VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

Fakulta informačních technologií

Faculty of Information Technology

BAKALÁŘSKÁ PRÁCE

BACHELOR‘s thesis

Brno, 2016 Ondřej Znojil

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER SYSTEMS

IOt na bázi ESP8266

ESP8266-based node for iot

BAKALÁŘSKÁ PRÁCE

BACHELOR‘s thesis

AUTOR PRÁCE Ondřej Znojil

AUTHOR

VEDOUCÍ PRÁCE doc. Ing. Vladimír Janoušek, Ph.D.

SUPERVISOR

BRNO 2016

Abstrakt

Todo...

Abstract

Todo...

Klíčová slova

Todo...

Keywords

Todo...

Citace

Znojil Ondřej: IoT uzel na bázi ESP8266, bakalářská práce, Brno, FIT VUT v Brně, 2017.

**Název bakalářské práce v jazyce práce**

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením doc. Ing. Vladimíra Janouška, Ph.D.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

……………………

Ondřej Znojil

29.1.2017

*Poděkování*

*V této sekci je možno uvést poděkování vedoucímu práce.*

© Ondřej Znojil, 2017

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

# Obsah

# Todo...Úvod

Todo...

# ESP8266 a IoT

## ESP8266

ESP8266 je Wi-Fi čip s TCP/IP stackem a schopností plně zastávat funkci mikrokontroleru. Byl navržen čínskou společností Espressif System. Jedná se o malé, levné (v současné době lze pořídit v řádech desítek korun) a dostupné zařízení. Jeho účelem je především zajistit internetovou konektivitu zařízení (tzv. IoT – Internet of things), je možné ho použít ve spojení s jinými typy mikro kontroléru (například Arduino), kde komunikace probíhá po sériové lince. Pomocí Wi-Fi dokáže vytvořit jednoduchá TCP/IP spojení pomocí AT (Hayes command set) příkazů. Čip je ovšem programovatelný i sám o sobě. První verze čipu byla vydána v srpnu 2014, jedná se tedy o poměrně mladou technologii. V rámci řešení této práce jsem jako referenční zařízení použil NodeMCU ESP-12. Jedná se o klon Arduina s integrovaným ESP8266 modulem, verze ESP-12. [1][2]

### ESP-12

ESP-12 momentálně k nejprodávanějším a nejpopulárnějším ESP8266 modulům.

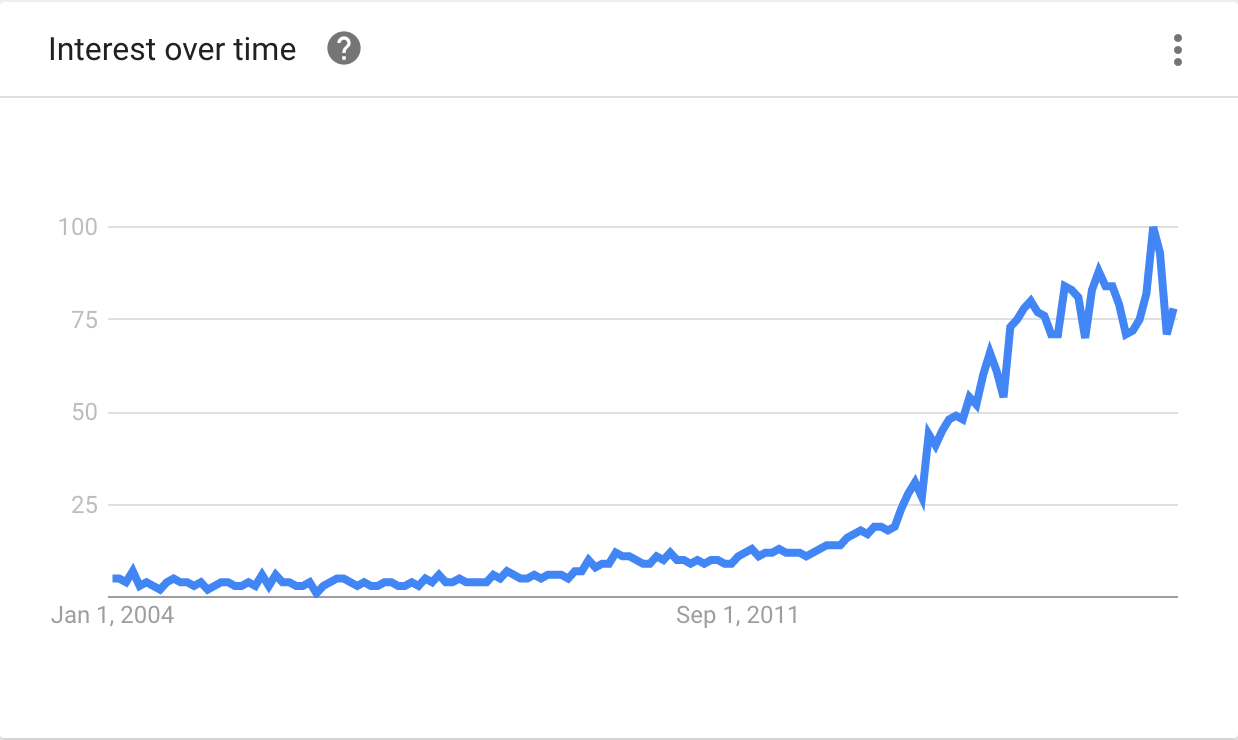
|  |  |
| --- | --- |
| Napětí | 3,3V |
| Spotřeba | 10uA – 170uA |
| Pamět Flash | 4096KB |
| Procesor | Tensilica L106 32 bit |
| Rychlost procesoru | 80-160Mhz |
| Paměť RAM | 32KB + 80KB |
| Vstupy/výstupy | 17 |
| Podpora 802.11 | b/g/n/d/e/i/k/r |
| Maximální počet konkurentních TCP připojení | 5 |

