



### ke stavebnici odjezdové tabule





Přijdeš na to, ze které pražské zastávky jsou odjezdy zobrazené na tabuli na obrázku? Správnou odpověď najdeš na straně 22.

### Co z toho bude?

Mám radost, že ses odhodlal(a) k sestavení této odjezdové tabule, která bude ukazovat odjezdy spojů MHD ze zastávky nebo zastávek, které si zvolíš. Velmi praktické je to, že umí zobrazovat i aktuální zpoždění, takže už nebudeš chodit na zastávku zbytečně brzo.

Zní to jednoduše, že? Opravdu na tom nic komplikovaného není. Ale já jsem se i tak při vymýšlení a sestavování dozvěděl a naučil leccos nového. A rád bych se s tebou prostřednictvím tohoto návodu podělil o to, co mě zaujalo. Proto jsem sem zařadil i žluté rámečky. Ty obsahují informace, které k sestavení nutně nepotřebuješ, ale dozvíš se trochu víc o tom, jak to celé funguje. A rovnou si to můžeme vyzkoušet na prvním rámečku:

#### Odkud se berou data?

Vozidla Pražské integrované dopravy jsou vybavena zařízeními, která sledují jejich aktuální polohu a průběžně odesílají tyto informace na dispečink. Odtud data doputují do datové platformy Golemio, kde se porovnají s jízdními řády a celkově upraví do čitelné podoby. Golemio je projekt Hlavního města Prahy a jedná se o službu, která sdružuje nejrůznější data, která se života ve městě týkají – kromě polohy vozů MHD bychom tam našli třeba údaje o kvalitě ovzduší nebo o naplněnosti popelnic. Golemio data nejen prezentuje, ale poskytuje také *API (Application Programming Interface)* – tedy rozhraní, přes které si mohou data načítat jiné aplikace. V případě informací o poloze vozů to mohou být třeba odjezdové tabule na zastávkách MHD nebo také naše "domácí, tabule.



### Co si mám připravit?

K sestavení budeš potřebovat:

- Především tento návod, ale to je snad jasné.
- Počítač nebo telefon s přístupem k internetu a možností připojení přes WiFi.
- Heslo k tvojí domácí WiFi síti.

A k tomu vše, co je součástí stavebnice:



### Copak to tu máme?

Začneme tím, že si ze všeho nejdřív pořádně prohlédneme všechny součástky, a přitom je budeme postupně montovat do krabičky.



Krabička, do které za chvíli schováme všechny součástky, je vyrobena metodou 3D tisku. Předpokládám, že 3D tisk představovat nemusím, ale pokud to není tvůj denní chleba, můžeš si v rámečku na následující stránce přečíst, jak se dají tisknout vícebarevné výrobky.

#### **Printables**

Na internetu se dá najít mnoho různých databází 3D modelů. Jednou z nich je server **Printables.com**, který provozuje český výrobce 3D tiskáren Josef Průša. Odtud si můžeš volně stáhnout i krabičku pro toto zařízení (odkaz najdeš na www.timion.cz), takže si ji můžeš sám(a) vytisknout, pokud máš k dispozici 3D tiskárnu. Navíc můžeš stažený model upravit podle sebe a ještě ho vylepšit! V takovém případě prosím nezapomeň nahrát svojí verzi zpátky do databáze jako remix, aby tvoje nápady mohli využít i ostatní uživatelé.



#### Jak 3D tisknout barevně?

Nejrozšířenější metodou 3D tisku (alespoň pro "domácí" použití) je *FMD (Fused Desposition Modeling)*, kdy tryska nanáší roztavený *filament* (strunu ze speciálního plastu) postupně odspoda jednu vrstvu na druhou, až vznikne trojrozměrný výrobek. V takovém případě existují v zásadě dva způsoby, jak docílit vícebarevného výrobku.

Buďto je tiskárna vybavena zařízením, které umí samo v průběhu tisku jedné vrstvy používat různé filamenty, což je velmi pohodlné. Nevýhodou může být, že změny filamentu tisk zdržují a také často vzniká nemalé množství odpadu, protože při každé výměně je potřeba vytlačit nějaké množství filamentu "na prázdno, než se tryska kompletně vyčistí od původní barvy. Některé tiskány ale dokáží v průběhu tisku střídat trysky, čímž tyto problémy odpadají.

Ale ani tiskárna bez možnosti tisknout více filamenty najednou není odkázaná jen na jednobarevné výtisky. Stačí na to myslet při přípravě modelu. Pak se dá předem nastavit, kdy se má tisk přerušit, a v tu chvíli filament vyměnit ručně. Nevýhodou oproti prvnímu řešení je, že zpravidla není možné filament vyměňovat v rámci tisku jedné vrstvy.



První součástka, kterou do krabičky přidáme vlastně sama o sobě nic nedělá.



