###### به نام خدا



###### پژوهشکده …

مقدمه‌ای بر مدارس هوشمند

پروژه:

کد پروژه:

|  |  |
| --- | --- |
| مجری: |  |
| تهیه‌کننده: |  |
| کد گزارش: |  |
| تاریخ ارائه: |  |
| نسخه/وضعیت: |  |

شناسنامه گزارش

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **عنوان:** | | | | **شماره نسخه:** |
| **كد:** | **نوع گزارش:** | | | **تاریخ ارائه گزارش:** |
| **نام پروژه:** | | | **نوع پروژه:** | |
| **تاریخ شروع:** | | | **تاریخ پایان:** | |
| **نام گروه:** | | | | |
| **كد پروژه:** | | | **شماره و تاریخ قرارداد:** | |
| **مجری:** | | | **ناظر/ ناظرین:** | |
| **تهیه كننده/ تهیه كنندگان:** | | | | |
| **نام و نشانی مجری:** | | | | |
| **نام و نشانی حمایت كننده:**  تهران، انتهای خیابان كارگر شمالی، پژوهشگاه ارتباطات و فناوری اطلاعات (مركز تحقیقات مخابرات ایران)\_ كد پستی: 1439955471 \_ تلفن: 10-8005508 | | | | |
| **ملاحظات:** | | | | |
| **چكیده:** | | | | |
| **كلمات كلیدی:** | | | | |
| **وضعیت گزارش:** | | **زبان گزارش:** | | |
| **وضعیت دسترسی:** | | **تعداد صفحات:** | | |

چکیده

افزایش تعداد دستگاه‌هایی که از طریق اینترنت به یکدیگر متصل می‌شوند، نشانه‌هایی از تحقق اینترنت اشیا است. دنیایی که در آن اشیا با حداقل دخالت انسانی با یکدیگر ارتباط دارند و سرویس‌های متنوعی را ارائه می‌کنند. به منظور دستیابی به اهداف اینترنت اشیا جنبه‌های متعددی باید در نظر گرفته شود. در این تحقیق ابتدا به معرفی اینترنت اشیا خواهیم پرداخت. سپس شهرهای هوشمند را معرفی میکنیم و چالش‌های پیش رو برای ایجاد شهرهای هوشمند را بررسی می‌کنیم. در نهایت به بررسی بسترهای موجود برای اینترنت اشیا می‌پردازیم.

کلمات کلیدی: اینترنت اشیا، شهر‌های هوشمند، بسترهای اینترنت اشیا

اطلاعات مرتبط

مستندات مرتبط

|  |  |  |
| --- | --- | --- |
| **شماره مستند** | **نوع مستند** | **نام مستند** |
|  |  |  |
|  |  |  |

تغییرات اعمال‌شده در نسخه‌های پیشین

|  |  |  |
| --- | --- | --- |
| **شماره نسخه** | **تاریخ** | **تغییرات اعمال شده** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

تأییدکنندگان

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **نام و نام خانوادگی** | **تاریخ** | **امضاء** | **ملاحظات** |
| **مجری پروژه** |  |  |  |  |
| **تهیه كننده/  تهیه كنندگان** |  |  |  |  |
| **ناظر پروژه** |  |  |  |  |
| **مدیر گروه** |  |  |  |  |
| **مسئول مستندات پژوهشكده** |  |  |  |  |
| **رئیس پژوهشكده/  معاون پژوهشی** |  |  |  |  |

تقدیر و تشکر

سرفصل مطالب

[1 فصل اول: مقدمه 8](#_Toc465534250)

[2 فصل دوم: شهرهای هوشمند 10](#_Toc465534251)

[2-1 مقدمه 10](#_Toc465534252)

[2-2 ویژگی‌ها 11](#_Toc465534253)

[2-3 چالش‌ها 11](#_Toc465534254)

[2-4 نمونه‌های بارز 12](#_Toc465534255)

[3 فصل سوم: مدرسه‌ی هوشمند 14](#_Toc465534256)

[3-1 کارها در یک مدرسه هوشمند 14](#_Toc465534257)

[3-2 وسایل هوشمند درون مدرسه 15](#_Toc465534258)

[3-3 زیرساخت مدرسه برای حمایت از مدرسه هوشمند 16](#_Toc465534259)

[3-4 سودهای فوری و ویژگی‌های ممکن 16](#_Toc465534260)

[4 فصل چهارم: پیاده سازی 18](#_Toc465534261)

[4-1 اپلیکیشن 18](#_Toc465534262)

[4-1-1 مقدمه 18](#_Toc465534263)

[4-1-2 ارتباط با سرور 18](#_Toc465534264)

[4-1-3 نحوه کارکردن اپلیکیشن 20](#_Toc465534265)

[4-2 سخت افزار 25](#_Toc465534266)

[4-2-1 مقدمه 25](#_Toc465534267)

[4-2-2 نحوه راه اندازی کلی ماژول ها 25](#_Toc465534268)

[4-2-3 مستندات مربوط به راه اندازی تک تک ماژول ها 27](#_Toc465534269)

[پیوست‌ها 37](#_Toc465534270)

# فهرست جداول

**No table of figures entries found.**

فهرست اشکال

[شکل 1 شمای یک شهر هوشمند 10](#_Toc465534007)

[شکل 2 شهرهوشمند آمستردام 13](#_Toc465534008)

[شکل 3 شبکه‌ی جدید اتوبوس در بارسلونا 13](#_Toc465534009)

[شکل 4 مدارس هوشمند در یک نگاه 14](#_Toc465534010)

[شکل 5 نمونه کد برای ارتباط با سرور 19](#_Toc465534011)

[شکل 6 نمای مدرسه 20](#_Toc465534012)

[شکل 7 صفحه‌ی کنتور 21](#_Toc465534013)

[شکل 8 نمای کلاس 22](#_Toc465534014)

[شکل 9 صفحه‌ی کولر 22](#_Toc465534015)

[شکل 10 صفحه تنظیمات کولر 23](#_Toc465534016)

[شکل 11 صفحه‌ی در 23](#_Toc465534017)

[شکل 12 صفحه‌ی گاز 24](#_Toc465534018)

[شکل 13 مدار سنسور DHT11 28](#_Toc465534019)

[شکل 14 مدار hy-srf05 30](#_Toc465534020)

[شکل 15 سنسور pir motion 35](#_Toc465534021)

# فصل اول: مقدمه

هنگامی که کاربران در ساختمان‌های اداری، مسکونی و تجاری توانستند از طریق اتصالات اینترنت باسیم با کاربران دیگر ارتباط برقرار کنند انقلابی در زمینه ارتباطات رخ داد. موج دوم این انقلاب زمانی بود که تجهیزات سیار مصرف کنندگان (لپ‌تاپ‌ها، گوشی‌های تلفن همراه هوشمند، تبلت‌ها) از طریق اتصالات بی‌سیم به یکدیگر و شبکه‌های عمومی متصل شدند؛ اما موج آخر مربوط به اتصال اشیا به کاربران، شبکه‌های تجاری و عمومی و دیگر اشیا از طریق ترکیبی از اتصال به اینترنت است. این موج اتومبیل‌ها، هواپیما‌ها، تجهیزات پزشکی، آسیاب‌های بادی، حسگر‌های محیطی، تجهیزات استخراج گاز طبیعی و بسیاری دیگر از دستگاه‌ها و تجهیزات را در بر می‌گیرد. کارایی و بهره‌وری این سیستم‌ها به شکل قابل توجهی به کمک اتصالات نظیر به نظیر و مشتری-سرویس‌دهنده که محصول پیشرفت‌های جدید در اتصالات و کنترل کننده‌های ارزان قیمت و پروتکل‌های استاندارد اینترنت است، افزایش می‌یابد.

