# 

# دانشکده فنی مهندسی

بخش مهندسی کامپیوتر

# پروژه کارشناسی

# دانشجویان:

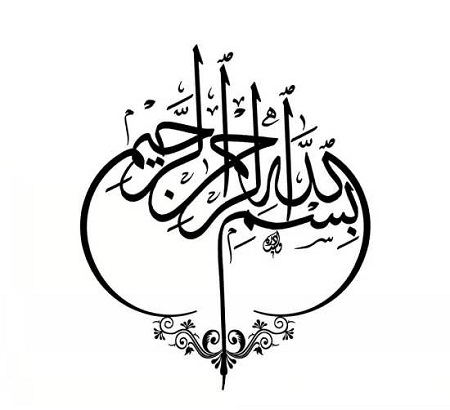
# محمد طه نامجو

متین جهانگیری

**استادان:**

**دکتر قاضی­زاده و دکتر محمدی**

**ترم4002**



# **به ﻧﺎم ﺧﺎﻟﻖ ﭘﺪﯾﺪه­ﻫﺎ و ﺷﮕﻔﺘﯽ­ﻫﺎ**

# موضوع پروژه:

این گزارش در رابطه با راه اندازی سیستم کنترل هوشمند خانه بوسیله میکرو کنترلر آردوینو و اپلیکیشن موبایل می­باشد.

# مقدمه:

در دنیایی که سرعت پیشرفت لحظه­ به لحظه افزایش می­یابد، کشورهای پیشرفته جهان رو به سیستم­های هوشمند اینترنت اشیا(IOT) آوردند تا کارهای محاسباتی را سریع تر و دقیق­تر انجام دهند که در نتیجه آن زندگی شهروندانشان را منظم­تر و ساده­تر کنند. یکی از محیط­هایی که سیستم­های هوشمند در آن نفوذ کرده­اند خانه­ها هستند، این هوشمند سازی میتواند به صاحبان خانه در کنترل تک تک اجزا خانه از راه دور کمک کند و یا حتی در مواردی خود خانه محیطش را برای افراد خانه مدیریت کند، چیزی که در سال­های اخیر در کشورهایی چون آمریکا کم کم تبدیل به نیاز مردم آن کشور شده در کشورما نیز نمود پیدا کرده و افراد زیادی شروع به نصب این سیستم­ها در خانه­هایشان کرده­اند تا بتوانند از راه دور و یا حتی با کمترین میزان جابجایی اجزا خانه خودشان (از چراغ­ها و وسایل برقی تا مدیریت دما و سیستم­های امنیتی، از همراهی در پخت و پز غذا تا همراهی در آبیاری گلدان­ها) را انجام دهند.

این پروژه هم در راستای همین سیستم­ها انجام شده تا شروعی باشد برای توسعه و گسترش این حوزه.

در ادامه به مراحل ساخت این پروژه در بخش سخت افزار و نرم افزار و ارتباط بین آن­ها اشاره شده و سعی کردیم تا توضیحات کاملی را ارائه دهیم.

# پیاده سازی­ها:

## پیاده سازی سخت افزاری نمونه اولیه وشبیه سازی سیستم:

همان طور که می­دانید یکی از بخش­های مهم یک سیستم IOT بخش سخت افزار آن و ارتباط تک تک اجزا آن بدرستی در کنار هم است و مدیریت درست بین آنهاست.

