Obsah

Kapitola 1

Úvod

Cílem této práce je vyhodnocní požadavků na real-time vlastnosti komunikace a synchronizaci času v SCADA QNX.

1.1 Precision Time Protocol

PTP (uvažujeme standard IEEE 1588-2008) je protokol zajišťující synchronizaci hodin mezi uzly síťové topologie. Tento protokol je navržen pro místa, kde je vyžadována vysoce přesná (odchylky řádově několik desítek nanosekund v laboratorním prostředí) synchronizace času mezi síťovými uzly, avšak z nějakého důvodu není možné použít řešení založené na GPS (viz. kap. 1.3). Praktické využití tedy nalézá v průmyslových oblastech a to zejména v řídících a monitorovacích systémech SCADA (viz. 1.4).

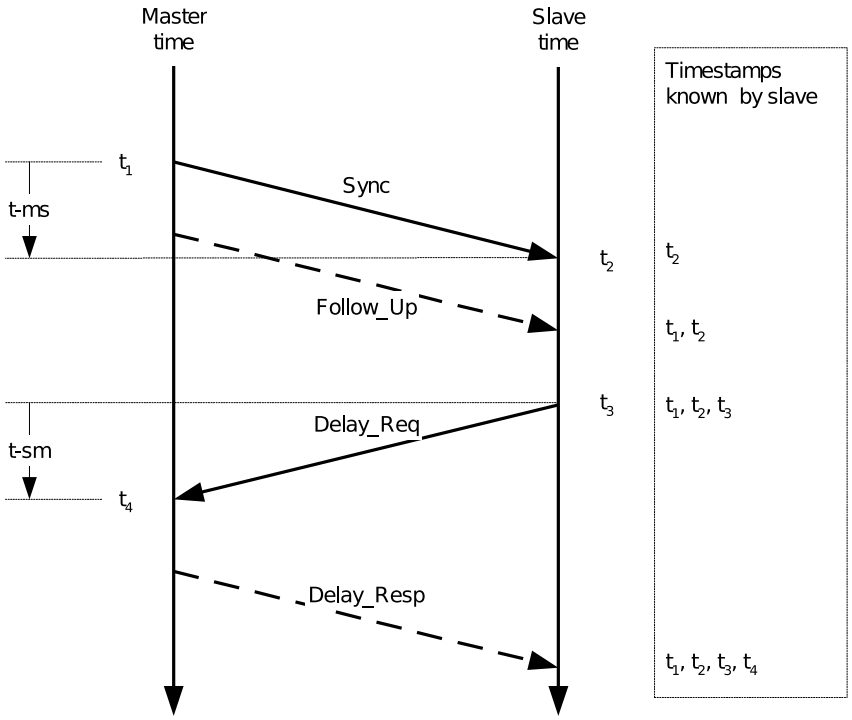
Principem funkčnosti je vztah `master-slave` mezi dvěma uzly, přičemž `master` periodicky zasílá zprávy, obsahující informace pro korekci času, stanicím `slave`, a to po krátkých intervalech (nejmenší povolený interval je $0.1s$, avšak běžně se používá $1s$). Komunikace probíhá přes multicast s tím, že je možné nasadit čistě unicastovou komunikaci.

Multicastová komunikace probíhá tak, že `master` periodicky zasílá `Announce` zprávy na adresy `224.0.0.107` (`peer delay` zprávy) a `224.0.1.129` (všechny ostatní zprávy) v případě IPv4 a na `FF02::6B` (`peer delay` zprávy) a `FF0x::181` (všechny ostatní zprávy) v případě IPv6 (pro adresování v ostatních podporovaných protokolech viz. [8]). Ukázka jednoduché multicastové komunikace s popisem jednotlivých zaznamenaných časových razítek je znázorněna na obrázku 1.1.

Při unicastové komunikaci `master` rozesílá periodicky `Announce` zprávy všem `slave` uzlům zvlášť. V tomto případě PTP nezajišťuje `objevování` stanic v síti, a je tedy nutné `master` uzlu sdělit které `slave` uzly má obsluhovat. K tomu lze kromě ručního nastavení využít i možností `hraničních hodin`, kterým by byla ručně nastaven seznam `master` uzlů a od nich by byly periodicky vyžadovány `Announce` zprávy.