اینترنت اشیا (IoT)، به اتصال فزاینده اشیا از هر نوع، از کاربردهای خانگی تا تجهیزات به کار رفته در کاربردهای صنعتی، به اینترنت یا ساختاری شبیه به اینترنت اشاره دارد. ایده‌ی اصلی در این مفهوم این است که تجهیزات هوشمند باید بتوانند با یکدیگر و با واسط‌های انسانی در سراسر جهان برای افزایش بهره‌وری ارتباط برقرار کنند.

به عقیده‌ی برخی کارشناسان، IoT پس از اینترنت و شبکه‌های تلفن همراه، سومین موج از فناوری‌های ICT خواهد بود. انستیتو فرستر پیش‌بینی کرده است که ارتباطات اینترنتی شی با شی تا سال ۲۰۲۰ در حدود ۳۰ برابر تعاملات ارتباطی انسان با انسان خواهد بود و این بدان معناست که میلیارد‌ها اتصال اینترنتی از آینده مخابرات، فقط مربوط به تعاملات بین اشیا خواهد بود.

انقلاب اینترنت منجر به ارتباط متقابل مردم با سرعت بی‌سابقه‌ای شده است. انقلاب آینده ارتباط میان اشیا برای ایجاد محیطی هوشمند خواهد بود. در سال ۲۰۱۱ تعداد دستگاه‌های متصل به یکدیگر به بیش از تعداد انسان‌های کره زمین رسید. بر اساس گزارش CISCO در سال ۲۰۱۲ حدود ۸.۷ میلیارد شی متصل به اینترنت جهانی وجود داشت. در سال ۲۰۱۳ این مقدار به بالغ بر ۱۰ میلیارد رسید. با کاهش هزینه برای هر اتصال و در نتیجه رشد سریع تعداد ارتباطات ماشین به ماشین، انتطار می‌رود که این تعداد در سال ۲۰۲۰ به ۵۰ میلیارد برسد. پیش‌بینی می‌شود زمان مورد نیاز به منظور بلوغ اینترنت اشیا در صنعت بین ۵ تا ۱۰ سال است.

راه اندازی کارخانه‌های هوشمند سال‌های قبل نیازمند سیستم‌های اتوماسیون گران‌قیمت و سفارشی بود که تنها کارخانه‌های بزرگ از پس هزینه‌های آن برمی‌آمدند. شرایط در هوشمندسازی منازل نیز به همین ترتیب بود و تنها بیلیونرهایی مانند بیل گیتس از پس هزینه‌های سیستم‌های سفارشی هوشمندسازی منازل بر‌می‌آمدند. موارد زیر را می‌توان از عوامل گسترش اینترنت اشیا در سال‌های اخیر برشمرد:

* نصف شدن قیمت سنسور‌ها و عملگرها در ۱۰ سال اخیر
* کوچکتر قدرتمندتر شدن سنسور‌ها و عملگر‌ها (سخت‌افزارها)
* دسترسی به ابزارات پشتیبانی مانند زیرساخت‌های ابری و ابزارات داده‌های عظیم

حوزه‌های کاربردی تحت تاثیر IoT را می‌توان به صورت زیر بیان کرد:

* خانه‌ها و ساختمان‌های هوشمند
* شهرهای هوشمند
* بخش سلامت و پزشکی
* پایش‌های محیطی
* تجارت‌های هوشمند
* امنیت و نظارت

# فصل دوم: شهرهای هوشمند

## مقدمه

شهر‌ هوشمند یک نگاه توسعه شهری برای به کارگیری روش‌هایی از اینترنت اشیا (IoT) و ICT به صورت امن برای مدیریت دارایی‌های شهری می‌باشد. هدف شهر‌های هوشمند افزایش کیفیت زندگی با استفاده از انفورماتیک شهری و تکنولوژی می‌باشد به ترتیب کیفیت سرویس‌ها افزایش می‌یابد.



شکل 1 شمای یک شهر هوشمند

از جمله کاربرد‌های اینترنت اشیا در شهر هوشمند می‌توان به مدیریت ترافیک، نور معابر، سیستم‌های سلامت شهری و سیستم‌های نظارت شهری اشاره کرد.

یک دسته‌ی بزرگ از شرکت‌های هوشمندسازی شهر‌ها در شهر تل‌‌آویو در اسرائیل قرار دارد و این شهر جایزه‌ی جهانی شهر هوشمند را در سال ۲۰۱۴ برنده شد، شرکت‌های اسرائیلی این راه‌کارها را به صورت جهانی پیاده‌سازی می‌کنند.

## ویژگی‌ها

شهرهای هوشمند به صورت مشخص تعریف نشده‌اند ولی یک شهر هوشمند به طور کلی از فناوری اطلاعات استفاده می‌کند تا بتواند:

* با استفاده از آنالیز داده‌ای و هوش مصنوعی از بستر فیزیکی (مانند جاده‌ها، دارایی‌های فیزیکی و ..) بهینه‌تر استفاده کند.

این فرم‌ها از هوشمندی در شهر‌ها می‌تواند به سه صورت‌ زیر نمود پیدا کند:

* هماهنگی‌های هوشمند: شهرهای هوشمند همکاری و ارتباطات بین انجمن‌ها و موسسات برای حل مشکلات را فراهم می‌آورد.
* اختیارات هوشمند: شهرها با فراهم آوردن بستر‌های باز و امکانات آزمایشی اجازه‌ی خلاقیت و نوآوری در مناطق را می‌دهند.
* ابزار‌های هوشمند: با وجود اینکه زیرساخت شهری با استفاده از جمع‌آوری داده‌ی بلادرنگ و پردازش و مدل‌های پیش‌بینی هوشمند می‌گردد. بحث‌های زیادی در این مورد وجود دارد که از جمله‌ی آن مشکلات نظارتی در شهر‌های هوشمند می‌باشد.

## چالش‌ها

به طور کلی چالش‌های زیر برای اینترنت اشیا مطرح هستند که قطعا در زمینه‌ی شهر‌های هوشمند نیز وجود دارند:

* تعداد بسیار زیاد اشیا

سیر کنونی در افزایش تعداد دستگاه‌های هوشمند نشان‌دهنده‌ اتصال هزاران شی به اینترنت در آینده نه چندان دور است. چگونگی دسترسی، نگهداری، محافظت، استفاده و پشتیبانی از این تعداد بسیار زیاد اشیا به عنوان یک چالش اساسی مطرح است.

* تولید دانش و داده‌های بزرگ

در دنیای IoT داده‌های خام بسیاری وجود دارند که پیوسته جمع آوری می‌شوند؛ بنابراین تکنیک‌هایی لازم است که این داده‌های خام را به دانش کاربردی مبدل سازد. چالش اصلی در تفسیر داده‌ها و شکل گیری دانش، آدرس‌دهی اطلاعات جهان فیزیکی و توسعه روش‌های استنتاج جدیدی است که محدودیت‌های روش‌های سنتی را نداشته باشد. در نظر گرفتن هزینه‌ی محاسبات از جمله این محدودیت‌ها است.

* امنیت

مشکل اساسی که امروزه در اینترنت فراگیر شده است، برخورد با حملات امنیتی است. حملات امنیتی به دلایل منابع محدود اشیا، عدم دسترسی فیزیکی به حسگرها، عملگرها و اشیا و ارتباطات بی‌سیم میان بسیاری از دستگا‌ه‌ها، در حوزه IoT مشکل‌ساز هستند.