به ﻋﻠﺖ این ﮐﻪ راه اﻧﺪازي ﯾﮏ ﺳﯿﺴﺘﻢ ﺳﺨﺖ اﻓﺰاري، ﭼﺎﻟﺶ­ﻫﺎي مهم ﻣﺮﺗﺒﻂ ﺑﺎ ﺧﻮد، ﻧﻈﯿﺮ ﻋﺪم وﺟﻮد راﻫﮑﺎر ﯾﮑﺴﺎن دﯾﺒﺎﮔﯿﻨﮓ ﺳﯿﺴﺘﻢ در ﻃﻮل ﻣﺪت ﺗﻮﺳﻌﻪ و ﺗﺴﺖ، ﻋﺪم ﺣﺼﻮل اﻃﻤﯿﻨﺎن از ﺳﻼﻣﺖ دﺳﺘﮕﺎه­ﻫﺎي ﻣﻮرد اﺳﺘﻔﺎده، اﻣﮑﺎن رﺧﺪاد اﺷﺘﺒﺎه در ﺳﯿﻢ ﺑﻨﺪي و ﻏﯿﺮه را دارد، ما ﺗﺼﻤﯿﻢ ﺑﻪ ﺳﺎﺧﺖ ﻧﻤﻮﻧﻪ اوﻟﯿﻪ این سیستم ﻫﻮﺷﻤﻨﺪ ﮔﺮفتیم و ﻫﻤﭽﻨﯿﻦ ﺑﻪ ﻣﻮازات ﺳﺎﺧﺖ ﻧﻤﻮﻧﻪ اوﻟﯿﻪ ﺳﺨﺖ اﻓﺰاري ﺳﯿﺴﺘﻢ، ﻧﻈﺮ ﺑﻪ اﯾﻦ ﺷﺪ ﮐﻪ ﺳﯿﺴﺘﻢ ﻣﻮرد ﻧﻈﺮ در ﯾﮏ ﺑﺴﺘﺮ ﺷﺒﯿﻪ ﺳﺎزي ﻧﯿﺰ اﯾﺠﺎد ﮔﺮدد ﺗﺎ ﺣﺘﯽ اﻻﻣﮑﺎن از ﻫﺪر رﻓﺖ ﻫﺰﯾﻨﻪ و زﻣﺎن ﺑﺮاي ﺗﺴﺖ و دﯾﺒﺎﮔﯿﻨﮓ ﮐﺪﻫﺎي ﻧﺮم اﻓﺰاري ﺻﺮﻓﻪ ﺟﻮﯾﯽ ﺑﻪ ﻋﻤﻞ آﯾﺪ.

### برنامه پیاده سازی:

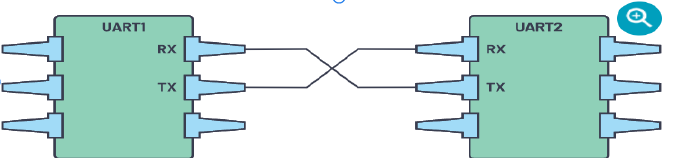
ﺑﺎ ﺗﻮﺟﻪ ﺑﻪ ﺗﻮﺿﯿﺤﺎت ﻓﻮق، ﻧﻈر ﺑﺮ اﯾﻦ ﺷﺪ ﮐﻪ ﻟﻮازم ﺳﺨﺖ اﻓﺰاري اﻋﻢ از ﻣﯿﮑﺮوکنترلر، ﺳﻨﺴﻮرﻫﺎ، ﻣﺎژول­ها و دﺳﺘﮕﺎهﻫﺎي ورودي و ﺧﺮوﺟﯽ ﺗﻬﯿﻪ ﮔﺮدد و ﻧﺴﺒﺖ ﺑﻪ ﭘﯿﺎده ﺳﺎزي ﻧﻤﻮﻧﻪ اوﻟﯿﻪ اﻗﺪام ﮐﻨﯿﻢ. راﻫﮑﺎر ﭘﯿﺎده ﺳﺎزي ﻧﯿﺰ ﺑﻪ ﺻﻮرت Unit Test و ﺳﭙﺲ Merging در دﺳﺘﻮرﮐﺎر ﻗﺮار ﮔﺮﻓﺖ ﺑﺪﯾﻦ ﺻﻮرت ﮐﻪ اﺟﺰاي ﺳﺨﺖ اﻓﺰار، ﺑﻪ ﺻﻮرت ﺗﮑﯽ راه اﻧﺪازي و ﺗﺴﺖ ﻣﯽ­ﺷﺪﻧﺪ و ﭘﺲ از اﻃﻤﯿﻨﺎن از ﻋﻤﻠﮑﺮد درﺳﺖ و آن­ها، وارد ﻣﺠﻤﻮﻋﻪ ﻧﺮم اﻓﺰاري و ﺳﺨﺖ اﻓﺰاري ﺗﻬﯿﻪ ﺷﺪه ﻣﯽ­ﺷﺪﻧﺪ.

در زﯾﺮ ﺗﺼﺎوﯾﺮ ﻣﺮﺗﺒﻂ ﺑﺎ ﻧﻤﻮﻧﻪ اوﻟﯿﻪ ﺳﺎﺧﺘﻪ ﺷﺪه را ﻣﺸﺎﻫﺪه ﺧﻮاﻫﯿﺪ ﮐﺮد.

از آنجایی که ما برای مدیریت تمام قطعات نیاز به یک میکرو کنترلر داشتیم از میکرو Arduino mega2560 استفاده کردیم. پس در ابتدا کمی راجع به انواع پروتوکل­های اتصال و ارتباط در Arduino سخن می­گوییم.