PTP využívá tzv. domény, které vytvářejí logické celky a tyto se navzájem neovlivňují. Je tedy možné v každé doméně mít jiný čas a zařízení může v každé doméně zastupovat jinou roli. V jedné doméně se však může nacházet pouze jeden uzel `grandmaster clock`, který slouží jako referenční hodiny pro danou doménu, a je tedy vrcholem celé hierarchie PTP uzlů. Tento uzel je volen pomocí `BMC` (Best Master Clock) algoritmu z připojených master uzlů.

BMC algoritmus beží na všech `běžných` a `hraničních` hodinách a nepřetržitě přizpůsobuje stavy fyzických linek změnám v síti a hodinách. Výpočet je prováděn pouze lokálně



Obrázek 1.1: PTP komunikace využívající multicast

nad každou fyzickou linkou zvlášť a to ve dvou následujících krocích:

1. porovnání `data sets` každých dvou hodin a vybrání nejlepších z nich

   Cílem je nalézt hodiny, které odvozují svůj čas od lepšího `grandmaster` uzlu, nikoliv hodiny, které jsou lepší (toto je nutná podmínka pro stabilitu algoritmu). Pro porovnání se využívá identita (na rovnost) , priority1, třída hodin, přesnost, koeficient `offsetScaledLogVariance` (viz. [8], kap. 7.6.3.2), priority2 a identita (na nerovnost) v tomto pořadí. Algoritmus je zachycen v [8], obr. 27 a 28 . Tento algoritmus též odhalí chyby, že PTP zpráva byla odeslána a přijata na stejném portu, nebo že zpráva byla zdvojena, případně, že jsou porovnávané zprávy obě od `grandmaster` uzlu.

2. výpočet „doporučeného stavu“ pro každou fyzickou linku

   Při porovnávání je využito charakteristik hodin (třída, kvalita topologie a fyzický port). K porovnání kvality topologie je znovu využit algoritmus pro porovnání `data sets`. Algoritmus je zachycen v [8], obr. 26 .

Ačkoliv jsou podporovány různé síťové topologie, protokol nezajišťuje zasílání zpráv s vlastní eliminací smyček a spoléhá v tomto ohledu na nosné protokoly. Smyčka by mohla způsobit potíže např. při výpočtu zpoždění na uzlu `boundary clock`, protože je podporováno maximálně 255 těchto uzlů. K provozu PTP však není zapotřebí spolehlivé linky (k přenosu časových informací se využívají nespolehlivé protokoly), ačkoliv je doporučeno předejít zdvojování paketů, jejich vysoké ztrátovosti a změnu pořadí.

PTP dále definuje chování pro zařízení, skrze která mají zprávy PTP pouze procházet. Pokud možno, mají tato zařízení upravovat pole `correctionField` ve zprávách PTP, a to tak, že přičtou zpoždění vzniklé přenosem přes vstupní médium a zpoždění vzniklé zpracováním

paketu uvnitř tohoto zařízení. Přesnost tohoto postupu závisí na rozdílu délky periody hodin tohoto zařízení a `master` uzlu. Lze tedy tyto průchozí uzly (tzv. transparentní hodiny) synchronizovat s `master` uzlem pasivním pozorováním PTP zpráv (`Sync` a `FollowUp`) spolu s aplikováním potřebných korekcí a poté buď změnit frekvenci oscilátoru (tzv. analogové řešení) a nebo neměnit frekvenci ale spočítat koeficient, kterým se vynásobí změřené zpoždění před přičtením k `correctionField` (tzv. digitální řešení).

Zařízení v PTP doméně jsou rozdělena na následující typy:

I. `ordinary clock` (běžné hodiny)

   Cílové zařízení pro synchronizaci času - může to být `master` nebo `slave`.

II. `boundary clock` (hraniční hodiny)

   Zařízení oddělující jednotlivé segmenty sítě a podporující korekci času v PTP zprávách. Mívá více fyzických portů pro připojení více uzlů.

