

QSocket: I/O

This thread has been locked by a moderator.



33

IMPORTANT If you haven't yet read [Calling BSD Sockets from Swift](#), do that first.

To read and write a connected socket, use the `FileDescriptor` [read](#) and [write](#) methods.

To read and write an unconnected socket, you need helpers to convert the address:

```
extension FileDescriptor {  
  
    /// Sends a datagram to an address.  
    /// Equivalent to the `sendto` BSD Sockets call.  
    /// If you're working with a TCP socket, use  
    /// `write(data:retryOnInterrupt:)` method.  
    /// - important: This builds the destination address from the supplied  
    /// string every time you send a datagram. That's horribly inefficient.  
    /// That's not a problem given the design constraints of this package but,  
    /// oh gosh, don't use this in a real project.  
    /// If the socket is non-blocking, be prepare for this to throw `EAGAIN`.  
    /// The result is discardable because this method is most commonly used with  
    /// a UDP socket and that's all or nothing.  
  
    @discardableResult  
    func send(data: Data, flags: CInt = 0, to destination: (address: String, port: UInt16), retryOnInterrupt: Bool = true) throws -  
    > Int {  
        try data.withUnsafeBytes { buf in  
            try QSocketAddr.withSockAddr(address: destination.address, port: destination.port) { sa, saLen in  
                try errnoQ(retryOnInterrupt: retryOnInterrupt) {  
                    // If `count` is 0 then `baseAddress` might be zero. We're  
                    // assuming that the `sendto` call will be OK with that.  
                    Foundation.sendto(self.rawValue, buf.baseAddress, buf.count, flags, sa, saLen)  
                }  
            }  
        }  
    }  
  
    /// Receive a datagram and its source address.  
    /// Equivalent to the `recvfrom` BSD Sockets call.  
    /// If you're working with a TCP socket, use the  
    /// `read(maxCount:retryOnInterrupt:)` method.  
    /// - important: This builds the destination address string from the  
    /// returned address every time you receive a datagram. That's horribly  
    /// inefficient. That's not a problem given the design constraints of this  
    /// package but, oh gosh, don't use this in a real project.  
    /// If the socket is non-blocking, be prepare for this to throw `EAGAIN`.  
    /// The result is non-optional because UDP allows us to send and receive  
    /// zero length datagrams.  
  
    func receiveFrom(maxCount: Int = 65536, flags: CInt = 0, retryOnInterrupt: Bool = true) throws -> (data: Data, from: (address: String, port: UInt16)) {  
        var result = Data(count: maxCount)  
        let (bytesRead, address, port) = try result.withUnsafeMutableBytes { buf in  
            try QSocketAddr.fromSockAddr { sa, saLen in  
                try errnoQ(retryOnInterrupt: retryOnInterrupt) {  
                    recvfrom(self.rawValue, buf.baseAddress, buf.count, flags, sa, &saLen)  
                }  
            }  
        }  
        result = result.prefix(bytesRead)  
        return (result, (address, port))  
    }  
}
```

Wrappers for the other BSD Sockets I/O primitives are left as an exercise for the reader [1].

Share and Enjoy

Quinn "The Eskimo!" @ Developer Technical Support @ Apple
let myEmail = "eskimo" + "1" + "@" + "apple.com"

[1] Good luck with `sendmsg` and `recvmsg`! (-:

Network

Reply

Posted 5 days ago by eskimo

Add a Comment

This site contains user submitted content, comments and opinions and is for informational purposes only. Apple disclaims any and all liability for the acts, omissions and conduct of any third parties in connection with or related to your use of the site. All postings and use of the content on this site are subject to the Apple Developer Forums Participation Agreement.

Platforms

- iOS
- iPadOS
- macOS
- tvOS
- watchOS
- Tools
- Swift
- SwiftUI
- SF Symbols
- Swift Playgrounds
- TestFlight
- Xcode
- Xcode Cloud

Topics & Technologies

- Accessibility
- Accessories
- App Extensions
- App Store
- Audio & Video
- Augmented Reality
- Business
- Design
- Distribution
- Education
- Fonts
- Games
- Health & Fitness
- In-App Purchase
- Localization
- Maps & Location
- Machine Learning
- Security
- Safari & Web

Resources

- Documentation
- Curriculum
- Downloads
- Forums
- Videos
- Support
- Support Articles
- Contact Us
- Bug Reporting
- System Status
- Account
- Apple Developer
- App Store Connect
- Certificates, IDs, & Profiles
- Feedback Assistant

Programs

- Apple Developer Program
- Apple Developer Enterprise Program
- App Store Small Business Program
- WiFi Program
- News Partner Program
- Video Partner Program
- Security Bounty Program
- Security Research Device Program
- Events
- App Accelerators
- App Store Awards
- Apple Design Awards
- Apple Developer Academies
- Entrepreneur Camp
- Tech Talks
- WWDC