#### ﭘﺮوﺗﮑﻞﻫﺎي ﻣﻮرد اﺳﺘﻔﺎده در ﭘﺮوژه و ﻧﺤﻮه ﻋﻤﻠﮑﺮد آﻧ­ﻬﺎ:

1. UART(Serial): ﯾﮏ ﭘﺮوﺗﮑﻞ اﻧﺘﻘﺎل داده دو ﻃﺮﻓﻪ اﺳﺖ ﮐﻪ ﺑﻪ ﺟﻬﺖ اﻧﺠﺎم اﯾﻦ اﻣﺮ از دو ﭘﺎﯾﻪ TX و RX در هر دو دستگاه(مبدا و مقصد) استفاده می­کند.



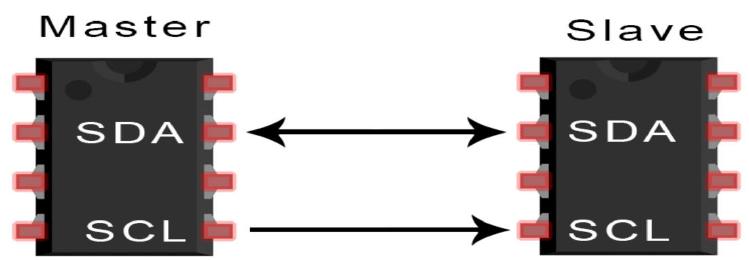
ﻧﺤﻮه ﮐﺎر ﭘﺮوﺗﮑﻞ ﺑﻪ اﯾﻦ ﻧﺤﻮ اﺳﺖ ﮐﻪ دﺳﺘﮕﺎه ﺷﻤﺎره 1 از ﻃﺮﯾﻖ ﭘﺎﯾﻪ TX ﺧﻮدش( ﮐﻪ ﺑﻪ ﭘﺎﯾﻪ RX دﺳﺘﮕﺎه ﻣﻘﺎﺑﻞ ﻣﺘﺼﻞ اﺳﺖ)، داده را ارﺳﺎل ﻣﯽﮐﻨﺪ و دﺳﺘﮕﺎه ﺷﻤﺎره 2 ﻫﻤﯿﻦ داده را از ﻃﺮﯾﻖ ﭘﺎﯾﻪ RX

ﺧﻮدش ﻣﯽﺧﻮاﻧﺪ. ﻫﻤﭽﻨﯿﻦ دﺳﺘﮕﺎه ﺷﻤﺎره 2 ﻧﯿﺰ ﺑﺮاي ارﺳﺎل داده، داده را روي ﭘﺎﯾﻪ TX ﺧﻮد ﮐﻪ ﺑﻪ ﭘﺎﯾﻪ RX دﺳﺘﮕﺎه ﻣﻘﺎﺑﻞ ﻣﺘﺼﻞ اﺳﺖ ﻗﺮار ﻣﯽدﻫﺪ.

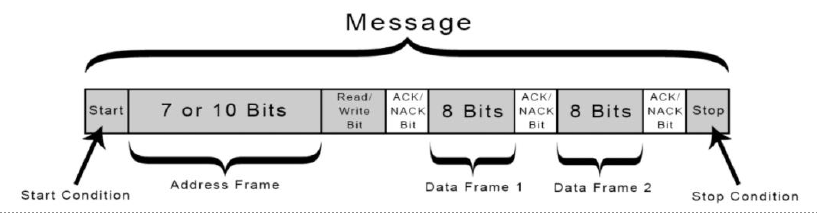
از ﻣﺤﺪودﯾﺖﻫﺎي اﯾﻦ ﭘﺮوﺗﮑﻞ ﻣﯽﺗﻮان ﺑﻪ ﻣﺤﺪود ﺑﻮدن ﺗﻌﺪاد دﺳﺘﮕﺎهﻫﺎ اﺷﺎره ﮐﺮد ﮐﻪ در ﻫﺮ اﺗﺼﺎل UART ﻓﻘﻂ و ﻓﻘﻂ دو دﺳﺘﮕﺎه ﺑﺎ ﻫﻢ ارﺗﺒﺎط ﺧﻮاﻫﻨﺪ داﺷﺖ.