## نمونه‌های بارز

* آمستردام: بنیان‌های شهر هوشمند آمستردام در سال ۲۰۰۹ آغاز شد که اکنون شامل ۷۹ پروژه می‌باشد که با همکاری تجارت‌ها، دولت و ساکنان محلی در حال توسعه‌اند. این پروژه‌ها روی یک بستر به هم پبوسته از طریق دستگاه‌های بی‌سیم اجرا می‌شوند تا توانایی شهر در تصمیم‌های آنی افزایش یابد. شهر آمستردام هدف خود را از هوشمندسازی کاهش ترافیک، حفظ انرژی و افزایش امنیت عمومی بیان داشته است.



شکل 2 شهرهوشمند آمستردام

* بارسلونا: بارسلونا تعدادی پروژه را کلید زده است که می‌توانند برنامه‌های شهر هوشمند با استراتژی سیستم عامل شهر (CityOS) تلقی گردند. بارسلونا یک شبکه جدید اتوبوس نیز بر اساس آنالیز داده‌ای از جریان‌های ترافیک معمول در بارسلونا طراحی کرده است.

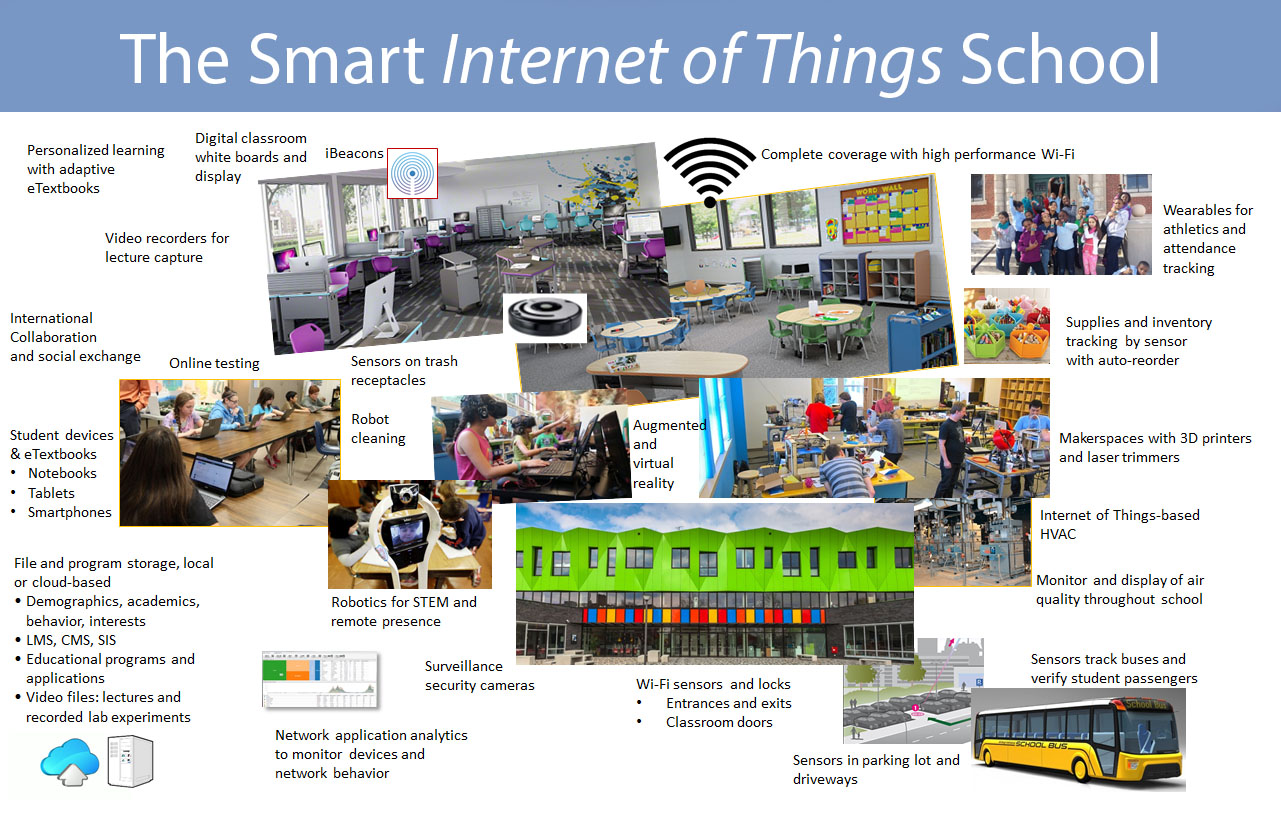


شکل 3 شبکه‌ی جدید اتوبوس در بارسلونا

# فصل سوم: مدرسه‌ی هوشمند

## کارها در یک مدرسه هوشمند

در مدرسه هوشمند حضور و غیاب دانش‌آموزان به طور خودکار انجام می‌شود، معلمان می‌فهمند که هر دانش‌آموز چقدر از مطالب را فهمیده و تا کجای مطلب را خوانده است. تست‌ها به خوبی به صورت دیجیتالی مدیریت می‌شوند. دانش‌آموزان در حین فعالیت‌های ورزشی با دستبندهای هوشمند تحت نظارت هستند. موجودی انبار به طور خودکار تامین می‌شود و سطل زباله‌های خودکار شناسایی شده و به مسئول خدمات اعلام می‌گردد.



شکل 4 مدارس هوشمند در یک نگاه

خارج از مدرسه، اتوبوس ها در نقشه های گوگل ردیابی و پارکینگ مدرسه با برنامه های گوشی های هوشمند مدیریت می شوند. روشنایی محوطه دانشگاه به نیازهای آنی بر اساس سطح محیط، آب و هوا، فعالیت های محلی، و الگوهای پیش بینی شده بهینه می‌شوند.

در داخل مدرسه، جریان هوا، کیفیت هوا، درجه حرارت و رطوبت به طور مداوم تحت نظارت و بهینه سازی می‌شود و با استفاده از دادگان آن یادگیری انجام می‌گیرد. مانیتورهای صفحه تخت در کلاس‌ها برای نمایش دادگان از وسایل دانش‌آموزان یا معلمان استفاده می‌شود.

## وسایل هوشمند درون مدرسه

عبارت اینترنت اشیا به طور کلی به ارتباط بین ماشین به ماشین (M2M) که شامل شبکه‌های مبتنی بر حسگرها و محرک‌ها از راه دور اشاره دارد. سنسورهای بی‌سیم دادگان را تولید می‌کنند (معمولا داده‌های حجیم) که در سایت‌ها یا سامانه‌های ابری ذخیره و پردازش می‌کنند. طیف وسیعی از وسایل هوشمند در مدارس استفاده می‌شوند از قبیل: کتاب‌های الکترونیکی و تبلت‌ها؛ در راهرو، ورودی، فضاهای کلاس درس، اتوبوس و تمام انواع گروه‌های تناسب اندام و پوشیدنی؛ هدست واقعیت مجازی و افزوده؛ روبات‌ها؛ سنسورهای ویدئویی. نمایش هوشمند؛ چراغ های هوشمند؛ و قفل های هوشمند.

داده‌های این وسایل را برای ردیابی‌های ساده مورد استفاده قرار می‌گیرد، به عنوان نمونه ردیابی اتوبوس مدارس، سامانه حضور و غیاب دانش‌آموزان، ردیابی تجهیزات و نظارت‌های پیچیده‌تر برای مثال درک الگوهای یادگیری دانش‌آموزان از طریق روند پیشرفت آن‌ها در کتاب‌های الکترونیکی. تمام فعالیت‌ها را به صورت زنده مشاهده و می‌توان آن‌ها را در مواقع بحرانی نظارت کرد. در سراسر مدرسه سنسورهای کم هزینه فعال هستند تا تهویه مطبوع را برای مدرسه فراهم سازند و هزینه‌ها را بهینه کنند.

