

به نام خدا

- Main : در ابتدا network handler برای راه اندازی اینترفیس وای فای و اینترفیس لوپ بک استفاده می شود سپس در دو تدر مقلف مدوله و دیمدوله را استارت می کنیم و اطلاعات را بارگیری میکند.

Network Handler : شامل تنها یک اینستنس می باشد و آن را به صورت استاتیک استفاده می کنیم حال دو متد آن را توضیح می دهیم

۱. initForAll : در این متد ابتدا تمام اینترفیس های pc را به ما نمایش داده و ما بین آن ها انتخاب می کنیم سپس تمام ترافیک اینترفیس مورد نظر بررسی می شود.

۲. initForUdpSafe : اینترفیس لوپ بک udpsafe ساخته شده و تمام ترافیک آن بررسی می شود.

۳. sendFrameForLoopBack : این تابع تمام اطلاعات تغییر یافته ی wifi را برای لوپ بک ارسال می کند.

۴. newConnectionWithPassword : برای ایبار اینترفیس لوپ بک نیاز است که به pc خود ssh بزنییم و کامند مورد نظر را اجرا کنیم که در این متد authenticate برای ایبار دسترسی ssh ایبار می شود.

۵. getBytes

- ArrayConverter : ابزار است برای تبدیل واحد های مقلف

- Modulation : این کلاس از کلاس تدر ارث می برد و در متد receiveFrameAndModulation اطلاعات را از wifi دریافت کرده و تغییرات مورد نیاز را اعمال می کند.

- DeModulation : این کلاس از کلاس تدر ارث می برد و در متد deModulationAndSendUpper اطلاعات را از udpsafe دریافت کرده و بررسی های لازم را انجام می دهد.

- Checksum : این کلاس با توابع خود checksum را می سازد.

۱. calculateChecksum : این تابع با دریافت بایت های مورد نظر که باید checksum آن ها مناسبه شود این مقدار را مناسبه نموده و باز می گرداند.

۲. checkChecksum : این تابع تابعی است که در سمت گیرنده اجرا شده و صحت داده را با checksum موجود در پکت می سنبد.

- CRC32 : این کلاس CRC را مناسبه می کند.

۱. xorOper : این تابع مقدار تقسیم دو عدد باینری را توسط عملگر xor حساب می کند و در خروجی باقیمانده را بر می گرداند.

۲. crcDriver : این تابع با استفاده از توابع این کلاس مقدار CRC را مناسبه می کند و بر می گرداند.

۳. deCrcDriver : این تابع مقدار CRC را با داده چک میکند که در صورت صفر بودن باقی مانده مقدار را تایید و در غیر این صورت پیغام خطا را نمایش می دهد.

- HammingCode : این کلاس برای مناسبه ی کد همینگ است.

۱. print : این تابع اعضای یک آرایه عدد صحیح را پرینت می کند.

۲. calculation : این تابع مقدار مناسب را برای هر بیت توازن مناسبه و در ممل مناسب آن بیت توازن از داده ها قرار می دهد.

۳. generateCode : این تابع آرایه جدید را برای بازسازی کد اصلی به کد همینگ ایجاد و بازمیکرداند و مقدار بیت های توازن را صفر می گذارد.

۴. degenerateCode : این تابع کد اصلی را (داده اصلی را) از بین بیت های توازن جدا کرده و آرایه داده های اولیه را برمی گرداند

۵. converter : این تابع آرایه اعداد صحیح را به آرایه ی بایت ها تبدیل کرده تا قابل مقدار دهی به پکت شود.

۶. `modulatorDriver` : این تابع مقدار بایت های داده ی مورد نظر را میگیرد و بایت های که مورد نظر که حالا دارای همینک که نیز هستند را بر می گرداند.

۷. `demodulatorDriver` : این تابع مقدار بایت های داده ماثوله شده را دریافت می کند و بایت های که اصلی داده که قبل از اضافه کردن که همینک وجود داشت را بر می گرداند.