ﺳﺮﻋﺖ اﻧﺘﻘﺎل داده در اﯾﻦ ﭘﺮوﺗﮑﻞ ﺗﺤﺖ ﻋﻨﻮان Baud Rate ﻣﺸﺨﺺ ﻣﯽ­ﺷﻮد ﮐﻪ ﻣﻌﻤﻮل ﺗﺮﯾﻦ آﻧﻬﺎ 9600 و 115200 اﺳﺖ.

1. ﭘﺮوﺗﮑﻞIIC(I2C): اﯾﻦ ﭘﺮوﺗﮑﻞ ﻧﯿﺰ ﯾﮏ ﭘﺮوﺗﮑﻞ اﻧﺘﻘﺎل داده دو ﻃﺮﻓﻪ اﺳﺖ اﻣﺎ ﻣﺰﯾﺖ ﺑﺴﯿﺎر ﻣﻬﻢ آن ﻧﺴﺒﺖ ﺑﻪ ﭘﺮوﺗﮑﻞ UART، ﻣﻮﺟﻮد ﺑﻮدن ﮔﺬرﮔﺎه داده اﺷﺘﺮاﮐﯽ اﺳﺖ ﮐﻪ ﺑﻪ اﯾﻦ ﺻﻮرت، ﻣﺤﺪودﯾﺖ ﺗﻌﺪاد دﺳﺘﮕﺎهﻫﺎي ﻣﺘﺼﻞ از ﻃﺮﯾﻖ اﯾﻦ ﭘﺮوﺗﮑﻞ از دو دﺳﺘﮕﺎه ﺑﻪ 255 دﺳﺘﮕﺎه اﻓﺰاﯾﺶ ﻣﯽﯾﺎﺑﺪ

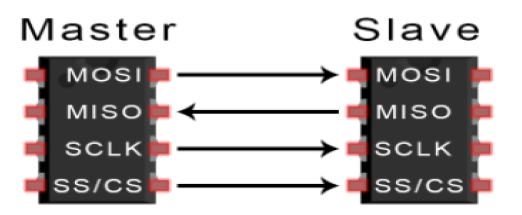


در اﯾﻦ ﭘﺮوﺗﮑﻞ دﺳﺘﮕﺎهﻫﺎ ﺑﻪ دو دﺳﺘﻪ Master و Slave ﺗﻘﺴﯿﻢ ﻣﯽﺷﻮﻧﺪ. در دﯾﺘﺎﻓﺮﯾﻢ ارﺳﺎﻟﯽ ﺗﻮﺳﻂ Master ﺑﺮ روي ﮔﺬرﮔﺎه داده، ﺑﻪ ﺟﻬﺖ اﻧﺘﺨﺎب دﺳﺘﮕﺎه ﻣﺪ ﻧﻈﺮي ﮐﻪ ﻣﺘﺼﻞ ﺑﻪ ﮔﺬرﮔﺎه اﺷﺘﺮاﮐﯽ اﺳﺖ، ﺑﺎ 8ﺑﯿﺖ، آدرس Slave ﻣﻮرد ﻧﻈﺮ ﻣﺸﺨﺺ ﻣﯽﺷﻮد. ﻫﻤﭽﻨﯿﻦ ﺑﺎ ﯾﮏ ﺑﯿﺖ R/W ﻣﺸﺨﺺ ﻣﯽﺷﻮد ﮐﻪ Master ﻗﺼﺪ ﻧﻮﺷﺘﻦ دﯾﺘﺎ ﺑﺮ روي دﺳﺘﮕﺎه ﺟﺎﻧﺒﯽ را دارد(0) و ﯾﺎ ﻗﺼﺪ ﺧﻮاﻧﺪن دﯾﺘﺎ از آن(1). در ﺻﻮرت 0 ﺑﻮدن اﯾﻦ ﺑﯿﺖ، ﺑﺎ ﮐﻼك ﺑﻌﺪي دﯾﺘﺎي ﻣﻮرد ﻧﻈﺮ Master ﺗﻮﺳﻂ Master ﺑﺮ روي ﮔﺬرﮔﺎه ﻗﺮار داده ﻣﯽﺷﻮد ﺗﺎ Slave ﺑﺘﻮاﻧﺪ دﯾﺘﺎ را از روي ﺧﻄﻮط داده ﺑﺮداﺷﺖ ﮐﻨﺪ و در ﺻﻮرت 1 ﺑﻮدن اﯾﻦ ﺑﯿﺖ، ﺑﺎ ﮐﻼك ﺑﻌﺪي دﯾﺘﺎي ﻣﻮرد ﻧﻈﺮ Master ﺗﻮسط Slave ﺑﺮ روي ﮔﺬرﮔﺎه ﻗﺮار ﻣﯽﮔﯿﺮد ﺗﺎ Master ﺑﺘﻮاﻧﺪ دﯾﺘﺎ را از ﮔﺬرﮔﺎه داده ﺑﺮداﺷﺖ ﮐﻨﺪ.