## زیرساخت مدرسه برای حمایت از مدرسه هوشمند

هسته‌ی اصلی اینترنت اشیا به یک Wi-Fi قابل اعتماد بستگی شدید دارد. اینترنت اشیا به یک بستر خوب نیازمند است، این بستر باید قابلیت مقیاس‌پذیری برای رشد سریع وسایل هوشمند را داشته باشد. پرداختن به امنیت، ایمنی و نگرانی‌های حریم خصوصی. آموزش‌ عالی نیاز به یک نرم‌افزار جامع مدیریت شبکه دارد. تجزیه و تحلیل ترافیک شبکه در این کار بسیار مهم است تا وسایل شبکه را در تنگنا قرار ندهد.

## سودهای فوری و ویژگی‌های ممکن

علاوه بر مزایای مستقیم آموزشی و یادگیری، دستگاه‌ها در مدرسه‌ی هوشمند می‌تواند تاثیر مثبتی در ایمنی دانش‌آموزان دارند. هنگامی که Fraser Public Schools در میشیگان سنسورهای ویدیوئی را در سراسر مدرسه قرار داد، آن‌ها متوجه شدند که به علت اینکه دانش‌آموزان می‌دانند تصاویر آن‌ها گرفته می‌شود وقوع دعوا در مدرسه به صفر نزدیک شده ‌است و به همین علت آن‌ها بد رفتاری کم‌تری می‌کنند.

ترکیب اینترنت اشیا با دوره‌های تحصیلی و معرفی وسایل اینترنت اشیا مثل Raspberry pi و Arduino کمک شایانی به آموزش مهندسی و ریاضی می‌کند، ولی از این مهم‌تر خلاقیتی است که در دانش‌آموزان ایجاد می‌کند. تنوع وجود داده‌های حجیم فرصت یادگیری تجزیه و تحلیل داده‌ها و بینش خوبی برای علم داده را بوجود می‌آورد.

جریان رو به رشد دادگان بوجود آمده از مدارس و دانش‌آموزان آن مدارس فرصت بوجود آمدن چالشی برای تهیه‌ی برنامه‌ی منحصر به فرد هر دانش‌آموز بر اساس تجربیات دانش‌آموز فراهم کرده است در حالی که هزینه‌ها را برای این کار کاهش داده است. الگوگیری از دانش‌آموزان موفق باعث افزایش بهره‌وری دانش‌آموزان می‌شود. در یک سناریو دورتر، هم از نظر زمان و هم از نظر عجیب بودن، نه تنها این سامانه حضور و غیاب دانش‌آموزان را بررسی می‌کند بلکه با بستن یک سربند به دانش‌آموز می‌توان امواج مغزی و میزان کارایی آن را نیز مشاهده کرد.

# فصل چهارم: پیاده سازی

## اپلیکیشن

### مقدمه

یکی از قسمت‌های مهم هر کار اینترنت اشیا قسمت اپلیکیشن است، این قسمت برای کاربران ساخته می‌شود تا با استفاده از اپلیکیشن برای نظارت، ارسال تنطیماتی که کاربر می‌خواهد اشیا مطابق با آن کار کنند و هشدار دادن اشیا به کاربران استفاده می‌شود. به این علت که این قسمت با کاربران و اشخاص سر و کار دارد در طراحی‌های خود در اپلیکیشن از material design گوگل و از پویا نمایی‌های برای پویا بودن صفحات استفاده شده است تا کاربران احساس راحتی داشته باشند.

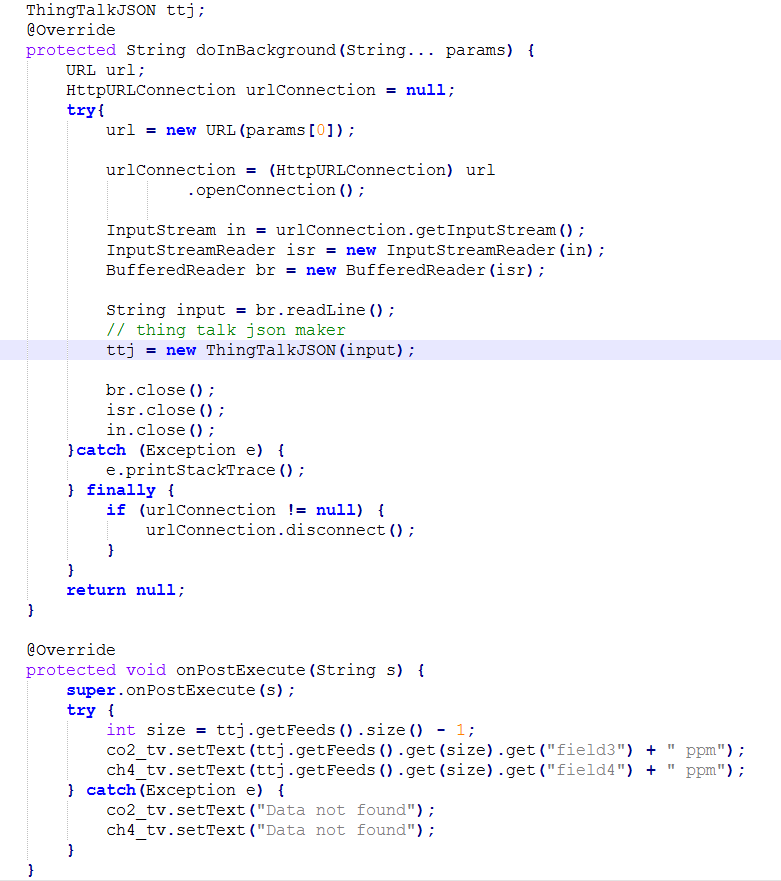
### ارتباط با سرور

برای ارتباط با سرور در اندروید یکی از راه‌ها استفاده از AsyncTask و HTTP است.

AsyncTask دارای چند تابع مهم است که عبارتند از:

* doInBackground: که در یک thread دیگر و در پشت UI اجرا می‌شود و در این قسمت connection را برقرار می‌کنیم.
* onPreExecution: این تابع در UI Thread و قبل از doInBackground صدا زده می‌شود و به عناصر صفحه دسترسی دارد.
* onPostExecution: این تابع در UI Thread و بعد از doInBackground صدا زده می‌شود و به عناصر صفحه دسترسی دارد.

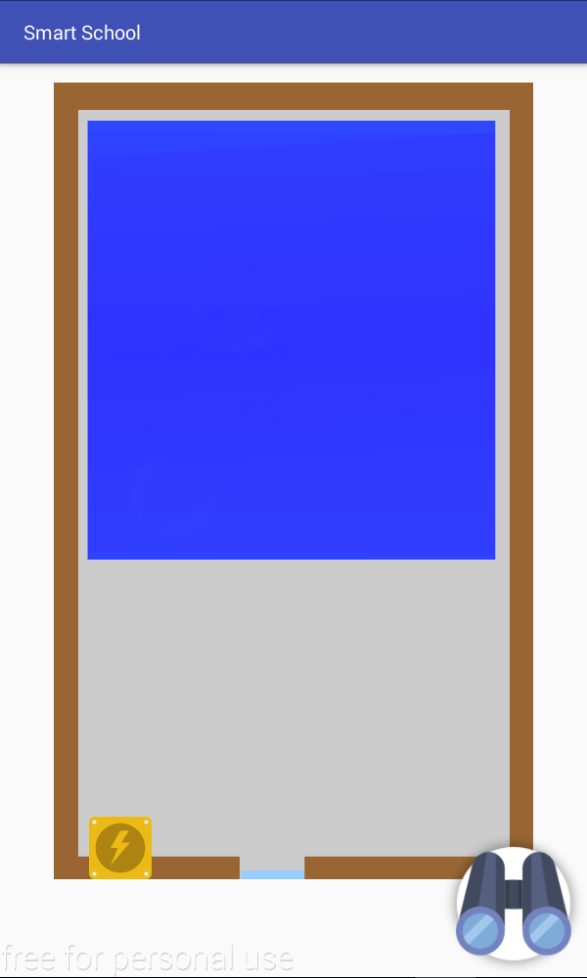
یک نمونه از کد برای این کار که در اپلیکیشن استفاده شده است.



