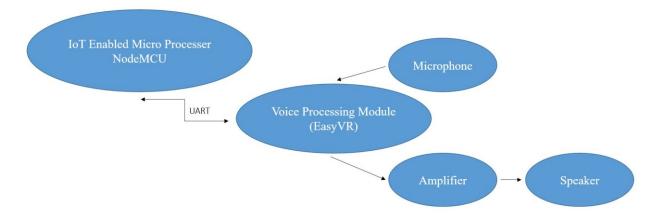
## توضيحات سخت افزارى:

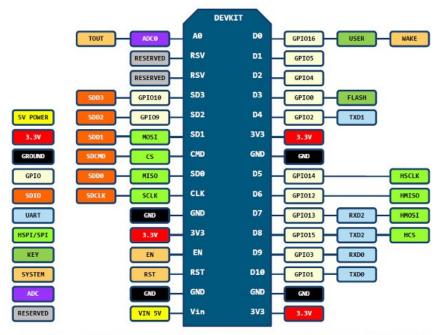
Diagram مرتبط با hardware استفاده شده براى ساخت ربات:



جزئيات دقيق سخت افزار طراحي شده و نحوه اتصالات ماژول هاي در فايل "Hardware Documentation " ضميمه شده است.

## توضیحات مربوط به ماژول NodeMCU و نحوه پروگرم کردن آن:

مهمترین ماژول استفاده شده برای توسعه ربات ماژول NodeMCU است که مبتنی بر ESP8266 است. تصویری از Layout پایه های برد:



D0(GPI016) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

برای برنامه نویسی به زبان Lua روی این برد ابتدا باید از firmware های توسعه داده شده متن باز استفاده کرد.

Firmware استفاده شده بر روی NodeMCU با استفاده از سایت /Nodemcu-build.com

Custom-build شده است و تنها از 10 ماژول ضروری تشکیل شده تا کمترین میزان از حافظه فلش را اشغال کند. برای ریختن firmware روی برد باید از نرم افزار NodeMCU Flasher استفاده کرد و Flash Size را روی 1MByte قرار داد.

برای فرستادن اسکریپت های Lua بر روی NodeMCU از یک IDE توسعه داده شده تحت Java به نام ESPlorer استفاده شده است که برای استفاده از آن باید Java Runtime Environment (JRE) برروی سیستم عامل نصب گردد.

فایل های مورد نیاز ، firmware استفاده شده، برنامه Flasher و ESPlorer IDE ضمیمه شده اند.

## توضیحات مربوط به کد توسعه داده شده به زبان Lua:

پیش از اجرا کردن کد اصلی (Stuffed\_Animal\_v3.2.lua) یک اسکریپت باید روی برد صرفا اجرا شود (نیازی به save نیست)، این اسکریپت دو فایل با تنظیمات پیش فرض برای log کردن status صداهای ذخیره شده روی EasyVR و اطلاعات مرتبط با نحوه استفاده کودک از ربات، روی NodeMCU ایجاد میکند تا برنامه با تنظیمات پیش فرض اجرا شود و در حین اجرا نیز آن ها را تغییر دهد به طوری که اطلاعات با خاموش و روشن شدن برد از بین نروند. پس از ایجاد فایل ها باید کد اصلی را تحت فایلی با نام init.lua روی برد ذخیره کرد. برد هر مرتبه پس از روشن شدن به دنبال فایل init.lua میگردد و آن را اجرا میکند.

این کد (createFiles.lua) و کد اصلی سیستم ضمیمه شده اند.

توجه شود که کد پس از اجرا شدن تنها درگاه serial com موجود روی برد را اشغال میکند و دیگر نمیتوان از طریق USB با pc ارتباط داشت و تنها راه پروگرم کردن دوباره برد این است که آن را فلش کرد.

کد Lua به صورت event driven توسعه داده شده است که در صورت رخ دادن هر event به عنوان مثال تغییر ولتاژ پایه ها، دریافت دستور از app، دریافت اطلاعات از برد پردازش صدا و ... function های مربوط که در ابتدا تعریف شده اند صدا زده میشوند.

پس از روشن شدن، برد در حالت Access Point قرار میگیرد و station ای با "ESP-AP" و "iz345678" Access Point پستیبانی میکند که ایجاد میکند تا کاربر، smartphone خود را به آن متصل کند. لازم به ذکر است برد از پروتکل های ارتباطی bgn پشتیبانی میکند که برای بهینه کردن مصرف انرژی و برد و فرکانس ارتباط از g mode استفاده شده است. سه حالت power saving نیز برای این برد تعبیه شده که حالت میانی یعنی Light\_Sleep بهترین بازدهی را برای کار مورد نظر ما دارد و از آن استفاده شده است.

پس از اتصال به App، برنامه به طور مداوم منتظر دریافت دستورات از سمت client است. این دستورات در اپلیکیشن android توسعه داده شده، به صورت encode ،json string میشوند و در سمت برد نیز ابتدا decode شده و سپس درخواست مربوطه اجرا میشود، در صورت نیاز به ارسال پاسخ به app نیز باید همچنان از فرمت json استفاده کرد.

Contract تنظیم شده برای ترجمه دستورات دریافت شده از سمت client در فایلی نوشته شده است که برای ارتباط با برد، client فقط مجاز است از این قرار داد استفاده کند. فایل contract و برنامه اندروید توسعه داده شده نیز ضمیمه شده اند.

## توضیحات مربوط به نحوه پاسخ دهی ربات به رفتارهای متفاوت کودک:

علاوه بر قابلیت کنترل ربات توسط یک اُپراتور شخص سوم، ربات میتواند به خودی خود نیز تا حد مطلوبی با کودک تعامل داشته باشد.

پاسخ های ربات شامل تو انایی لرزش بدن و پخش کردن صداهای synthesize شده ایست که بر روی EasyVR ذخیره شده است.

به طور کلی میتوان 32 فایل صوتی روی برد EasyVR ذخیره کرد که 18 تای آن مربوط به صداهای مورد نیاز برای Turn به طور کلی میتوان 32 فایل صوتی روی برد Taking Therapy و 14 تای آن مربوط به پاسخ های تعاملی (Interactive) ربات برای تعامل با کودک و پاسخ مناسب به او است.

لیست صداهای موجود بر روی برد نیز ضمیمه شده است.