Slouží totiž jen k tomu, že propojuje vodiče zapojené do obou řad. Bude nám sloužit podobně jako elektrická "rozdvojka", protože elektrickým proudem budeme potřebovat napájet více součástek.

Destičku přišroubuj pomocí **dvou menších** šroubků na místo u horní strany krabičky, jak je vidět na obrázku. Se správným otočením si nelam hlavu, je to úplně jedno.





Nemáš-li extra šikovné prsty, budeš možná bojovat s tím, že v krabičce není příliš místa pro přidržení šroubku na místě. S tím pomůže magnetický šroubovák, případně se mi osvědčilo prostrčit nejdříve šroubky dírkami součástky, kterou chci přišroubovat, a teprve potom jí dát na místo spolu s připravenými šroubky. Na šrouby se při šroubování neboj zatlačit silou.

### | 7 |

Největší a nejviditelnější částí našeho zařízení je samozřejmě LCD Displej, který slouží pro zobrazení výsledků.



Používáme *grafický* displej, což znamená, že můžeme "rozsvítit" každý z jednotlivých bodů, ze kterých je displej složený. (Je jich 128 v každém řádku a 64 v každém sloupci.) To je pro nás výhodné oproti tzv. *alfanumerickým* displejům, které umí zobrazovat jen určitý počet písmen nebo číslic, protože můžeme použít menší písmo a díky tomu nacpat na displej více informací.

#### Co to znamená LCD?

LCD je zkratka pro *liquid crystal display,* tedy displej z tekutých krystalů. U klasických krystalů jsou molekuly – stavební kameny, ze kterých jsou složené všechny látky – pevně umístěné a nemohou se ani hnout. Naopak u tekutiny jsou molekuly volné a pohybují se. No a tekutý krystal je něco mezi tím – molekuly jsou zde sice uspořádané do nějakého útvaru, ale pokud dodržují toto uspořádání, mohou se pohybovat, například se otáčet.

Tekuté krystaly jsou v displeji sevřené mezi dvěma sklíčky. Zezadu na ně svítí světlo přes polarizační vrstvu, která nechá projít jen vlny určité orientace. Tím pádem záleží na natočení molekul v krystalu, jestli přes něj světlo projde nebo ne. Představ si to, jako když otáčíš žaluzie na okně – když jsou zavřené, světlo neprochází; když jsou kolmo na okno, světlo přes ně projde. V případě krystalů můžeme ovládat změnu orientace molekul pomocí elektrického proudu. A tak v displeji elektrickými signály otevíráme nebo zavíráme pomyslné žaluzie na každém z malinkatých okének, které pak tvoří světelné body na displeji.

Pro přišroubování displeje využiješ čtyři větší šroubky. Nelekej se, jestli zadní strana tvého displeje vypadá trošku jinak než na obrázku. Důležité je, aby lišta s kolíčky pro připojení byla u horního okraje krabičky.





Srdcem – nebo spíš mozkem – zařízení je modul Wemos (Lolin) D1 Mini.



Správně bychom ho asi měli označit jako *protypovací desku*, což znamená, že už je zde připraveno všechno, co se může hodit pro rychlé a snadné testování nových nápadů: například převodník USB pro programování přes počítač, digitální i analogové vstupy a výstupy a tak podobně. A to se nám hodí. Vstupy a výstupy jsou připravené k použití v podobě pinů – kovových kolíčků, přes které budeme připojovat další součástky.

Řídicí deska patří do pravého dolního rohu, k přišroubování slouží zbylé dva malé šroubky.

#### ESP8266

Modul D1 Mini obsahuje mikrokontrolér ESP8266, což je vlastně miniaturní počítač v jednom integrovaném obvodu, který obsahuje všechny součásti klasického počítače jako je procesor, paměť a rozhraní pro připojení dalších zařízení. A čip ESP8266 je skvělý v tom, že už je zde integrovaný i WiFi modul, takže se přes něj můžeme snadno připojit k internetu.

K našemu zařízení budeme připojovat také několik *senzorů*, tedy součástek, které nám budou dodávat informace o tom, co se děje v okolním světě. Prvním z nich je **světelný senzor**.



Jak už název napovídá, tato součástka hlídá, kolik je v okolí světla. K čemu to využijeme? Budeme předpokládat, že když je okolo naší tabule tma, odjezdy nejspíš nikoho nezajímají a svícení displeje by naopak mohlo být otravné. Při určitém množství světla se tedy zařízení "uspí". Citlivost na světlo je možné individuálně přizpůsobit, ale k tomu se dostaneme až později.