شکل 5 نمونه کد برای ارتباط با سرور

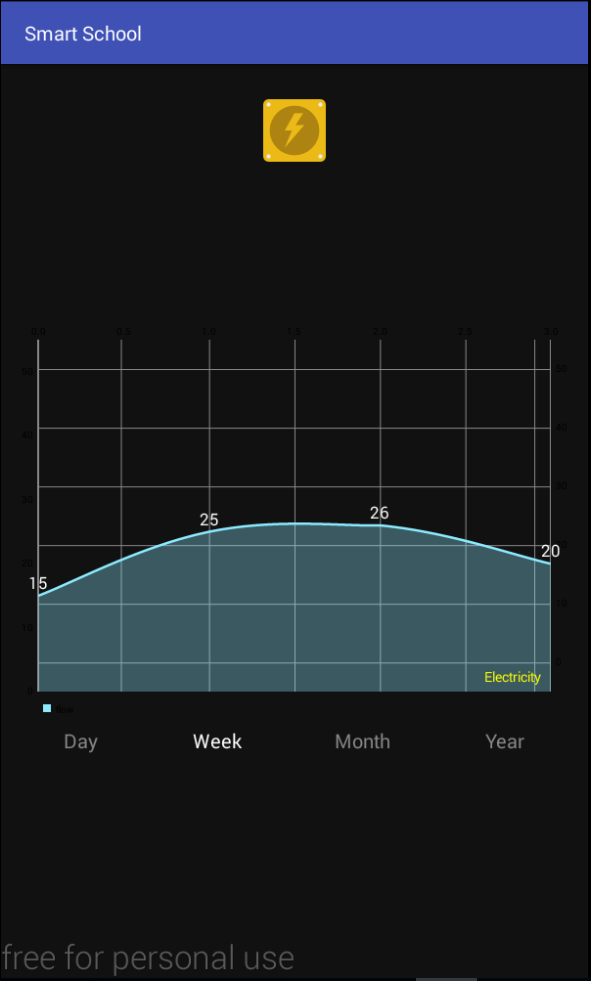
### نحوه کارکردن اپلیکیشن

در اولین صفحه یک نمای پرنده‌ای (bird view) از مدرسه وجود دارد که با انتخاب کنتور به صفحه‌ی مربوطه به کنتور می‌رویم و با انتخاب دوربین قابلیت جابه‌جایی در بین نماهای مختلف مدرسه را بدست می‌آوریم.



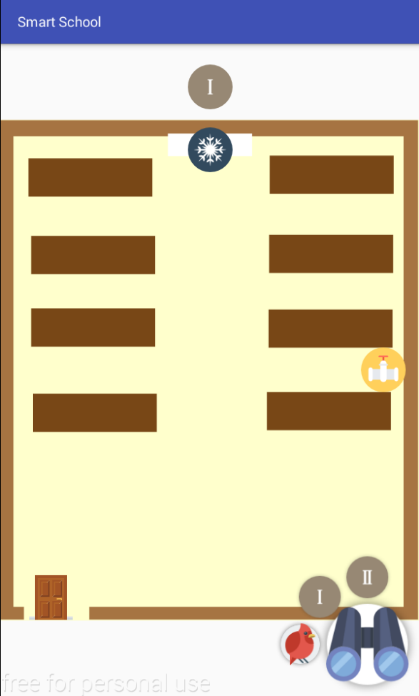
شکل 6 نمای مدرسه

در صفحه‌ی کنتور یک نمودار نشان داده می‌شود که با لمس بازه‌های مختلفی که وجود دارد می‌توان آن بازه زمانی را انتخاب کرد و نمودار مربوط به آن بازه را دید.



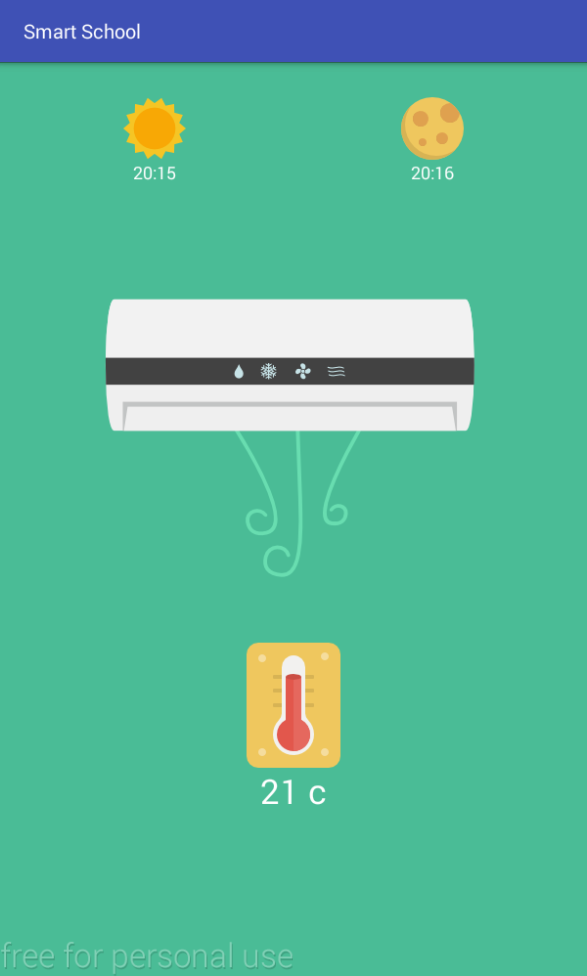
شکل 7 صفحه‌ی کنتور

در این اپلیکیشن با انخاب دوربین و شماره کلاس می‌توان به کلاس‌های مختلف رفت.



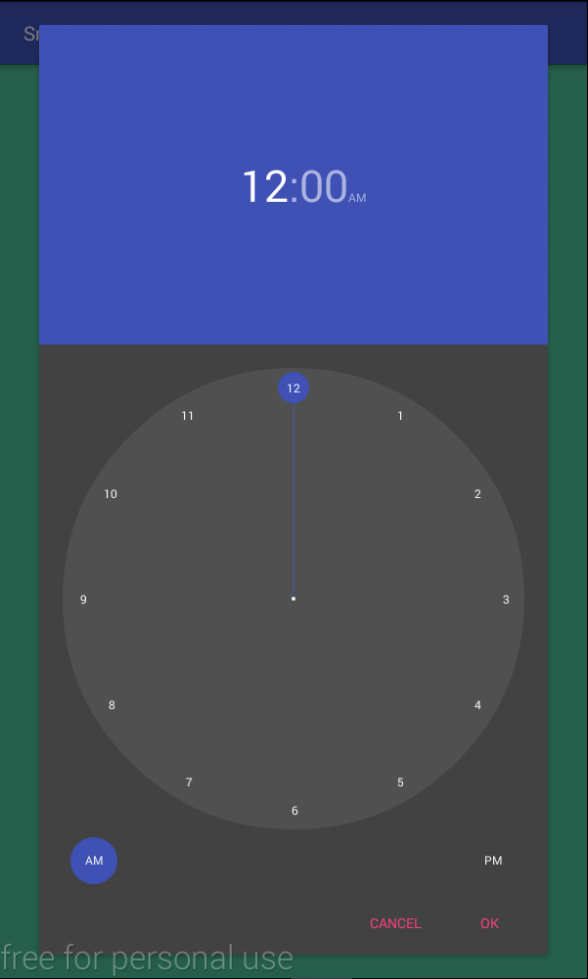
شکل 8 نمای کلاس

با انتخاب کولر می‌توان به صفحه‌ی کولر رفت و دمای کلاس را دید.