از ﻣﺰاﯾﺎي ﻣﻬﻢ اﯾﻦ ﭘﺮوﺗﮑﻞ، راه اﻧﺪازي و ﮐﺎر ﺑﺎ 255 دﺳﺘﮕﺎه ﻣﺨﺘﻠﻒ ﻓﻘﻂ و ﻓﻘﻂ ﺑﺎ اﺳﺘﻔﺎده از 2 ﭘﯿﻦ اﺳﺖ ﮐﻪ ﻣﯽﺗﻮان ﮔﻔﺖ ﺑﺮاي ﭘﺮوژه ﻓﻌﻠﯽ، ﯾﮑﯽ از ﺑﺰرﮔﺘﺮﯾﻦ ﻣﺤﺪودﯾﺖ ﻫﺎي ﭘﺮوژه را ﻣﺮﺗﻔﻊ می­سازد و ﺑﻪ ﺗﻨﻬﺎﯾﯽ ﯾﮏ ﻋﺎﻣﻞ ﺳﻮدﺳﺎز ﺑﺮاي ﭘﺮوژه ﺧﻮاﻫﺪ ﺑﻮد ﭼﺮا ﮐﻪ ﺑﺎ اﺳﺘﻔﺎده از ﻣﺎژول­هاﯾﯽ ﮐﻪ I2C را ﭘﺸﺘﯿﺒﺎﻧﯽ ﻣﯽﮐﻨﻨﺪ، دﯾﮕﺮ ﻧﯿﺎز ﺑﻪ ﭘﯿﭽﯿﺪه ﺳﺎزي اﺗﺼﺎﻻت و ﻃﺮاﺣﯽ PCB ﭘﯿﭽﯿﺪه ﻧﺨﻮاﻫﯿﻢ ﺑﻮد و از ﻃﺮﻓﯽ در ﻣﺼﺮف ﺗﻮان اﻟﮑﺘﺮﯾﮑﯽ دﺳﺘﮕﺎه ﺻﺮﻓﻪ ﺟﻮﯾﯽ ﺧﻮاﻫﺪ ﺷﺪ.

1. ﭘﺮوﺗﮑﻞ SPI: اﯾﻦ ﭘﺮوﺗﮑﻞ ﻧﯿﺰ ﯾﮏ ﭘﺮوﺗﮑﻞ ﺳﺮﯾﺎل اﺳﺖ ﺑﺪﯾﻦ ﺻﻮرت ﮐﻪ دﯾﺘﺎ از ﻃﺮﯾﻖ ﯾﮏ ﺳﯿﻢ و ﺑﺎ ﻫﺮ ﮐﻼك ﯾﮏ ﺑﯿﺖ ارﺳﺎل ﯾﺎ درﯾﺎﻓﺖ ﻣﯽﺷﻮد. در اﯾﻦ ﭘﺮوﺗﮑﻞ ﻧﯿﺰ دﺳﺘﮕﺎهﻫﺎ ﺑﻪ دو دﺳﺘﻪ Master و Slave ﺗﻘﺴﯿﻢ ﻣﯽﺷﻮﻧﺪ. ﺑﻪ ﻃﻮر ﮐﻠﯽ در راﺑﻄﻪ Master-Slave، دﺳﺘﮕﺎه Master ﮐﻨﺘﺮل دﺳﺘﮕﺎه Slave را ﺑﺪﺳﺖ ﮔﺮﻓﺘﻪ و دﺳﺘﮕﺎه Slave دﺳﺘﻮراﺗﯽ ﮐﻪ از ﺳﻤﺖ Master درﯾﺎﻓﺖ ﻣﯽﮐﻨﺪ را اﻧﺠﺎم ﻣﯽ دﻫﺪ. در اﯾﻦ ﭘﺮوﺗﮑﻞ از 4 ﭘﯿﻦ در ﻫﺮ دﺳﺘﮕﺎه اﺳﺘﻔﺎده ﻣﯽﺷﻮد.