#### Jak funguje fotorezistor?

Ano, naše světelné čidlo se správně jmenuje *fotorezistor*, někdy také *LDR (Light dependent resistor)*. Rezistor je (velmi zjednodušeně) součástka, která v elektrickém obvodu klade elektrickému proudu určitý odpor, takže přes ní protéká hůř nebo vůbec. A v tomto případě se odpor mění podle toho, kolik na něj dopadá světla. Zhruba to funguje tak, dopadající částečky světla zvané *fotony* předávají svoji energii elektronům obsaženým v polovodiči. Tím se elektrony uvolní, mohou sloužit pro přenos elektrického proudu, což zvyšuje vodivost rezistoru. Takže čím více je dopadajícího světla, tím lépe fotorezistor propouští elektrický proud, který pak můžeme změřit. A tím vlastně měříme světlo.

.....

# |11|

Pokud není "hlavička" fotoresistoru správně nasměrovaná, bude jí potřeba nejprvě opatrně ohnout tak, aby směřovala do připraveného otvoru v krabičce. Potom ji provlékni dírou ve stěně krabičky, aby koukala ven. Nakonec celý modul připevni **posledním větším** šroubkem.



Z boku by to mělo vypadat asi takto:



Druhým senzorem je **pohybový senzor**, kterým budeme detekovat, jestli se zrovna někdo pohybuje kolem naší odjezdové tabule, nebo ne.



K čemu nám to bude? Jednak budeme vypínat podsvícení displeje, když zrovna nikdo nebude poblíž, abychom zbytečně neplýtvali energií. A po nějaké době bez provozu okolo zařízení ho uspíme, takže nebudeme zbytečně stahovat informace o odjezdech, když zrovna nikoho nezajímají. Ale samozřejmě budeme pořád kontrolovat, jestli se někdo nepřiblížil. A pokud ano, rychle zase všechno nastartujeme.

#### Jak funguje PIR?

V našem případě používáme pro sledování pohybu okolo zařízení čidlo označované jako *PIR*, což je zkratka pro *Passive Infra-Red*. Proč pasivní? Protože narozdíl od jiných senzorů toto zařízení samo aktivně nic nevysílá, jen přijímá. A asi nemusím ani psát, že přijímá infračervené záření, protože to už také víš z jeho názvu. Infračervené záření vysílá do svého okolí každý objekt, který má nějakou teplotu, tedy i lidé. Toto čidlo je samozřejmě nastavené tak, aby zachytilo právě tu část záření, která odpovídá teplotě člověka.

A proč má součástka podobu bílé kopule? Jedná se o *Fresneovu čočku*, která pomáhá směřovat záření ze širokého okolí a koncentrovat ho do samotného senzoru, který je umístěný pod ní. Díky tomu má toto čidlo široký úhel snímání, což znamená, že dobře "vidí" nejen přímo před sebe, ale i do stran.

Z pohybového čidla nejdříve sundej bílou čočku (všechny bílé plastové části), kterou napasuj do díry z přední strany krabičky. Schválně je to navržené tak, že to jde ztuha, aby senzor nevypadával. Pokud se ti ale nedaří tam bílou část nacpat, můžeš díru trochu obrousit šmirgl papírem. (Naopak pokud by to náhodou šlo příliš lehce, bude potřeba pod čočku kápnout trošku lepidla.) Všimni si, že na jedné straně čočky je malá rýha. Ta by měla směřovat do strany směrem ke středu krabičky.



Následně z opačné strany (z vnitřku krabičky) zatlač pohybový senzor do bílé čočky. Na senzoru je malý výstupek, který musí zapadnout do té rýhy, kterou jsi před chvílí hledal(a). Výstupek tedy musí také směřovat ke středu krabičky.



# |14|

### Prosím zapojit!

Teď nás čeká úkol propojit jednotlivé součástky dohromady. Naštěstí je to velmi jednoduché – stačí zástrčky na konci vodičů zasunout na kovové kolíčky jednotlivých součástek. Je potřeba mít šikovné prsty a občas jde nasazení trochu ztuha, ale určitě to hravě zvládneš.

Pokud jsou kabílky spojené dohromady, opatrně je od sebe odtrhni. Při zapojování nezáleží, kterou barvu vodiče použiješ, barvy nemusí odpovídat těm na schématech.





Nejspíš sis všiml(a), že v balení máš dvě délky vodičů. Na schématech jsou rozlišené plnou a přerušovanou čárou, tak to nepopleť.

Nejdříve si připravíme naši "rozdvojku". Na spodní řadu zapojíme z řídicí desky napětí, pin je označený jako **3V3**. Horní řada bude sloužit jako společná zem, takže jí propojíme s pinem označeným jako **GND**.



#### Elektrický obvod

Jednotlivé součástky našeho zařízení jsou vlastně elektrické spotřebiče. Aby jimi mohl protékat elektrický proud, musíme je připojit do uzavřeného elektrického obvodu, proto připojujeme vždy dva vodiče. Z jedné strany připojujeme zdroj napětí, který bývá označen jako *VCC (Voltage Common Collector)* nebo přímo pomocí hodnoty napětí (na desce máme zdroje napětí 5 a 3,3 voltů). Na druhé straně obvodu bývá označení *GND (Ground)*, tedy zem. Proč zrovna zem?