III. `transparent clock` (transparentní hodiny)

   (a) `end-to-end` (E2E)
   Zařízení podporující `end-to-end delay` měřící mechanismus mezi `master` a `slave` uzly. Toto zařízení tedy neumí měřit zpoždění samotné fyzické linky, nýbrž pouze zpoždění způsobené zpracováním v tomto zařízení.

   (b) `peer-to-peer` (P2P)
   Zařízení podporující i korekci zpoždění fyzické linky. V případě přítomnosti těchto transparentních hodin se pro měření zpoždění mezi `master` a `slave` využívá `peer-to-peer delay` mechanismus.

IV. `management nodes` (řidící uzly)

   Zařízení pro konfiguraci a monitorování hodin (není však podmínkou pro provozování PTP).

Pro měření zpoždění se využívá dvou mechanismů. Mechanismus `request-response delay` měří průměrnou dobu, jakou trvá zaslání zprávy od `master` uzlu ke `slave` uzlu. Zde narážíme na problém s různou délkou zpoždění pro příchozí a odchozí směr komunikace a je tedy nutné navíc provést asymetrické korekce (viz. [8], kap. 11.3). Druhým mechanismem je `peer delay`, který měří zpoždění způsobené propagací skrze fyzickou linku (pro každou zvlášť). Princip tohoto měření je popsaný v [8], kap. 11.4 .

Základní vzorec 1.1 pro výpočet zpoždění nebere v úvahu asymetrické korekce a využívá mechanismus `request-response delay`. Časy t_n odpovídají časům na obrázku 1.1 a δ je offset mezi časem `master` a `slave`. Předpokládá se tedy, že hodiny na `master` i `slave` se mezi časy t_1 až t_4 nezrychlí ani nezpomalí (tedy, že offset je konstantní).

$$\begin{aligned} t_2 - t_1 - \delta &= t_4 - t_3 + \delta \\ \delta &= (t_2 - t_1 - t_4 + t_3)/2 \end{aligned} \tag{1.1}$$

PTP definuje následující typy zasílaných zpráv:

- Zprávy `Sync`, `Delay_Req`, `Delay_Resp` a `Follow_Up` jsou využívány k přenosu časových informací využívaných k synchronizaci běžných a hraničních hodin podle měřeného zpoždění mezi dotazem a odpovědí.

- Zprávy `Pdelay_Req`, `Pdelay_Resp` a `Pdelay_Resp_Follow_Up` slouží k měření meření zpoždění linky mezi dvěmi hodinami implementujícími mechanismus `peer delay`. Toto zpoždění je využito ke korekci časové informace ze zpráv `Sync` a `Follow_Up` v topologiích složených z peer-to-peer transparentních hodin. Běžné a hraniční hodiny implementující `peer delay` mechanismus se tedy synchronizují podle změřených zpoždění linky a informací obsažených ve zprávách `Sync` a `Folow_Up`.

- Zpráva `Announce` ustavuje synchronizační hierarchii.

- Zprávy `Management` slouží k vyžádání a přenosu `PTP data set` udržovaných jednotlivými hodinami. Pomocí nich je možné síť PTP nastavovat, a to zejména při inicializaci a výpadcích. Tyto zprávy si mezi sebou posílají řídící uzly a hodiny.

- Zprávy `Signalling` slouží k výměně veškerých ostatních informací mezi jednotlivými hodinami. Například jsou využívány pro dohodnutí jak často si budou `master` a `slave` zasílat unicast zprávy.

Každá zpráva obsahuje hlavičku, jejíž struktura je pro všechny zprávy totožná. Tato je znázorněna na obrázku 1.2. Je zřejmé, že struktura je nezávislá na technologii přenosu této informace, ačkoliv je PTP nejčastěji využíván ve spojení s IPv4/6 a Ethernetem. Všechny zprávy lze dodatečně rozšířit pomocí `type length value` (TLV) formátu (viz. [8], kap. 14).