MOSI: ﺧﻂ ﺣﺎوي دﯾﺘﺎ از ﺳﻤﺖ Master ﺑﻪ ﺳﻤﺖ Slave

MISO: ﺧﻂ ﺣﺎوي دﯾﺘﺎ از ﺳﻤﺖ Slave ﺑﻪ ﺳﻤﺖ Master

SCLK: ﺧﻂ ﺳﯿﮕﻨﺎل ﮐﻼك

SS/CS: ﺧﻄﯽ ﮐﻪ Master ﺗﻮﺳﻂ آن ﻣﺸﺨﺺ ﻣﯽﮐﻨﺪ ﺑﻪ ﮐﺪام Slave ﻣﯽﺧﻮاﻫﺪ ارﺳﺎل دﯾﺘﺎ(دﺳﺘﻮر) اﻧﺠﺎم دﻫﺪ.

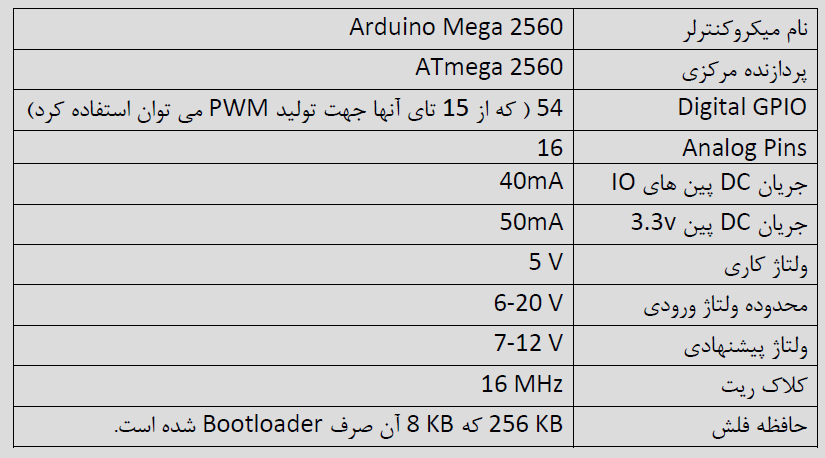
ﺑﻪ ﺻﻮرت ﮐﻠﯽ در اﯾﻦ ﭘﺮوﺗﮑﻞ، ﺳﻪ ﭘﯿﻦMISO ، MOSI و SCLK ﺑﯿﻦ ﺗﻤﺎﻣﯽ دﺳﺘﮕﺎهﻫﺎي ﻣﺘﺼﻞ ﺑﻪ ﭘﺮوﺗﮑﻞ ﺑﻪ اﺷﺘﺮاك ﮔﺬاﺷﺘﻪ ﻣﯽﺷﻮد و Master ﺑﺎ اﺳﺘﻔﺎده از ﭘﯿﻦ SS/CS ﺟﻬﺖ اﻧﺘﺨﺎب دﺳﺘﮕﺎه ﺟﺎﻧﺒﯽ ﺑﺮاي ﺧﻮاﻧﺪن ﯾﺎ ﻧﻮﺷﺘﻦ ﺑﺮ روي دﺳﺘﮕﺎه ﻋﻤﻞ ﻣﻮرد ﻧﻈﺮ ﺧﻮد را اﻧﺠﺎم ﻣﯽدﻫﺪ. ﻣﺰﯾﺖ اﯾﻦ ﭘﺮوﺗﮑﻞ ﻧﺴﺒﺖ ﺑﻪ I2C اﯾﻦ اﺳﺖ ﮐﻪ ﺗﻌﺪاد دﺳﺘﮕﺎهﻫﺎي ﻗﺎﺑﻞ اﺳﺘﻔﺎده و ﻣﺘﺼﻞ در اﯾﻦ ﭘﺮوﺗﮑﻞ ﻧﺎﻣﺤﺪود ﻫﺴﺘﻨﺪ. ﺑﺮاي ﻣﺜﺎل ﺑﺎ اﺳﺘﻔﺎده از ﯾﮏ دﯾﮑﻮدر 255\*8، ﻣﯽﺗﻮان ﺑﺎ اﺳﺘﻔﺎده از 8 ﭘﯿﻦ GPIO ﯾﮏ ﻣﯿﮑﺮو، روي 255 دﺳﺘﮕﺎه ﻋﻤﻞ ﭼﯿﭗ ﺳﻠﮑﺖ را اﻧﺠﺎم داد. و ﻗﺎﺑﻞ ﺗﻮﺟﻪ اﺳﺖ ﮐﻪ ﺑﺎ اﺳﺘﻔﺎده از ﯾﮏ ﺑﯿﺖ ﺑﯿﺸﺘﺮ، ﺗﺎ 511 دﺳﺘﮕﺎه را ﻣﯽﺗﻮان ﮐﻨﺘﺮل ﮐﺮد.