Napětí vyjadřuje rozdíl v elektrickém potenciálu mezi dvěma body. Můžeš si to představit, jako když měříme výšku hory – také musíme udat, vzhledem k čemu je výška vztažená, jinak by to nedávalo smysl. U hor se jako vztažná hodnota využívá hladina moře. U elektrického potenciálu se jako nulová hodnota často používá naše Země. Odtud pochází označení GND.

V praxi nemusí být každý obvod vždy spojený se zemí, ale je důležité, aby pro všechny součástky, které spolu mají komunikovat, byla neutrální úroveň stejná, aby měly *společnou zem*. Proto postupně všechny *zem*ě propojíme dohromady.

Teď propojíme řídicí desku s **displejem**. Tady je to složitější, protože potřebujeme vodiče pro přenos dat do displeje i pro podsvícení.

Začneme podsvícením, u kterého nám stačí připojit napájení elektřinou. Nezapojíme ho ale přímo na zdroj napětí, místo toho využijeme jeden z digitálních pinů. Ten bude v programu nastaven jako výstupní, což znamená, že na tomto kolíčku můžeme pomocí programu napětí zapínat a vypínat. A to je přesně to, co potřebujeme, abychom mohli podsvícení displeje vypnout ve chvíli, kdy není nikdo poblíž. Na displeji tedy zapoj první kolíček zleva na společnou zem a druhý zleva na pin **D8** řídicí desky.





Na schématu jsou šedou barvou vyznačená propojení, z předešlých kroků, tak se tím nenech zmást.

# |17|

Dále potřebujeme zapojit obvod, který bude zpracovávat data a zobrazovat je na displeji. Začneme opět napájením. První kolíček displeje zprava připoj na společnou zem. Na druhý kolíček zprava přivedeme napětí. Ale pozor, protože displej vyžaduje napájení pěti volty, použijeme tentokrát přímo výstup na desce označený **5V**.



#### Binární logika

Počítače pracují s tzv. *binární logikou*, což neznamená nic jiného, než že rozlišují pouze dvě hodnoty, které se většinou označují jako jedničky a nuly. A obě hodnoty jsou v praxi vyjádřeny pomocí elektrického napětí: logická 0 bývá vyjádřena nulovým napětím a logická jednička bývá vyjádřena přítomností nějakého napětí. Hodnota tohoto napětí může být pro různé čipy různá. Například náš řídicí čip ESP8266 považuje za jedničku napětí 3,3 volty. Řadič displeje ovšem používá pětivoltovou logiku, takže jedničku vyjadřuje pěti volty. Budou si zařízení spolu rozumět, když každé používá jinou hodnotu pro jedničku? V tomto případě naštěstí ano, protože ESP8266 je tzv. *5V tolerantní*. Napětí pro logickou jedničku totiž nemusí být úplně přesné, v praxi se za jedničku považuje vždy určitý rozsah. A v tomto případě se oba rozsahy překrývají, takže to funguje. (Kdyby ne, museli bychom použít *převodník logických úrovní*, který by sloužil jako takový "překladatel".)

Komunikace mezi našim programem a displejem bude probíhat *sériově*. K tomu potřebujeme mezi displejem a deskou zapojit 3 vodiče – udělej to podle schématu. Nakonec musíme displeji oznámit, že budeme komunikovat pomocí sériového rozhraní, a k tomu slouží poslední kolíček z levé šestice. Pro sériovou komunikaci musí být připojen na logickou nulu, což zajistíme tím, že pin propojíme s pinem D1, který je nastavený jako výstupní s nulovým napětím.



#### Jak funguje sériová komunikace?

Sériová komunikace znamená, že signály (tedy data) posíláme postupně za sebou (v sérii) po jednom vodiči, můžeš si to představit trochu jako vysílání morseovky telegrafem, kdy se také vysílá jeden znak za druhým. Proč jsme tedy zapojili 3 vodiče? První slouží pro samotný přenos dat. Aby obě zařízení správně vyhodnotila, kdy se začíná vysílat další znak, musí se domluvit na společném časování a k tomu slouží druhý vodič, který vysílá *hodinové signály*. Na stejné sběrnici může být v některých případech připojeno více zařízení, a pak se třetím vodičem řídí, kdy které zařízení může přenášet data.

Zapojení **fotorezistoru** bude oproti displeji mnohem jednodušší. Především musíme zapojit senzor do elektrického obvodu, tedy jeden vodič na "rozdvojku", kde máme třívoltové napětí a druhý na společnou zem. Tento modul je udělaný tak, že převádí naměřenou intenzitu světla na digitální signál, který má tím pádem jen dvě hodnoty ("světlo" a "tma"). Díky tomu můžeme posledním vodičem senzor připojit na jeden z digitálních pinů nastavených jako vstup, na kterém si program tuto informaci přečte.