شکل 9 صفحه‌ی کولر

همچنین با انتخاب خورشید یا ماه می‌توان زمان شروع و اتمام کولر را تعیین کرد.

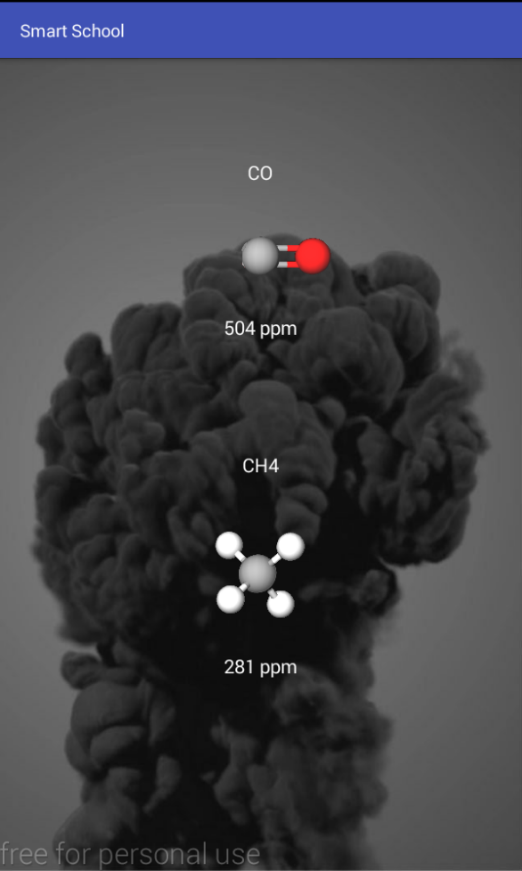


شکل 10 صفحه تنظیمات کولر

در صفحه‌ی کلاس با انتخاب در و انتخاب لوله گاز می‌توان به صفحات مربوطه آن‌ها رفت و اطلاعات آن‌ها را مشاهده کرد.



شکل 11 صفحه‌ی در



شکل 12 صفحه‌ی گاز

## سخت افزار

### مقدمه

یکی از مهم ترین بخش های کار در اینترنت اشیا خواندن داده ها و اطلاعات موجود در دنیای واقعی و منتقل کردن آن به فضای مجازی و اینترنت است. برای خواندن اطلاعات از دنیای واقعی نیاز به استفاده از ماژول ها و سنسورهایی است که از طریق سخت افزار و مدارهای طراحی شده درون خود اطلاعات را ثبت کنند. سپس با اتصال سنسورها و ماژول های مذکور به بردهایی که حاوی پردازشگر هستند می توان داده ها را پردازش کرد و به اطلاعات مفید و مورد نیاز دست پیدا کرد. در پروژه ی ما برد پردازشگر ما برد Raspberry PI 3 بود که بر روی آن سیستم عامل لینوکس Raspbian که مخصوص همین پردازشگر است نصب شده و در آن سیستم عامل داده های مربوط به سنسورهای مختلف را خوانده و پردازش می کنیم. برد رزبری علاوه بر پردازشگر بر روی خود تعدادی pin دارد که قابلیت تنظیم به عنوان ورودی یا خروجی دارند و با استفاده از آنها می توان با سنسورها و ماژولهای مختلف ارتباط برقرار کرد. تعدادی از Pinها نیز به طور پیش فرض برای کارهای پراستفاده رزرو شده اند، برای مثال تعدادی از pinها حاوی برق 5 یا 3.3 ولت به عنوان منبع تغذیه ی سنسورها استفاده می شوند و همچنین برخی دیگر به عنوان Ground رزرو شده اند و ... .

### نحوه راه اندازی کلی ماژول ها

برای راه اندازی هر ماژول نیاز به جستجو در اینترنت و پیدا کردن نحوه ی کار آن است. زبان برنامه نویسی مورد استفاده ی ما که از طریق آن برد رزبری و سنسورها را کنترل می کردیم زبان پایتون بوده است که به تناسب هر سنسور از کتابخانه های مختلف این زبان در کد نوشته شده استفاده شده است.

سنسورهای مورد استفاده در پروژه ی ما به دو دسته تقسیم می شوند. تعدادی از آن ها حاوی خروجی دیجیتال و تعدادی دیگر حاوی خروجی آنالوگ هستند. برای کار کردن با سنسورهای حاوی خروجی دیجیتال کافی است مدار مربوط به هر سنسور را بست و پورت خروجی سنسور را به یکی از pinهای برد رزبری که از پیش رزرو نشده است متصل کرد. سپس در کد مربوط به راه اندازی آن Pin مذکور را به عنوان ورودی برد رزبری تنظیم کرده و با استفاده از کتابخانه های مربوط به سنسور مورد استفاده داده ها را می خوانیم و پردازش های مورد نظر را انجام می دهیم.

برای کار کردن با سنسورهای حاوی خروجی آنالوگ معمولا نیاز به استفاده از ماژول دیگری به نام adc یا analog to digital convertor است. وظیفه ی این ماژول خواندن اطلاعات آنالوگ و بافر کردن آنها و تبدیل آن به داده ی دیجیتال است که این داده ی دیجیتال قابلیت خواندن توسط pinهای برد رزبری را داراست ( تمام Pinهای رزبری دیجیتال هستند و به همین دلیل نمی توان خروجی آنالوگ را به طور مستقیم به آن ها متصل کرد و داده خواند ). در زیر مدار مربوط به چگونگی راه اندازی adc ( MCP3008 ) و در ادامه تکه کد تابع خواندن از adc به زبان پایتون آورده می شود:

MCP3008 VDD to Raspberry Pi 3.3V

MCP3008 VREF to Raspberry Pi 3.3V

MCP3008 AGND to Raspberry Pi GND

MCP3008 DGND to Raspberry Pi GND

MCP3008 CLK to Raspberry Pi pin 18

MCP3008 DOUT to Raspberry Pi pin 23

MCP3008 DIN to Raspberry Pi pin 24

MCP3008 CS/SHDN to Raspberry Pi pin 25

Adc:

# read SPI data from MCP3008 chip, 8 possible adc's (0 thru 7)

def readadc(adcnum, clockpin, mosipin, misopin, cspin):

if ( (adcnum > 7) or (adcnum < 0) ):

return -1

GPIO.output(cspin, True)

GPIO.output(clockpin, False) # start clock low

GPIO.output(cspin, False) # bring CS low

commandout = adcnum

commandout |= 0x18 # start bit + single-ended bit

commandout <<= 3 # we only need to send 5 bits here

for i in range(5):

#print ("commandout",commandout)

if (commandout & 0x80):

GPIO.output(mosipin, True)

else:

GPIO.output(mosipin, False)

commandout <<= 1

GPIO.output(clockpin, True)

GPIO.output(clockpin, False)

adcout = 0

# read in one empty bit, one null bit and 10 ADC bits

for i in range(12):

#time.sleep(0.1)

GPIO.output(clockpin, True)

GPIO.output(clockpin, False)

adcout <<= 1

#print ("misopin",GPIO.input(misopin))

#if (GPIO.input(misopin)):

input = GPIO.input(misopin)

#print i,"->",input

if(input):

adcout |= 0x1

GPIO.output(cspin, True)

adcout >>= 1 # first bit is 'null' so drop it

return adcout

# change these as desired - they're the pins connected from the

# SPI port on the ADC to the Raspberry Pi</h1>

SPICLK = 18

SPIMISO = 23

SPIMOSI = 24

SPICS = 25

GPIO.setwarnings(False)