Tabulka 1.1: Specifikace ESP-12

## Inernet of Things (IoT)

Internet of things (Internet věcí), také známo jako internet objektů označuje propojení různých zařízení s internetem. IoT je součástí budoucnosti internetu a může být definováno jako dynamická globální infrastruktura se samokonfigurovatelnou schopností využívající standardní internetové protokoly. Tyto technologie se netýkají pouze chytrých, nebo průmyslových zařízení. Například připojení televizoru k internetu je dnes již standardem. Dopad IoT sahá ovšem mnohem dál; k internetu jsou připojovány nejrůznější senzory, drobná zařízení jako termostaty, žárovky. Síťové připojení je pak použito především pro automatické ovládání a sběr dat, které mohou být posléze podrobeny analýze. [3]

Internet věcí stále více nabývá na popularitě. Tomu napovídá i statistika vyhledávání společnosti Google(obr. 1.1), která ukazuje enormní nárust zájmu o tyto technologie v posledních letech. [5] Je odhadováno, že už dnes je připojeno k internetu více zařízení, než lidí. [4]



## Přístupy programování ESP8266

ESP8266 umožňuje psaní a běh nativních aplikací bez jiných externích mikrokontrolérů. Je také možné napojení modulu k jinému zařízení pomocí sériové linky a využívat ESP8266 pro potřeby Wi-Fi připojení.

### Programování bez použití externích zařízení

Před dvěma lety byl komunitou vyvinut projekt, který poskytuje modulu ESP8266 Arduino jádro, umožňující vytvářet nativní aplikace využívající velmi podobné knihovny těm, které používá Arduino. [6] Toto je velká výhoda zejména proto, že popularita Arduina se zdá být podstatně větší. To přináší výhodu v širší a více přispívající komunitě.

Zdrojový kód PNOS (operační systém Petriho sítí) dodaný společně se zadáním této práce byl navrhnut a otestován na platformě Arduino Mega, proto se nabízí využít tohoto přístupu pro přenesení softwaru na ESP8266.

### AVR-GCC Toolchain

AVR-GCC Toolchain je kolekce nástrojů/knihoven používana pro vytvoření aplikací pro AVR mirkokontrolery. Arduino software, a tedy i jeho knihovny, využívá vždy nejnovější verze těchto nástrojů. Tato kolekce se stará o celkový vývoj programu. Kromě přeložení kompilátorem, a tedy vygenerování binárního kódu, také obsahuje nástroje pro nahrání výsledku do mikrokontroleru. Nezbytnou součástí tohoto toolchainu je také GNU Make, který slouží k řízení celého překladu. Všechny tyto nástroje jsou open-source. [7] Tato kolekce obsahuje:

* GCC
* Binutils
* Libc-avr
* Avrdude

**GCC**

GCC (GNU Compiler Collection) je vysoce flexibilní kompilační systém. Zahrnuje různá front-endová rozhraní pro různé jazyky a také mnoho back-endových rozhraní, které generují kód pro různé procesory a operační systémy. V případě, že je GCC sestaveno k tomu, aby bylo spouštěno na hostitelském systému jako Linux, Windows a generuje kód pro AVR mikrokontrolery, říkáme, že se jedná o tzv. cross-compiler, a tato verze je běžně nazývána AVR-GCC. Tento nástroj umožňuje překlad ze tří jazyků: C, C++, Ada. Překladač sám o sobě nevytváří binární kód, ani nelinkuje knihovny. To má na starosti další z nástrojů, Binutils.