Zapojení **pohybového čidla** je obdobné, tak už to nebudeme protahovat a šup tam s tím. Při nasazování kolíčků na čidlo doporučuju čočku ze spodní strany přidržet, abys ji omylem nevytlačil(a) ven.



Uf. Tak snad se vše povedlo – o tom se ostatně přesvědčíme za chvíli. Sice už máme zapojeno, ale víko krabičky ještě nezavírej! Ještě budeme potřebovat mít k součástkám přístup.

Ve skutečnosti to teď nejspíš vypadá přibližně takto, že?



### Jak to budu ovládat?

Chápu, že bys nejraději hned vyzkoušel(a), jestli se ti podařilo všechno zapojit správně, ale ještě chvilku s tím vydrž, prosím! Než odjezdovou tabuli oživíme, uděláme si rychlou odbočku a ukážeme si, jak tvému zařízení zadat to správné nastavení, především z jaké zastávky má odjezdy vlastně zobrazovat.

Na počítači v internetovém prohlížeči zadej adresu https://admin.timion.cz a vytvoř si účet (pomocí odkazu *Vytvořit účet*). Registraci je potřeba potvrdit kliknutím na odkaz, který ti přijde do e-mailu. Po přihlášení by ses měl(a) dostat na hlavní stranu, kde můžeš zaregistrovat nové zařízení pomocí tlačítka *Přidat zařízení*. Pro případ, že bys v budoucnu měl(a) pod jedním účtem více zařízení, je potřeba vymyslet mu nějaký název (například podle toho, kde bude umístěné). Potom si nech vygenerovat svůj unikátní token a prozatím si ho někam zkopíruj nebo zapiš.

Po uložení se nové zařízení objeví v seznamu, kde se kliknutím na jeho název dá nastavit, co se na tvém displeji bude zobrazovat a jak se má tabule chovat. To nejdůležitější se nachází na záložce *Zastávky*. První hodnota určuje, kolik minut předem se mají odjezdy zobrazovat. Jinými slovy to znamená, za jak dlouho dojdeš na zastávku, protože pokud ti cesta tam trvá třeba 5 minut, nemá smysl zobrazovat bližší odjezdy, které už stejně nemáš šanci stihnout.

Tlačítkem **Přidat zastávku** vybereš zastávku, odkud se budou odjezdy zobrazovat. Do vyhledávacího políčka začni psát název zastávky a pak ji vyber z nabízeného seznamu. Potom se ti nabízí dvě možnosti – buď necháš zaškrtnuté políčko *Všechny směry* a zařízení bude vyhledávat všechny odjezdy ze zastávky, lépe řečeno z celého *přestupního uzlu*. V takovém uzlu může být někdy zahrnuto i více sousedících zastávek a hodně dopravních prostředků a směrů (jestli tě to zajímá, zkus si vyhledat třeba Smíchovské nádraží nebo Dejvickou).

Přidat novou zastávku		
náměstí mí		
Náměstí Míru (Praha) X		
	Všechny směry	
✓	Náměstí Míru: A	
	• tram 4: Slivenec	
	<ul> <li>tram 10: Sídliště Řepy</li> </ul>	
	<ul> <li>tram 13: Olšanské hřbitovy</li> </ul>	
	<ul> <li>tram 16: Kotlářka</li> </ul>	
	<ul> <li>tram 22: Bílá Hora</li> </ul>	
	<ul> <li>tram 91: Divoká Šárka</li> </ul>	
	• tram 97: Bílá Hora	
	<ul> <li>tram 99: Sídliště Řepy</li> </ul>	
	Náměstí Míru: M1	
	<ul> <li>metro A: Depo Hostivař</li> </ul>	
	Náměstí Míru: M2	
	metro A: Nemocnice Motol	
<	Náměstí Míru: B	
	<ul> <li>tram 4: Kubánské náměstí</li> </ul>	
	<ul> <li>tram 10: Sídliště Ďáblice</li> </ul>	
	<ul> <li>tram 13: Čechovo náměstí</li> </ul>	
	<ul> <li>tram 16: Ústřední dílny DP</li> </ul>	
	<ul> <li>tram 22: Nádraží Hostivař</li> </ul>	
	<ul> <li>tram 91: Staré Strašnice</li> </ul>	
	<ul> <li>tram 97: Nádraží Hostivař</li> </ul>	
	<ul> <li>tram 99: Zahradní Město</li> </ul>	
	• bus 135: Chodov	
	Náměstí Míru: E	
	• bus 135: Florenc	

Proto jsou tyto *dopravní uzly* ještě dále rozděleny na menší jednotky – jednotlivá *stanovišt*ě. A ty máš možnost omezit zobrazování odjezdů jen na některá z nich. To se může hodit, třeba když tě zajímají jen odjezdy tramvají, ale nepotřebuješ vidět metro. Nebo pokud většinou jezdíš jen směrem do centra, nebudou tě zajímat odjezdy opačným směrem. U každého *stanoviště* máš jako nápovědu uvedeny linky, které odtud odjíždí a jejich konečnou stanici. Vybrat jich můžeš kolik potřebuješ. a potom svůj výběr potvrď tlačítkem **Uložit** na konci stránky. Zpátky na přehled se naopak vrátíš pomocí odkazu **Zpět na detail zařízení**, který najdeš nahoře.