ﻫﻤﭽﻨﯿﻦ ﻻزم ﺑﻪ ذﮐﺮ اﺳﺖ ﮐﻪ ﺑﺎ اﺳﺘﻔﺎده از ﯾﮏ ﻣﺎژول GPIO Extender ﮐﻪ ﺧﻮدش از ﻃﺮﯾﻖ I2C ﺑﺎ ﻣﯿﮑﺮو در ارﺗﺒﺎط اﺳﺖ، ﺗﺎ 100 ﭘﯿﻦ GPIO ﻣﯽﺗﻮان از اﯾﻦ ﻣﺎژول درﯾﺎﻓﺖ ﮐﺮد.

ﭘﺲ در اﯾﻦ ﭘﺮوژه، ﮐﻪ ﺗﻌﺪاد دﺳﺘﮕﺎهﻫﺎي ﺟﺎﻧﺒﯽ ﻣﺘﺼﻞ ﺑﻪ ﻣﯿﮑﺮو از 20دﺳﺘﮕﺎه ﺗﺠﺎوز ﻧﻤﯽﮐﻨﺪ، ﺑﻪ ﻃﻮر ﮐﻠﯽ ﻫﯿﭻ ﻣﺤﺪودﯾﺘﯽ ﺑﺮ روي ﺗﻌﺪاد ﭘﯿﻦﻫﺎي ﻣﻮرد اﺳﺘﻔﺎده ﻧﺨﻮاﻫﯿﻢ داﺷﺖ و ﺑﻪ ﻃﻮر ﺗﻘﺮﯾﺐ، ﻫﺮ ﻣﯿﮑﺮوﯾﯽ ﭘﺎﺳﺨﮕﻮي اﯾﻦ ﻧﯿﺎز ﺧﻮاﻫﺪ ﺑﻮد.

حال که بعد از آشنایی با پروتکل­های ارتباطی آشنا شدیم به سراغ بیان:

1. ﻣﯿﮑﺮو ﻣﻮرد اﺳﺘﻔﺎده: اﺻﻠﯽ ﺗﺮﯾﻦ ﺗﺠﻬﯿﺰ ﻣﻮرد اﺳﺘﻔﺎده در اﯾﻦ ﭘﺮوژه، ﻣﯿﮑﺮو ﯾﺎ ﻫﻤﺎن ﻗﻠﺐ ﺗﭙﻨﺪه دﺳﺘﮕﺎه اﺳﻤﺎرت رك ﺧﻮاﻫﺪ ﺑﻮد.



1. اﻃﻼﻋﺎت دﺳﺘﮕﺎهﻫﺎي ﺟﺎﻧﺒﯽ ﻣﻮرد اﺳﺘﻔﺎده در ﭘﺮوژه اﻋﻢ از ﻣﺎژولﻫﺎ، ﺳﻨﺴﻮرﻫﺎ، وﺳﺎﯾﻞ ورودي و ﺧﺮوﺟﯽ ارﺗﺒﺎط ﺑﺎ ﮐﺎرﺑﺮ ﺣﺎﺿﺮ و ... ﺑﻪ ﺷﺮح ﺟﺪول زﯾﺮ ﻣﯽ ﺑﺎﺷﺪ.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ردیف | نام دستگاه | مورد کاربرد | تعداد پایه­های مورد استفاده دستگاه | پروتوکل­ها | پروتوکل مورد استفاده | پین مشترک | پین اختصاصی | ولتاژ کاری |
| 1 | ماژول رله 4تایی | 3رله کنترل کولر-1 رله کنترل پمپ | 6 | Digital | Digital | 2 | 4 | 5 |
| 2 | ماژول رله 2تایی | یک رله در- یک رله لامپ | 4 | Digital | Digital | 2 | 2 | 5 |
| 3 | ماژول بلوتوث HM-10 | ارسال دستورات به آردوینو | 4پایه از 6 تا | Digital  UART(Serial) | Serial | 2 | 2 | 3.3 |
| 4 | ماژول رطوبت خاک و زمین  YL-100 | سنجش رطوبت خاک گلدان | 3 | Digital  Analog | Analog | 2 | 1 | 5 |
| 5 | سنسور دما و رطوبتGHT11 | سنجش دمای محیط | سه پایه از 4 پایه | Digital | Digital | 2 | 1 | 5 |