Bits								**Octets**	**Offset**
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**		
transportSpecific				messageType				1	0
reserved				versionPTP				1	1
messageLength								2	2
domainNumber								1	4
reserved								1	5
flagField								2	6
correctionField								8	8
reserved								4	16
sourcePortIdentity								10	20
sequenceId								2	30
controlField								1	32
logMessageInterval								1	33

Obrázek 1.2: Společná struktura hlavičky všech PTP zpráv

Mezi podporované transportní protokoly patří Ethernet (IEEE 802.3), UDP IPv4, UDP IPv6, DeviceNet, ControlNet a PROFINET. Avšak je možné využít speciálních hodnot v polích určujících typ i pro další nosné protokoly, které nejsou zahrnuty ve specifikaci.

Pro potřeby měření a vyhodnocení byla využita implementace protokolu PTPd verze 2 [10], která byla pro tyto účely nejprve portována do prostředí QNX. PTPd implementuje pouze podmnožinu IEEE 1588-2008, a sice pouze části potřebné pro splnění požadavků v IEEE 802.1AS (Timing and Synchronization in Bridged Local Area Networks). Tyto části zahrnují zejména podporu pro multicastovou, unicastovou a hybridní komunikaci skrze UDP IPv4/6 a podporu pro `end-to-end` a `peer-to-peer` měřící mechanismy.

Hybridní mód využívá multicastová oznámení pouze k objevování nových `slave` uzlů, ale veškerá další komunikace mezi `master` a jednotlivými `slave` uzly probíhá pouze unicastově.

1.2 Network Time Protocol

NTP představuje/je... multicast FF00:...

Princip funkcionality... => nelze dosáhnout vysoké přesnosti

Použití na místech... + nelze použít na místech... + proč?

Praktické zkušenosti z provozu/nasazení... (jenom krátce natuknout - rozebere se to az nize)

QNX používá jakou implementaci? je to NTP nebo SNTP?

1.3 Modul GPS

Global Positioning System poskytuje nejpřesnější běžně dostupnou metodu synchronizace času. Ve SCADA systémech se nejčastěji používají samostatné GPS moduly pro každou stanici. Tím je zajištěna velice přesná synchronizaci času ve všech stanicích, a to nezávisle na sobě. Tento způsob je však finančně velice nákladný a v některých prostředích je komplikované přivést ke každé stanici zvlášť samostatný anténní modul, který je základní podmínkou funkčnosti GPS modulu. Stále rychleji rostoucí penetrace ethernetové kabeláže v průmyslových oblastech a budovách umožňuje využít např. protokol `PTP` pro synchronizaci stanic a GPS modul využít pouze pro referenční stanici `grandmaster`.

technick vyhody a nevyhody GPS (antena, napajeni, podpora v HW, podpora v SW, nedostacujici signal, pri selhani jineho sitoveho uzlu neni synchronizace ohrozena)...

1.4 SCADA QNX

Praktické využití tedy nalézá v průmyslových oblastech a zejména v řídících a monitorovacích systémech jako je SCADA (

SCADA (Supervisory Control And Data Acquisition) systémy jsou řídící a monitorovací systémy využívané především v průmyslových oblastech, výrobě, automotive apod. Tyto systémy tvoří jeden úzce propojený celek složený z heterogenních prvků, kterými jsou různá čidla a elektro-mechanická zařízení pro řízení strojů, výroby apod. Jelikož tento celek obsahuje mnoho časově závislých procesů, je nutné, aby každý prvek byl řízen stejným časem, a proto tyto prvky zpravidla spadají do tříd `firm` a `hard` v real-time klasifikaci.

Příklady - co dela nas testovaci scada system; co nedela a mel_by/mohl_by; ???

QNX je... posix, real-time, linux-like, stary-dlouho-vyvijeny, automotive, elektrarny, atd.

1.5 Generátor provozu (Spirent)

Zařízení Spirent je generátor síťového provozu...

1.6 OMNeT++

OMNeT++ je diskrétní simulátor napsaný v jazyce C++ a ...

Kapitola 2

Měření a vyhodnocení

Cílem měření bylo stanovit dosažitelnou přesnost pro získávání a přiřazování časových údajů k zaznamenávaným datům za použití prokolů PTP, NTP a nezávislé synchronizace stanic skrze modul GPS.