Někdy se může stát, že máš v okolí několik zastávek, které nepatří do stejného přestupního uzlu. Proto můžeš do jednoho zařízení přidat více zastávek a zobrazovat se pak budou odjezdy ze všech najednou. A v detailu zařízení to pak bude vypadat třeba takto:

TIMION: Odjezdové tabule MHD	
Dashboard Účet	Administrace Odhlásit
Kancelář	
Nastavení	Základní nastavení
Nejbližší odjezd (minuty) 3 Upravit	Zastávky
Dřívější odjezdy se nebudou zobrazovat (odpovídá docházkovému času na zastávku.)	Vlastnosti zařízení
	Přidat novou zastávku
Zastavky	Informace o zařízení
Strašnická (Praha)	Odjezdy
Strašnická: A • tram 7: Radlická	
• tram 19: Vozovna Pankrác	
tram 26: Nádraží Hostivař     tram 01: Staré Strašnice	
tram 95: Vozovna Kobylisy	
	J
Pod Rapidem (Praha) Všechny směry	



### Bude to fungovat?

Konečně je tady ta chvíle! Teď můžeš zařízení zapojit do zásuvky. Než se ale pokocháme výsledkem naší práce, čeká nás ještě trochu nastavování.

Ze všeho nejdřív musíme nastavit správný kontrast displeje, aby byl text dobře čitelný. K tomu slouží *trimr*, který najdeš v podobě malého stříbrného plíšku tady:



Vezmi si malý šroubováček a zkus plíškem opatrně otáčet na obě strany. Přitom sleduj, jak se mění kontrast displeje. Najdi polohu, při které je text nejlépe čitelný.

#### Co je to trimr?

Už jednou jsme narazili na součástku s názvem rezistor neboli odpor. Vzpomínáš si, že? No a trimr je zase rezistor, u kterého je možné měnit velikost elektrického odporu. Akorát tentokrát ji měníme mechanicky – otáčením posouváme jezdec po odporové dráze.

Pokud se ti podařilo nastavit displej tak, aby byl dobře čitelný, měl(a) bys na něm vidět něco jako tohle:

```
Nemohu se pripojit k MIFI.
Aktualizovat nastaveni?
Pripoj se k wifi siti:
timion793
192.168.4.1
```

Zařízení potřebuje, abychom mu zadali základní údaje, a proto vytvořilo vlastní WiFi síť, ke které se můžeme připojit a komunikovat přes ní. Vezmi si telefon nebo počítač a připoj se k síti, která je uvedená na displeji (v tomto případě *timion793*).

Po připojení k síti ti telefon/počítač možná sám nabídne přesměrování na úvodní portál. Pokud ne, otevři si internetový prohlížeč a do adresního řádku zadej adresu, která svítí na displeji (v tomto případě *192.168.4.1*). Měl(a) by ses dostat na tuto stránku:

Přihlaste se k síti timion793 192.168.4.1	:
WiFiManager	
timion793	
Configure WiFi	
Info	
Exit	
Update	

Pokračuj na volbu **Configure WiFi**. Najdi v seznamu sítí svojí domácí WiFi síť a klikni na ni. Do spodního políčka napiš heslo k této síti. Ještě pod tím najdeš políčko,

do kterého napiš token, který sis vygeneroval(a) při registraci zařízení. Kdybys ho náhodou ztratil(a) nebo jestli ho třeba po sobě nemůžeš přečíst, můžeš jít kdykoliv zpátky do administrace a vygenerovat si nový.

SSID		
SuiGeneris		
Password		
*****		
Show Password		
Token		
test003		
Spustit test senzorů?		
<ul> <li>Normální režim</li> <li>Test senzorů</li> </ul>		
Save		
Refresh		

Vzhledem k tomu, že zařízení spouštíme poprvé, bude dobré si ověřit, že jsme zapojili správně senzory a že fungují tak, jak bychom očekávali. Proto tentokrát přepni do režimu *Test senzorů.* Pak už stačí jen vše uložit kliknutím na tlačítko **Save**.