لازم به ذکر است علاوه بر موارد بالا از ابزارهای جانبی زیر نیز استفاده شده است:

* کابل program برای برنامه نویسی و تغذیه برد اصلی.
* برد بورد برای اتصال اجزای سیستم.
* جامپیر(سیم اتصال):نر-ماده، ماده-ماده، نر-نر
* ﺑﺮد آردﯾﻮﻧﻮ UNO CH340 ﺑﻪ ﺟﻬﺖ ﯾﻮﻧﯿﺖ ﺗﺴﺖ ﮐﺪ اﺻﻠﯽ و اﻃﻤﯿﻨﺎن از ﻋﻤﻠﮑﺮد ﻣﯿﮑﺮو اﺻﻠﯽ

در ﻗﺴﻤﺖ ﮐﺪﻧﻮﯾﺴﯽ ﻣﯿﮑﺮو، ﺗﻤﺎم ﺳﻌﯽ ﺗﯿﻢ ﻣﺎ ﺑﺮ ﮐﺪﻧﻮﯾﺴﯽ ﻣﺎژوﻻر و ﻣﻘﯿﺎس ﭘﺬﯾﺮ ﺑﻮده اﺳﺖ ﮐﻪ در آﯾﻨﺪ ه ﭼﻨﻨﺎﭼﻪ ﺗﺼﻤﯿﻢ ﺑﻪ اﺳﺘﻔﺎده از ﻣﺎژوﻟﻬﺎ، ﺳﻨﺴﻮرﻫﺎ و ﯾﺎ ﻣﯿﮑﺮوﻫﺎي دﯾﮕﺮي ﺑﺮاي ﭘﯿﺎده ﺳﺎزي ﺳﯿﺴﺘﻢ ﺷﺪ، ﻫﺰﯾﻨﻪ ﮔﺰاﻓﯽ ﺑﺮاي ﺗﻮﻟﯿﺪ ﮐﻨﻨﺪه ﻧﺪاﺷﺘﻪ ﺑﺎشد.

ﺑﻪ ﺻﻮرت ﮐﻠﯽ، ﺳﯿﺎﺳﺖ اﺟﺮاي ﮐﺪ ﺑﺮ روي ﻣﯿﮑﺮو ﺑﻪ ﺳﻪ ﻓﺎز Initialize ،Setup و Run ﺗﻘﺴﯿﻢ ﻣﯽﮔﺮدد.

در ﻓﺎز Setup، دﺳﺘﮕﺎهﻫﺎ ﺗﻮﺳﻂ ﺗﻮاﺑﻊ ﻣﺨﺼﻮص ﺑﻪ ﺧﻮد ﮐﻪ ﺑﺮاي ﻫﺮ دﺳﺘﮕﺎه ﯾﮏ ﺗﺎﺑﻊ Setup ﺗﻬﯿﻪ ﺷﺪه اﺳﺖ، ﺑﺎ ﻓﺮاﺧﻮاﻧﯽ آن ﺗﻮاﺑﻊ، دﺳﺘﮕﺎهﻫﺎي ﻣﺘﺼﻞ ﺑﻪ ﻣﯿﮑﺮو Setup ﻣﯽﺷﻮﻧﺪ و ﻋﻤﻼً ﺑﺮاي اﺟﺮاي ﻋﻤﻞ در ﻣﺪار ﻗﺮار ﻣﯽﮔﯿﺮﻧﺪ.

در ﻓﺎز Initialize، ﺗﻌﺪادي از دﺳﺘﮕﺎه ﻫﺎ ﮐﻪ ﺑﺮاي ﺷﺮوع ﺑﻪ ﮐﺎر ﻧﯿﺎز ﺑﻪ ﻣﻘﺪاردﻫﯽ اوﻟﯿﻪ دارﻧﺪ ﺗﻮﺳﻂ ﺗﺎﺑﻊ Initialize ﻣﻘﺪاردﻫﯽ اوﻟﯿﻪ ﻣﻮرد ﻧﯿﺎز آﻧﻬﺎ اﻧﺠﺎم ﻣﯽﺷﻮد.

ﻓﺎز Run ﮐﻪ آﺧﺮﯾﻦ ﻓﺎز اﺟﺮا اﺳﺖ، اﺟﺮاي ﻣﺘﻮاﻟﯽ و داﺋﻢ ﮐﺪ ﺑﺮ روي ﻣﯿﮑﺮو اﺗﻔﺎق ﻣﯽاﻓﺘﺪ.