes

ERROR_CLOSED	소켓이 닫혀있음
ERROR_NOT_TCP_SOCKET	TCP 소켓이 아님
ERROR_CLOSE_WAIT	소켓 연결이 종료되었음
ERROR_NOT_ESTABLISHED	소켓이 연결되어 있지 않음

See Also

[TCPServerOpen](#), [TCPClientOpen](#), [TCPSend](#), [TCPReSend](#), [TCPClose](#)

3.3.6 TCPClose

TCPClose 함수는 소켓을 닫는 함수이다.

```
int8 TCPClose(
    SOCKET s,
);
```

Parameters

s
[입력] 소켓의 번호를 명시한다.

Remarks

소켓이 연결되어 있다면 연결을 끊고 소켓을 닫는다.

Return Values

에러가 발생하지 않는 경우 TCPClose 함수는 SUCCESS를 반환 한다. 에러가 발생한 경우에는 FAIL을 리턴한다.

Error Codes

SUCCESS	성공적으로 소켓을 닫았음
---------	---------------

FAIL	소켓 종료를 실패 하였음
------	---------------

See Also

[TCPServerOpen](#), [TCPClientOpen](#), [TCPSend](#), [TCPReSend](#), [TCPRecv](#)

3.3.7 GetTCPSocketStatus

GetTCPSocketStatus 함수는 소켓을 TCP로 설정된 소켓의 상태를 알려준다.

```
int8 GetTCPSocketStatus(
    SOCKET s,
);
```

Parameters

s

[입력] 소켓의 번호를 명시한다.

Remarks

Return Values

소켓의 상태에 따라 아래 표와 같은 값을 리턴한다.

SOCK_CLOSED	소켓이 닫혀있음
SOCK_INIT	소켓이 TCP 모드로 초기화 되었음
SOCK_LISTEN	TCP Server로 Listen 상태
SOCK_SYNSENT	Sync 패킷을 전송하였음
SOCK_SYNRECV	Sync 패킷을 수신하였음
SOCK_ESTABLISHED	소켓이 연결 되었음
SOCK_FIN_WAIT	소켓이 닫히고 있는 중 나타나는 상태
SOCK_CLOSING	
SOCK_TIME_WAIT	
SOCK_CLOSE_WAIT	FIN 패킷을 수신하였음
SOCK_LAST_ACK	FIN 패킷에 대한 ACK를 기다리고 있음

Error Codes

없음

See Also

[TCPServerOpen](#), [TCPClientOpen](#), [TCPSend](#), [TCPReSend](#), [TCPRecv](#), [TCPClose](#)

3.4 UDP

3.4.1 UDPOpen

UDPOpen 함수는 소켓을 UDP 모드로 초기화 한다.

```
int8 UDPOpen(
    SOCKET s,
    uint16 port
);
```

Parameters

s

[입력] 초기화할 소켓의 번호를 명시한다.

port

[입력] UDP 포트 번호를 명시한다.

Remarks

Return Values

에러가 발생하지 않는 경우 UDPOpen 함수는 SUCCESS를 반환 한다. 에러가 발생한 경우에는 FAIL을 리턴한다.

Error Codes

SUCCESS	성공적으로 소켓을 초기화 하였음
FAIL	소켓 초기화를 실패 하였음

See Also

[UDPSend](#), [UDPRecv](#), [UDPClose](#)

3.4.2 UDPSend

UDPSend 함수는 소켓으로 데이터그램을 보내는 함수이다.

```
int16 UDPSend(
    SOCKET s,
    const uint8 * buf,
```



```
uint16 len,
uint8 * addr,
uint16 port
);
```

Parameters

s

[입력] 접속되어 있는 소켓의 번호를 명시한다.

buf

[입력] 전송하려는 데이터를 가지고 있는 버퍼

len

[입력] 전송하려는 데이터의 길이

addr

[입력] 목적지 IP 주소를 명시

port

[입력] 목적지 포트 번호를 명시

Remarks

Return Values

에러가 발생하지 않는 경우 UDPSend 함수는 전송한 데이터 길이를 반환 한다. 에러가 발생한 경우에는 아래 표와 같은 값을 리턴한다.