#set up the SPI interface pins

GPIO.setup(SPIMOSI, GPIO.OUT)

GPIO.setup(SPIMISO, GPIO.IN)

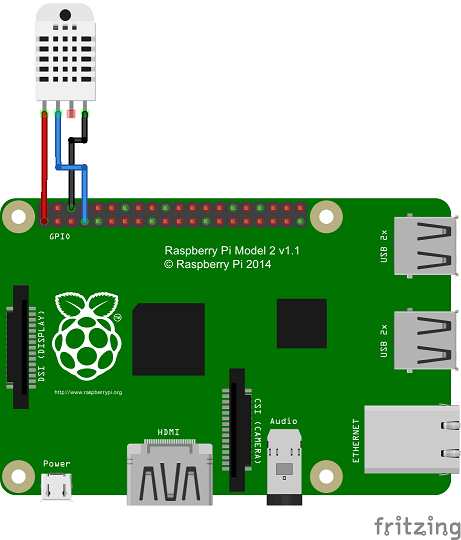
GPIO.setup(SPICLK, GPIO.OUT)

GPIO.setup(SPICS, GPIO.OUT)

### مستندات مربوط به راه اندازی تک تک ماژول ها

#### سنسور DHT11

در بخش سخت افزار پروژه ی مدرسه ی هوشمند ماژول های مختلفی در بخش سخت افزار پیاده سازی شده و با واسطه ی سرور ( و ارسال اطلاعات بر روی سرور Thingtalk.ir ) با اپلیکیشن ارتباط برقرار کردند. از جمله ی این ماژول ها می توان به ماژول DHT11 برای اندازه گیری دمای هوا اشاره کرد که خروجی آن دیجیتال بوده و نمونه ی شکل آن و کد آن در ادامه آورده می شود:



شکل 13 مدار سنسور DHT11

import RPi.GPIO as GPIO

import dht11

import time

import datetime

import requests

import json

# initialize GPIO

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)

GPIO.cleanup()

# read data using pin 14

instance = dht11.DHT11(pin=14)

coolerstatus = False

GPIO.setup(16, GPIO.OUT)

while True:

result = instance.read()

if result.is\_valid():

#send data to server

j = requests.get("http://thingtalk.ir/channels/117/feed.json?key=7RU9LTSI8X4E8Y0G")

past = j.json()

x = past["feeds"][len(past["feeds"])-1]

sttime = x["field1"]

sclk, smin = (int(i) for i in sttime.split(':', 1))

endtime = x["field2"]

eclk, emin = (int(i) for i in endtime.split(':', 1))

#cast server time to local time

currenttime = x["created\_at"]

currenttime = datetime.datetime.strptime(currenttime, "%Y-%m-%dT%H:%M:%SZ")

currenttime = currenttime + datetime.timedelta(hours=3, minutes=30)

curmin = currenttime.time().minute

curclk = currenttime.time().hour

#checking on & off cooler time with local time

if((curclk<sclk)or((curclk == sclk) and (curmin<smin))):

coolerstatus = False

elif((curclk>eclk)or((curclk == eclk) and (curmin > emin))):

coolerstatus = False

else:

coolerstatus = True

GPIO.output(16, not(coolerstatus))

#send data to server

s = "http://thingtalk.ir/update?key=7RU9LTSI8X4E8Y0G"

for i in range(1,3):

s += "&field"+str(i)+"="+x["field"+str(i)]

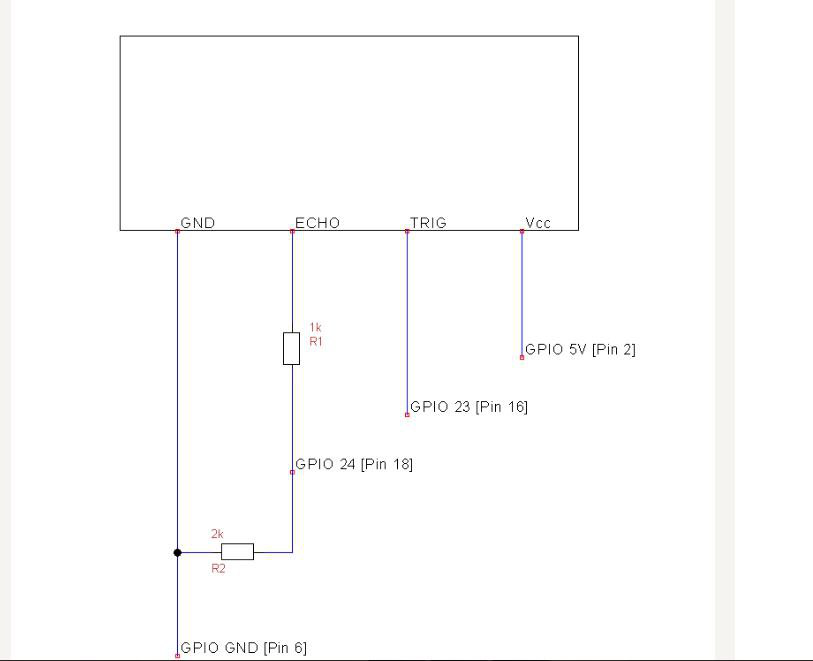
s += "&field3="+str(result.temperature)

requests.get(s)

time.sleep(1)

#### سنسور HY-SRF05

از دیگر سنسورهای راه اندازی شده می توان به سنسورهای Ultrasonic hy-srf05 برای سنجش فاصله اشاره کرد که در پروژه ی ما کاربرد این سنسورها تشخیص ورود و خروج دانش آموزان به کلاس بوده است که در زیر کد آن و شکل بستن مدار یک سنسور ultrasonic آورده می شود. در پروژه ما از دو سنسور اولتراسونیک در کنار هم استفاده شده است که یکی داخل کلاس و دیگری بیرون کلاس در ابتدای در کار گذاشته می شود و با توجه به اینکه ابتدا کدام سنسور تشخیص عبور دانش آموز را می دهد می توان تعداد دانش آموزان داخل کلاس را کم و زیاد کرد.



شکل 14 مدار hy-srf05

import RPi.GPIO as GPIO

import time

import threading

import datetime

from gpiozero import MotionSensor

import requests

import json

GPIO.setmode(GPIO.BCM)

GPIO.setup(16, GPIO.OUT)

ipir = MotionSensor(4)

GPIO.setup(27,GPIO.OUT)

GPIO.setup(22,GPIO.IN)

GPIO.setup(23,GPIO.OUT)

GPIO.setup(24,GPIO.IN)

humans = 0

count = [0, 0]

mo = False #lamp status

def sendtoserver():

j = requests.get("http://thingtalk.ir/channels/118/feed.json?key=CSL9Y4UV0XPJSOB7")

past = j.json()

s = "http://thingtalk.ir/update?key=CSL9Y4UV0XPJSOB7"

s += "&field1="

s += str(humans)

s += "&field2="

s += str(int(mo))

if(past["feeds"] != []):

x = past["feeds"][len(past["feeds"])-1]

for i in range(3,5):

s += "&field"+str(i)+"="+x["field"+str(i)]

else:

s += "&field3=0&field4=0"

requests.get(s)

def ultra( threadName, TRIG\_04, ECHO\_04 ):

global humans

global count

print ("humans: ", humans)

i = 0

defa = 0;

start = False

pre = 0

while True:

GPIO.output(TRIG\_04, False)

time.sleep(0.2)

GPIO.output(TRIG\_04, True)

time.sleep(0.00001)