#### WiFi Manager

Základním předpokladem pro fungování naší odjezdové tabule je připojení k internetu, v tomto případě prostřednictvím WiFi sítě. Při programování zařízení, jako je tohle, naštěstí není potřeba řešit úplně všechno od nuly, protože pro spoustu běžných situací existují už hotové kusy kódu (tak zvané knihovny), které je možné využít. A tak se nám například o připojení k internetu stará skvělá knihovna *WiFi Manager*. Ta pečuje o to, aby bylo zařízení stále připojené. A když není, sama vytvoří vlastní WiFi síť, přes kterou se můžeš k zařízení připojit a zadat správné údaje k dostupné wifině. Ostatně, to už sis před chvílí sám či sama vyzkoušel(a), takže si to dokážeš představit.

Vrať se zpátky k naší odjezdové tabuli. Na displeji zařízení by se mělo objevit něco takového.



Testovací režim se sám ukončí po 120 vteřinách, jak ukazuje odpočet vpravo nahoře.

"Vrtulka" v levé polovině displeje by se měla začít točit, když pohybový senzor zaznamená pohyb. Nenech se zmást tím, že senzor má několik vteřin "setrvačnost". Když se před čočkou nebude nikdo pohybovat, po nějaké chvíli by se vrtulka měla zastavit.

V pravé části displeje je ukazatel, jestli světelný senzor vnímá v okolí zařízení světlo nebo tmu. Když senzor na boku krabičky zakryješ rukou, hodnota by se měla změnit na "TMA". Hranici mezi světlem a tmou je možné upravit pootočením trimru na světelném senzoru:



Citlivost na tmu můžeš už teď zkusit přizpůsobit svým představám nebo se k tomu můžeš vrátit později. Teď je pro nás hlavně důležité, že je senzor správně zapojený a na světlo reaguje.

Doufejme, že sis úspěšně ověřil(a), že se ti oba senzory podařilo zapojit správně!

V tuhle chvíli už také můžeš krabičku zavřít. Opatrně nasměruj všechny drátky do středu krabičky a víčkem je přitlač dovnitř. Víko je potřeba nejdříve nasadit do drážky na horní straně krabičky a upevnit zacvaknutím paciček do otvorů ve spodní straně. Doporučuju použít malý šroubováček, který se vloží do díry na spodní straně víka a jemným zapáčením proti spodní straně krabičky pomůže zasadit víko na místo. Stejným způsobem můžeš víko zase kdykoliv otevřít, kdyby bylo potřeba

Na displeji odjezdové tabule bys už teď měl(a) vidět nejbližší odjezdy z tebou zvolené zastávky:



Odteď už nebudeš muset na zastávce čekat zbytečně dlouho.



### Co se děje uvnitř?

Do detailního popisu kódu se tady pouštět nebudu, ale alespoň velmi zjednodušeně zkusím nastínit, co se uvnitř zařízení odehrává, abys měl(a) představu.

Po spuštění se zapne displej a zobrazí se úvodní obrazovka. Dále se spustí už zmíněný WiFi Manager a zařízení se připojí k internetu. Následně se zahájí připojení k serveru, ze kterého se bude načítat přesný čas.

Zbytek kódu je uzavřený ve *smyčce*, která se opakuje stále a stále dokola, a to zhruba čtyřikrát za vteřinu. V ní se vždy nejdříve zkontrolují hodiny, jestli nepřibyla další minuta. Pokud ano, přepíší se hodiny na displeji a přepočítá se počet minut do odjezdu.

Potom se zkontrolují senzory – světelný i pohybový – a zařízení na ně zareaguje, například rozsvícením displeje. Dále se kontroluje, kolik času uběhlo od posledního zaznamenaného pohybu, aby se po nastaveném čase podsvícení displeje zase vypnulo.

A v neposlední řadě se (pokud je zařízení v aktivním režimu) každou minutu a půl pošle dotaz na server, který jako odpověď vrátí nejbližší odjezdy ze zadané zastávky. Tato data se zpracují, dopočítá se počet minut do odjezdu a všechno se vypíše na displej.

Samozřejmě je to celé ještě trochu složitější, ale ne o moc. Žádná velká věda to není, co?



#### JSON

{

}

Než se zobrazí odjezdy na displeji, musí proběhnout několik výměn informací: například mezi zařízením a serverem se předávají předpřipravená data o odjezdech a nastavení, server si zase stahuje data o zpoždění spojů, ale je potřeba načítat třeba i seznam zastávek. Aby i velké objemy dat zůstaly přehledné a zpracovatelné, musí se předávat v nějakém strukturovaném formátu.