2.1 Princip měření

Princip měření zpoždění protokolu PTP spočívá v pravidelném, periodickém překlápění hrany iniciovaném z `grandmaster` stanice a zároveň ze `slave` stanice ve stejný reálný čas. Délka periody pro generování hrany je korelována s periodou zasílání synchronizačních informací od `grandmaster` (offset mezi nimi je irelevantní). V opačném případě by byla zbytečně měřena přesnost samotných RTC hodin `slave` stanice a ne zpoždění způsobené propagací synchronizační informace od `grandmaster` stanice.

K tomuto účelu bylo sestaveno měřící zařízení využívající platformu `FitKit` [6], jež zachytává hrany nezávisle z `grandmaster` i `slave` stanic a porovnává zpoždění. Měří se počet hodinových cyklů měřícího zařízení uplynulých od první zachycené hrany. Po detekování hrany od `grandmaster` nebo `slave` je uložen aktuální stav čítače (tj. počet uplynulých cyklů od první zachycené hrany) do pole v paměti. Po dosažení určitého počtu měření je toto pole přeneseno z měřícího zařízení do pracovní stanice (collector) a vyhodnoceno.

Zápis Protože se však hodiny v `grandmaster` a `slave` mohou rozcházet v čase různě (zpožďovat nebo zrychlovat), je nutné při záchytu hrany uložit obě hodnoty a až později statisticky rozlišit, zdali každé dvě po sobě jdoucí naměřené hodnoty odpovídají vygenerovaným hranám ve stejný reálný čas na `grandmaster` i `slave` stanici. Z tohoto důvodu je zachycen stav globálního čítače při každém detekování hrany (jak od `grandmaster`, tak od `slave`) a uložen pro vyhodnocení probíhající mimo měřící zařízení.

`grandmaster` a `slave` využívají ke generování hrany sériový spoj, který má však nevýhodu, že je připojen až na nejpomalejší části hierarchie sběrnic uvnitř těchto stanic. Podle dokumentace k čipovým sadám těchto stanic je zpoždění způsobené propagací skrze tuto hierarchii konstantní a lze tedy naměřené rozdíly považovat za přesné, ačkoliv sériové rozhraní neposkytuje tak vysoké přenosové rychlosti, odpovídající zpoždění desítek nanosekund.

```
#define BIT64 0x8000000000000000  /* MSB = 1 */
#define MAX 2048
UINT64_HARDWARE_COUNTER cnt = 0;
static uint8_t WILL_SEND_MEASUREMENTS_TO_COLLECTOR = 0;
// each second at 100MHz
```

```
static uint64_t arr[MAX];
static uint16_t i = 0;
from_master() {
    disable_interrupts();
    arr[i++] = cnt | BIT64;  // indicates master
    if (i == MAX)
        WILL_SEND_MEASUREMENTS_TO_COLLECTOR = 1;
    else
        enable_interrupts();
}
from_slave() {
    disable_interrupts();
    arr[i++] = cnt;  // assume BIT64 is not set
    if (i == MAX)
        WILL_SEND_MEASUREMENTS_TO_COLLECTOR = 1;
    else
        enable_interrupts();
}
```

programy:

- PTPd (@ptpdSourceforge) - portace na QNX
- edges generator (through serial port)
- edges acceptor (MSP430 firmware)
- measurements accumulator

Byly simulovány různé typy síťového provozu pomocí zařízení `Spirent` a pro každý z nich byly měřeny výsledky. Tyto slouží pro určení přesnosti a spolehlivosti jednotlivých metod synchronizace v různých situacích a pro tvorbu simulačního modelu.

Byly stanoveny následující typy:

a. 10% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 70B$

b. 10% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 1480B$

c. 30% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 70B$

d. 30% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 1480B$

e. 60% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 70B$

f. 60% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 1480B$

g. 98% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 70B$

h. 98% šířky pásma, exponenciální rozložení velikosti paketů $\mu = 1480B$

Všechna měření byla prováděna na následujících třech síťových topologiích:
FIXME obrázky!!!!!!!!!!

1. switch (Cisco XXXX 100Mb ethernet) -+- MA
   +- Spirent
   +- SL

2. switch -+- router - SL
   +- Spirent
   +- router - MA

3. switch -+- router VPN gateway - SL
   +- Spirent
   +- router VPN gateway - MA

Byly zjišťovány hodnoty PTD (Packet Transmission Delay) a ?????????????PDV (Packet Delay Variation)????????????.