**Binutils**

GCC a Binutils jsou značně provázané a lidé pracující na jednom projektu často pracují zároveň i na tom druhém. Binutils znamená Binary Utilities. Obsahuje assembler a linker, ale také mnoho dalších nástrojů, které pracují s binárními soubory vytvořené částmi tohoto toolchainu. Verze nástrojů sestavené pro generování kódu pro AVR platformy mají pro lepší rozlišitelnost předponu AV, například avr-as (gas), avr-ld (ld), atd...

**Avr-libc**

GCC a Binutils poskytují mnoho nástrojů pro vývoj softwaru, ale neobsahují jednu důležitou knihovnu: standardní C knihovny. AVR toolchain má vlastní pro tuto potřebu vlastní soubor těchto knihoven nazývaných avr-libc. Obsahují mnoho funkcí standardních knihoven, ale také mnoho funkcí specifických přímo pro AVR. Některé funkce běžně užívané v PC prostředí mají své limity, nebo problémy spojené s užití ve vestavěných systémech.

Avr-libc navíc obsahuje většinu dokumentace ohledně celého AVR toolchain.

**AVRdude**

Po přeložení je vytvořen .hex soubor, což je výsledný binární kód. Ten je potřeba nahrát do zařízení. K tomu slouží nástroj AVRdude, který dokáže pracovat s řadou hardwarových zařízení, který dokáže naprogramovat daný procesor. I když byl .hex soubor vygenerován úspěšně, nemusí to znamenat, že na cílové platformě pojede v pořádku. Pokud nebyl překlad proveden s vlastnostmi specifickými pro cílové zařízení, nemusí jít vůbec spustit.

AVRdude je velice flexibilní balíček. Všechny informace ohledně AVR procesorů a různém typu hardware jsou uloženy v textové databázi, kterou může kterýkoliv uživatel upravovat, aby přidal dosud nedefinovaný hardware.

# Problematika přenesení softwaru

Hlavním cílem projektu je přenesení softwaru z platformy Arduino na ESP8266, proto je nutné nejprve si ujasnit rozdíly mezi těmito moduly. Pokud pomineme nejzásadnější rozdíly jako je internetové připojení nebo cena, odlišnosti po hardwarové stránce nejsou malé.

## Arduino Mega a ESP-12 – HW rozdíly

Todo.... (napájení, usb ovladače, pamět, procesor)

## Arduino Mega a ESP-12 – SW rozdíly

### Mapování pinů

Todo – SW mapování pinů

### Čekání (delay)

Pokud se zařízením nepotřebujeme nic dělat, je vhodné ho nějakým způsobem uspat, aby neběžely zbytečné procesy a snížil se výkon. V tomto ohledu je mezi těmito dvěma platformami kruciální rozdíl. [8]

ESP8268 provádí mnoho úkonů na pozadí – udržování Wi-Fi spojení, obsluha TCP/IP stacku a provádění dalších potřebných funkcí. Blokace těchto procesů může způsobit spadnutí a reset systému. K zamezení resetů je doporučeno vyhnutí se dlouhých blokujících smyček v programu.

K tomuto účelu byla vývojáři Arduinovských ESP8266 knihoven vytvořena funkce yield(), která program zdánlivě uspí, ale umožní provádění všech úkonů běžících na pozadí, zaručujících běh úkonů na pozadí. To poskytuje celkovou stabilitu systému.

### Knihovny

Většina knihoven Arduinovské ESP8266 knihovny zachovává stejné názvy tříd a funkcí, aby se používali stejně jako původní. Je zde ovšem několik rozdílů:

Wire -

SPI -

Sériový monitor -

# Závěr

Todo...

Literatura

[1] Neil Kolban: Kolban`s book on ESP8266 [online]. Brno: c1993-2008, aktualizováno 2016-01 [cit. 2017-01-30]. Dostupné na URL: < https://www.academia.edu/23729696/ESP8266\_ESP>

[2] https://www.academia.edu/28304468/Home\_Automation\_Using\_Arduino\_and\_ESP8266

[3] 3rd International Conference on Advances in Engineering Sciences & Applied Mathematics (ICAESAM’2015) March 23-24, 2015 London (UK)

[4] Daves Evans (2011) The Internet of Things: How the Next Evolution

of the Internet Is Changing Everything Cisco IBSG © 2011 Cisco

and/or its affiliates. Available at

http://www.cisco.com/web/about/ac79/docs/innov/IoT\_IBSG\_0411FIN

AL.pdf

[5] <http://www.google.com/trends/explore?hl=en-US#q=internet+of+things>

[6] https://github.com/esp8266/Arduino

[7] <http://www.atmel.com/tools/atmelavrtoolchainforwindows.aspx>

[8] https://learn.sparkfun.com/tutorials/esp8266-thing-hookup-guide/using-the-arduino-addon

Seznam příloh

Todo...?