V našem případě používáme formát *JSON (JavaScript Object Notation)*, což je spolu s XML jeden z nejčastěji používaných způsobů, jak ukládat a sdílet informace na internetu. Data jsou opatřena speciálními značkami, které určují jejich strukturu. Například složené závorky { } vymezují *objekty*. V každém objektu jsou hodnoty přiřazeny ke *klíčům*, které si můžeš představit jako názvy sloupců v tabulce. Dvojice *klíč – hodnota* je oddělena dvojtečkou. Pokud potřebujeme uložit seznam více hodnot, využijeme *pole*, které je uzavřené hranatými závorkami []. Nejspíš to bude zřejmější z tohoto jednoduchého příkladu:

```
"jmeno": "Jan",
"prijmeni": "Novák",
"adresa": {
    "ulice": "Hlavní",
    "cislo_popisne": 123,
    "mesto": "Praha"
},
"telefonni_cisla": [
    "+420 123 456 789",
    "+420 987 654 321"
]
```

Data obsahují objekt, ve kterém jsou hodnoty přiřazeny ke klíčům "jmeno" a "prijmeni". Klíč "adresa" obsahuje jako hodnotu další (podřazený) objekt. Klíč "telefonni\_cisla" má jako hodnotu pole, tedy seznam několika hodnot.

Výhodou formátu JSON je, že je zápis jednoduchý a díky tomu je poměrně snadno čitelný i pro člověka. Zároveň ale umožňuje zapsat jakkoliv složitou datovou strukturu, protože hierarchie objektů je neomezená.

### A je to!

Gratuluju! Mám radost, že jsi všechno zvládl(a) a dočetl(a) až sem. Doufám, že ti bude odjezdová tabule dobře sloužit a také že ses při sestavování dozvěděl(a) něco nového.

Než se rozloučíme, mám pro tebe ještě jednu informaci a jednu prosbu.

Na spodní straně krabičky najdeš malý otvor, který vede k resetovacímu tlačítku na desce. Pomocí malého šroubováku můžeš takto zařízení restartovat. Pokud bys v budoucnu potřeboval(a) změnit nějaké nastavení (např. připojit zařízení na jinou WiFi síť), stačí provést dva restarty rychle po sobě. Pro zařízení je to signál, že má vytvořit vlastní wifinu, přes kterou je možné se dostat do rozhraní s nastavením.



A úplně nakonec slíbená prosba: snažil jsem se ti v tomto návodu předat to, na co jsem sám přišel (většinou metodou *pokus – omyl*), ale určitě se mi nepodařilo zodpovědět všechny otázky. Něco je možná nepřesné nebo nefunguje dobře. Budu moc rád, když mi na můj e-mail **ahoj@timion.cz** napíšeš, jak se ti stavebnice líbila, co by se dalo ještě vylepšit nebo cokoliv dalšího tě k tomu napadá. Předem díky!



### Bylo nebylo...

Možná sis v zápalu sestavování nestihl(a) ani pořádně uvědomit, jak to celé dohromady funguje. Pojďme si to pro jistotu ještě jednou zopakovat na jednoduchém příkladu. Představ si, že se to může odehrávat například takto.

V předsíni na poličce ti teď stojí nenápadná krabička vyrobená metodou **3D tisku** (*str.4*) připojená USB kabelem do zásuvky. Uvnitř se skrývá změť vodičů, kterými jsi propojil(a) jednotlivé součástky tak, abys na ně přivedl(a) **elektrické napětí** (*str. 15*) a ony pak mohly **komunikovat** (*str. 17*) mezi sebou.

Je ráno, světelný senzor (*str. 10*) v zařízení detekuje světlo v okolí a probudí odjezdovou tabuli ze spánku do **režimu pohotovosti** (*str. 29*). Jakmile se k ní teď přiblížíš, pohybový senzor (*str. 12*) tě zachytí a uvnitř se vše okamžitě dá do pohybu. Mikrokontrolér (*str. 9*) se připojí k internetu prostřednictvím tvé domácí wi-fi sítě (*str. 26*). Podle tvého jedinečného kódu (tokenu) se nejdříve ověří, z jaké zastávky (nebo zastávek) se mají odjezdy zobrazovat – podle toho, jak sis to dříve nastavil(a) na speciální webové stránce (*str. 21*).

Mezitím se o kus dál šine ulicemi tvoje tramvaj (nebo jiný dopravní prostředek) a s sebou veze zařízení, které odesílá informace o jeho poloze do **centrální databáze** (*str. 2*). Tam se data porovnají s jízdními řády a spočítá se, že tvůj spoj má dnes ráno třeba 6 minut zpoždění. Tuto informaci si stáhne tvoje zařízení (ve formátu **JSON**, *str. 30*), dopočítá podle aktuálního času, kolik minut ti ještě zbývá do reálného odjezdu, a tuto informaci pošle přes **sériové rozhraní** (*str. 18*) na **LCD displej** (*str. 7*), kde se vše přehledně vypíše.

No a ty si při čištění zubů přečteš, že dneska nemusíš tolik pospíchat, protože bys stejně zbytečně stál(a) na zastávce. Informace na displeji se aktualizují každých 90 vteřin, takže kdyby se něco změnilo, budeš o tom včas vědět. Pár minut potom, co odejdeš z domu, se zařízení zase přepne do pohotovostního režimu a čeká, až ti opět bude moct naservírovat potřebné informace.