Error Codes

ERROR_CLOSED	소켓이 닫혀있음
ERROR_NOT_UDP_SOCKET	UDP 소켓이 아님
ERROR_TIME_OUT	Time Out으로 인하여 패킷이 전송되지 않음

See Also

[UDPOpen](#), [UDPRecv](#), [UDPClose](#)

3.4.3 UDPRecv

UDPRecv 함수는 소켓으로부터 들어오는 데이터그램을 수신하고, 관련된 어드레스 및 포트

번호를 반환하는 함수이다.

```
int16 UDPRecv(
    SOCKET s,
    UInt8 * buf
    uint16 len
    uint8 * addr,
    uint16 *port
);
```

Parameters

s

[입력] 초기화할 소켓의 번호를 명시한다.

buf

[출력] 수신한 데이터를 저장할 버퍼

len

[입력] 버퍼의 길이

addr

[출력] 데이터그램을 송신한 단말의 IP 주소를 저장할 버퍼

port

[입력] 데이터그램을 송신한 단말의 포트 번호를 저장할 변수

Remarks

Return Values

에러가 발생하지 않는 경우 UDPRecv 함수는 수신한 데이터 길이를 반환 한다. 에러가 발생한 경우에는 아래 표와 같은 값을 리턴한다.

Error Codes

ERROR_CLOSED	소켓이 닫혀있음
ERROR_NOT_UDP_SOCKET	UDP 소켓이 아님

See Also

[UDPOpen](#), [UDPSend](#), [UDPClose](#)

3.4.4 UDPClose

UDPClose 함수는 소켓을 닫는 함수이다.

```
int8 UDPClose(
    SOCKET s,
);
```

Parameters

s

[입력] 소켓의 번호를 명시한다.

Remarks

Return Values

에러가 발생하지 않는 경우 UDPClose 함수는 SUCCESS를 반환 한다. 에러가 발생한 경우에는 FAIL을 리턴한다.

Error Codes

SUCCESS	성공적으로 소켓을 닫았음
FAIL	소켓 종료를 실패 하였음

See Also

[UDPOpen](#), [UDPSend](#), [UDPRecv](#)

3.4.5 GetUDPSocketStatus

GetUDPSocketStatus 함수는 소켓을 TCP로 설정된 소켓의 상태를 알려준다.

```
int8 GetUDPSocketStatus(
    SOCKET s,
);
```

Parameters

s

[입력] 소켓의 번호를 명시한다.

Remarks**Return Values**

소켓의 상태에 따라 아래 표와 같은 값을 리턴한다.

SOCK_CLOSED	소켓이 닫혀있음
SOCK_UDP	소켓이 UDP 모드로 초기화 되었음

Error Codes

없음

See Also

[UDPOpen](#), [UDPSend](#), [UDPRecv](#), [UDPClose](#)

4. APPENDIX

4.1 Return Values

Type	Name	Value
-	SUCCESS	1
	FAIL	0
ERROR Message	ERROR_TIME_OUT	-8
	ERROR_WINDOW_FULL	-7
	ERROR_CLOSE_WAIT	-6
	ERROR_FIN_WAIT	-5
	ERROR_NOT_ESTABLISHED	-4
	ERROR_CLOSED	-3
	ERROR_NOT_UDP_SOCKET	-2
	ERROR_NOT_TCP_SOCKET	-1
Status Check	STATUS_CLOSED	-1
	STATUS_INIT	0
	STATUS_LISTEN	1
	STATUS_SYNSENT	2
	STATUS_SYNRECV	3
	STATUS_ESTABLISHED	4
	STATUS_FIN_WAIT	5
	STATUS_CLOSING	6
	STATUS_TIME_WAIT	7
	STATUS_CLOSE_WAIT	8
	STATUS_LAST_ACK	9
	STATUS_UDP	10

Document History Information

Version	Date	Descriptions
Ver. 0.9beta	Nov,2011	

Structure

[wiz_NetInfo](#)

Common

[wizInit](#), [wizSWReset](#), [wizMemInit](#), [SetNetInfo](#), [GetNetInfo](#), [SetSocketOption](#)

TCP

[TCPServerOpen](#), [TCPClientOpen](#), [TCPSend](#), [TCPReSend](#), [TCPrecv](#), [TCPClose](#), [GetTCPSocketStatus](#)

UDP

[UDPOpen](#), [UDPSend](#), [UDPrecv](#), [UDPClose](#), [GetUDPSocketStatus](#)