

PAttachPH

Add a protocol handler to the protocol table

#include <AppleTalk.h>

AppleTalk Manager

OSErr **PAttachPH**(*thePBptr*, *async*);
MPPPBPtr *thePBptr*; pointer to an LAPparms structure
Boolean *async*; 0=await completion; 1=immediate return
returns Error Code; 0=no error

PAttachPH adds the protocol handler pointed to by the handler field of the LAPparms structure to the node's protocol table.

thePBptr is a pointer to an LAPparms structure.

<u>Out-In</u>	<u>Name</u>	<u>Type</u>	<u>Size</u>	<u>Offset</u>	<u>Description</u>
→	csCode	<u>short</u>	2	26	always <u>attachPH</u>
→	protType	<u>char</u>	1	28	ALAP protocol type
→	handler	<u>Ptr</u>	4	30	protocol handler

async is a Boolean value. Use FALSE for normal (synchronous) operation or TRUE to enqueue the request and resume control immediately. See Async I/O.

Returns: an operating system Error Code. It will be one of:

noErr	(0)	No error
lapProtErr	(-94)	Error attaching protocol type

Notes: handler is the protocol handler to add to the protocol table. protType specifies what kind of frame the protocol handler can service. After **PAttachPH** is called, the protocol handler is called for each incoming frame whose ALAP protocol type equals protType.

Most programs will never need to call ALAP, because higher-level protocols will automatically call ALAP as necessary. If you do want to send a frame directly via ALAP, call the **PWriteLAP** function. There is no **PReadLAP** function. If you want to read ALAP frames, you must call **PAttachPH** to add your protocol handler to the node's protocol handler table. The ALAP module will examine every incoming frame and call your protocol handler for each frame received with the correct ALAP protocol. When your program no longer wants to receive frames with a particular ALAP protocol type value, it can call **PDetachPH** to remove the corresponding protocol handler from the protocol handler table.