Network #	load (%)	TA	TB	PTD
a	20	x1	x2	???
b	50	y1	y2	???
c	70	z1	z2	???
...	...	...	...	...

Tabulka 2.1: Výsledky měření

2.2 Metody vyhodnocení

ADEV
MDEV
TDEV
minTDEV
bandTDEV + percentileTDEV

2.3 Výsledky vyhodnocení

PTP vysoce přesné za běžných okolností, avšak méně přesné pro třídu provozu X, nepoužitelné pro třídu provozu Y atd. (čísla, grafy, reference na články jak to vycházelo jiným...)

NTP mnohem méně přesné (čísla, grafy, reference kdo to kdy měřil apod.)

GPS ...

Bylo ověřeno, že protokol PTP je schopný udržovat synchronní čas na mnoha stanicích v topologii hvězda s vysokou přesností a ve většině případů nahradí GPS modul... FIXME

Kapitola 3

Demonstrační úlohy

V závislosti na vhodnosti metody pro synchronizaci času ve SCADA QNX byly připraveny konfigurace a demonstrační aplikace pro PTP, NTP a GPS.

Každá aplikace se zaměřuje na jinou třídu SCADA systémů. Vychází se přitom z provedené klasifikace.

3.1 Aplikace 1

soft systémy => postačuje NTP

3.2 Aplikace 2

firm systémy => je využito PTP, avšak lze využít za okolností uvedených v ??????????????????????vysledky_mereni_a_zhodnoceni_NTP????????????????? i NTP

3.3 Aplikace 3

hard systémy => je využito GPS synchronizace, avšak lze využít za okolností uvedených v ??????????????????????????vysledky_mereni_a_zhodnoceni_PTP???????? i PTP

Kapitola 4

Model v OMNeT++

Na základě provedených měření byl vytvořen model scada systému parametrizovatelný třídou, náročností na šířku přenosového pásma (minimální, průměrná a maximální) a ???????

Všechna provedená měření (viz. kap. 2) byla odsimulována a výsledky porovnány v tabulce níže. Je zřejmé, že model odpovídá reálným měřením a lze ho tedy použít pro návrh komplexních síťových topologií pro SCADA systémy.

Network #	load (%)	TA	TB	PTD simulation	PTD real
a	20	x1	x2	???	???
b	50	y1	y2	???	???
c	70	z1	z2	???	???
...	...	...	...	...	...

Tabulka 4.1: Výsledky simulace a reálných měření

Kapitola 5

Závěr

Nějaký neomalený závěr...

Literatura

citace upravit podle CSN!!!; jak citovat standardy?

[1] F. Zezulka and O. Hyncica. *Synchronizace v distribuovaných řídících systémech: Precision Time Protocol podle IEEE 1588*, AUTOMA, vol. 2, pp. 17-19, 2010.

[2] J. C. Eidson and K. Lee, „Sharing a Common Sense of Time," *IEEE Instrumentation and Meassurement Magazine*, no. 3, 2003.

[3] V. Smotlacha, „Time Issues in One-way Delay Measurement," Czech Technical University in Prague, 2005.

[4] Q. M. Chaudhari, „A Simple and Robust Clock Synchronization Scheme," *IEEE Transactions on Communications*, vol. 60, no. 2, pp. 328-332, Feb. 2012.

[5] V. Igure, S. Laugher and R. Williams, „Security issues in SCADA networks," *Computers & Security*, vol. 25, no. 7, pp. 498-506, Oct. 2006.

[6] http://merlin.fit.vutbr.cz/FITkit/, 2012-06-23, cit. 2013-01-18

[7] Cosart L. *Studying network timing with precision packet delay measurements*, R&D, Symmetricom, Inc., 2009, ISSN ??-???-????-?. str. 999-9999

[8] IEEE 1588-2008

[9] RFC 5904 (NTPv4)

[10] PTPd. *Precision Time Protocol daemon* [online]. 2012——, [cit. 2013-01-08]. Dostupné na: http://ptpd.sourceforge.net/.