GPIO.output(TRIG\_04, False)

while GPIO.input(ECHO\_04)==0:

pulse\_start = time.time()

while GPIO.input(ECHO\_04)==1:

pulse\_end = time.time()

pulse\_duration = pulse\_end - pulse\_start

distance\_04 = pulse\_duration \* 17150

distance\_04 = round(distance\_04, 2)

#detecting human

if((pre - distance\_04) > 20):

if(start):

if((defa-distance\_04) > 20):

count[int(threadName)] += 1

if(count[0] == count[1]):

if(threadName == "0"):

humans -= 1

if(humans < 0):

humans = 0

count = [0,0]

print("-1")

else:

humans += 1

print("+1")

sendtoserver()

pre = distance\_04

#understand default distance

if(i < 10):

defa += distance\_04

elif(i == 10):

start = True

defa /= 10

pre = defa

i += 1

try:

threading.Thread(target=ultra, args=("0", 23, 24, )).start()

threading.Thread(target=ultra, args=("1", 27, 22, )).start()

except:

print ("Error: unable to start thread")

i = 0

try:

while True:

if ((ipir.motion\_detected) or (humans>0)):

i = 0

if(mo == False):

mo = True

sendtoserver()

GPIO.output(16, True)

if(humans <= 0):

print(datetime.datetime.now().time(), "Motion detected!")

if((i == 17)):

i = 0

if(humans == 0):

GPIO.output(16, False)

if(mo):

mo = False

sendtoserver()

i += 1

print(count)

time.sleep(0.3)

except KeyboardInterrupt:

print("Errr",humans,"\n\n\n\n\n\n")

GPIO.cleanup()

#### سنسورهای MQ

*ماژول های MQ برای اندازه گیری آلودگی هوا و مقدار ppm موجود از هر گاز در فضای اتاق استفاده می شوند که ما از MQ4 و MQ7 برای تشخیص CH4 و CO در فضای اتاق استفاده کردیم. MQ حاوی خروجی آنالوگ است و به همین دلیل باید توسط adc به رزبری متصل شود و داده های مربوط به آن را خواند که در ادامه کد مربوط به این خواندن آورده می شود. لازم به ذکر است که سنسورهای MQ نیاز به کالیبره شدن دارند تا مقدار آن ها به مقدار ppm تبدیل شود که در پروژه ی ما کالیبره شدن رخ نداده است و داده ی خام بر روی سرور ارسال می شود.*

#!/usr/bin/env python

import time

import RPi.GPIO as GPIO

import urllib2

import math

import json

GPIO.setmode(GPIO.BCM)

###

GPIO.setup(5, GPIO.OUT)

try:

while True:

#read the analog pin

analog\_value = readadc(pulse\_adc, SPICLK, SPIMOSI, SPIMISO, SPICS)

analog\_value2 = readadc(1, SPICLK, SPIMOSI, SPIMISO, SPICS)

print (analog\_value, analog\_value2)

#send to channel

j = urllib2.urlopen("http://thingtalk.ir/channels/118/feed.json?key=CSL9Y4UV0XPJSOB7").read()

past = json.loads(j)

x = past["feeds"][len(past["feeds"])-1]

s = "http://thingtalk.ir/update?key=CSL9Y4UV0XPJSOB7"

for i in range(1,3):

s += "&field"+str(i)+"="+x["field"+str(i)]

s += "&field3="

s += str(analog\_value)

s += "&field4="

s += str(analog\_value2)

urllib2.urlopen(s)

print ("sent")

#hang out and do nothing for a tenth of a second

time.sleep(1)

except KeyboardInterrupt:

GPIO.cleanup()

#### سنسور acs712 5A

*این ماژول برای اندازه گیری جریان برق مدرسه ی هوشمند بررسی شده است که تا جریان 5 آمپر را تشخیص می دهد. برای راه اندازی به دلیل خروجی آنالوگ آن باید آن را به adc وصل کرد منتهی با تغییر کوچکی در مدار که در زیر ادامه ذکر می شود. ابتدا VDD و Vref، adc را به منبع تغذیه 5 ولت رزبری متصل کرده و سپس مدار Dout را به شکل زیر می بندیم:*

DO (from ADC) ----> 2k4 resistor ----> MISO on Pi ----> 4k7 resistor ----> GND

*در ادامه کد مربوط به راه اندازی این ماژول آورده می شود ( با توجه به ضرایبی که در نحوه ی کار با این ماژول آورده شده است مقدار adc به mA تبدیل شده و چاپ می شود ):*

#!/usr/bin/env python

import time

import RPi.GPIO as GPIO

import urllib2

import math

GPIO.setmode(GPIO.BCM)

###

GPIO.setup(5, GPIO.OUT)

factor= 185.

count = 0

try:

while True:

#read the analog pin

analog\_value = readadc(pulse\_adc, SPICLK, SPIMOSI, SPIMISO, SPICS)

tmp = analog\_value-512

print ("current:", ((tmp)/factor))

count += 1

#send to channel

s = "http://www.thingtalk.ir/update?key=LY1NL1IP5EHQPLD7&field1="

s +=str(analog\_value)

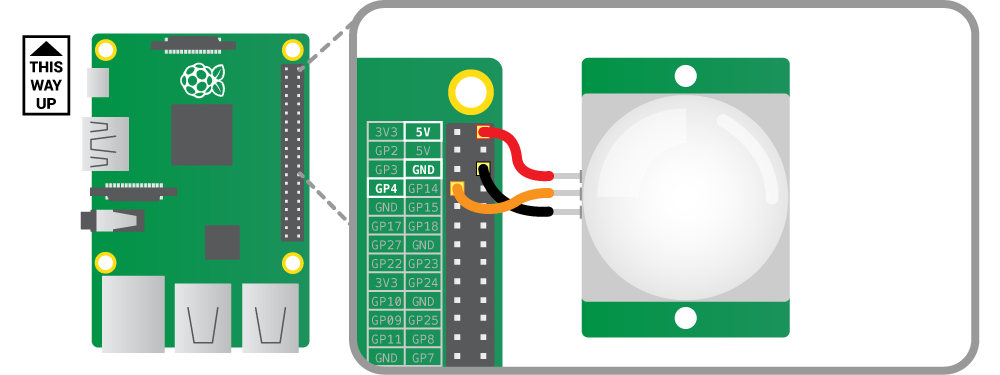
time.sleep(0.2)

except KeyboardInterrupt:

pass

#### سنسور pir

*با استفاده از این سنسور می توان حرکت را داخل اتاق تشخیص داد و در صورت وجود دانش آموز در کلاس لامپ درون اتاق را روشن کرده ( این کار با استفاده از ارسال وضعیت روشن به اپلیکیشن اندروید انجام می شود ). در ادامه شکل مدار و کد مربوط به این ماژول آورده می شود.*



شکل 15 سنسور pir motion

from gpiozero import MotionSensor

import datetime

import time

import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)

GPIO.setup(16, GPIO.OUT)

pir = MotionSensor(4)

i = 0

try:

while True:

if pir.motion\_detected:

i = 0

GPIO.output(16, True)

print(datetime.datetime.now().time(), "Motion detected!")

if(i == 50):

GPIO.output(16, False)

i += 1

time.sleep(0.1)

except KeyboardInterrupt:

GPIO.cleanup()

#### ماژول رله

برای راه اندازی و روشن سازی کولر و تنظیم دمای اتاق های مدرسه از راه دور با توجه به زمان تنظیم شده در اپلیکیشن رله ای در قسمت سخت افزار فعال و غیر فعال شده است که نماینده ی ارسال سیگنال به کولر است. کد مربوط به تشخیص زمان و روشن و خاموش کردن رله در کد مربوط به سنسور DHT در بالاتر آورده شده است.

پیوست‌ها