

Mobilní aplikace pro sběr dat ze smart elektroměrů

Bc. Andrej Pečimuth

Diplomová práce
2016



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2015/2016

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Andrej Pečimuth**
Osobní číslo: **A14478**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Informační technologie**
Forma studia: **kombinovaná**

Téma práce: **Mobilní aplikace pro sběr dat ze smart elektroměrů**
Téma anglicky: **A Mobile Application for Data Collection from Smart Grid Devices**

Zásady pro vypracování:

1. Sestrojte hardwarový modul měřicího systému, který bude obsahovat mobilní telefon, elektroměr, podpůrné zařízení a jejich vzájemné propojení po sériové sběrnici.
2. Navrhněte vhodnou grafickou a funkcionální podobu aplikace pro OS Android.
3. Určte vhodnou datovou strukturu a typ komunikačního přenosu mezi odpočítáváním elektroměrem a mobilním telefonem a mobilním telefonem a centrálním serverem pro odpočet měřících zařízení.
4. Naprogramujte potřebnou množinu odečtových skriptů pro běžný provoz elektroměru.
5. Sestrojený hardwarový model a naprogramovaný software prakticky otestujte v testovacím provozu na elektrickém hnacím drážním vozidle nebo jiném reálně měřeném odběru.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. Grant Allen. **Android 4: Průvodce programováním mobilních aplikací.** Přeložil Mužík Jakub. Brno: Computer Press, 2013. ISBN 978-80-2513-782-6.
2. Herbert Schild. **Java 7.** Brno: Computer Press, 2012. ISBN: 978-80-251-3748-2.
3. Miroslav Ujbányai. **Programujeme pro Android.** Praha: Grada Publishing, 2012. ISBN 978-80-247-7983-6.
4. **SMERNICA EURÓPSKEHO PARLAMENTU A RADY 2009/72/ESz 13. júla 2009 o spoločných pravidlách pre vnútorný trh s elektrinou, ktorou sa zrušuje smernica 2003/54/ES. Úradný vestník Európskej únie, 2009.**
5. **STN EN 62056-21. Meranie elektrickej energie: Výmena údajov na odčítanie elektromerov, na ovládanie sadzieb a zaťaženia.** Prešov: Krížik a.s., 2009.
6. Herbert Schildt. **Mistrovství – Java: Kompletní průvodce vývojáře.** Přeložil Daněk Milan. Brno: Computer Press, 2014. ISBN: 978-80-251-4145-8.

Vedoucí diplomové práce:

Ing. David Malaník, Ph.D.

Ústav informatiky a umělé inteligence

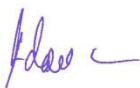
Datum zadání diplomové práce:

5. února 2016

Termín odevzdání diplomové práce:

20. května 2016

Ve Zlíně dne 5. února 2016



doc. Mgr. Milan Adámek, Ph.D.
děkan



doc. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s přípuštěním, tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považuji se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové/bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

17.5.2016



.....
podpis diplomanta

ABSTRAKT

Cieľom tejto práce bolo navrhnuť a naprogramovať aplikáciu pre operačný systém Android, ktorá dokáže zabezpečovať automatický aj lokálny zber dát zo smart elektromerov. Teoretická časť tejto práce pojednáva o inteligentných sieťach, chytrých meracích systémoch, elektromeroch, technológiách, protokoloch a systémoch, ktoré boli použité pre vývoj našej aplikácie.

Praktická časť sa venuje použitým hardvérovým komponentom, ktoré sú potrebné pre chod aplikácie a samotnou aplikáciou ELMReader. Je opísaná funkcionálna vytvorenej aplikácie, jej grafický návrh, detailný popis programovej konštrukcie a testovanie aplikácie v laboratórnych podmienkach a testovanie v ostrej prevádzke na lokomotíve škoda radu 671.

Kľúčové slová: Inteligentné meranie, Android, inteligentné siete, Elektromer, Zber dát.

ABSTRACT

The aim of this work was to design and build application for the Android operating system that can provide automatic and local data collection of smart electromechanical-ROV. The theoretical part of this work deals with smart grids, smart metering, electricity meters, technologies, protocols and systems that have been used for the development of our application.

The practical part is devoted to the use of hardware components that are needed to run the application and by the application ELMReader. Described functionality of the application created a graphic design, a detailed description of software design and test applications in laboratory and testing in live operation for locomotives shame advice 671st

Keywords: Smart metering, Android, Smart grids, Electromer, Data collection.

OBSAH

ÚVOD.....	7
I TEORETICKÁ ČASŤ.....	8
1 SMART GRID	9
1.1 SMART GRID V LEGISLATÍVE	10
1.1.1 Vízie smart grid v EU a ich podpora členskými štátmi	11
1.2 DEFINÍCIA SMART GRID	13
1.3 DEFINÍCIA INTELIGENTNÉHO MERACIEHO SYSTÉMU	13
1.3.1 Základné technické parametre Inteligentného meracieho systému.....	14
1.4 ELEKTROMERY POUŽÍVANÉ V INTELIGENTNÝCH MERACÍCH SYSTÉMOCH.....	14
1.5 EXISTUJÚCE MERACIE SYSTÉMY NA HNACÍCH DRÁHOVÝCH VOZIDLÁCH	16
2 POUŽITÉ TECHNOLOGIE.....	18
2.1 OPERAČNÝ SYSTÉM ANDROID	18
2.2 PROGRAMOVACÍ JAZYK JAVA.....	20
2.3 JAZYK XML	22
2.4 VÝVOJOVÉ PROSTREDIE ANDROID STUDIO.....	23
2.5 KOMUNIKAČNÉ ROZHRANIE RS 485.....	25
2.6 FTP PROTOKOL	28
II PRAKTICKÁ ČASŤ	30
3 TECHNICKÉ PROSTRIEDKY	31
3.1 POUŽITÝ MOBILNÝ TELEFÓN	31
3.2 POUŽITÝ PREVODNÍK USB NA RS 485	32
3.3 POUŽITÝ ELEKTROMER PRE VÝVOJ A TESTOVANIE APLIKÁCIE	33
3.4 POUŽITÝ ELEKTROMER PRI TESTOVANÍ NA HDV ŠKODA.....	40
4 APLIKÁCIA ELM-READER.....	43
4.1 GRAFICKÝ NÁVRH APLIKÁCIE.....	44
4.2 NASTAVENIE ANDROIDMANIFEST.....	47
4.3 OVLÁDANIE USB	48
4.4 FTP PRIPOJENIE	54
4.5 POPIS FUNKCIÍ APLIKÁCIE.....	56
4.5.1 Automatický režim aplikácie	56
4.5.2 Odčítanie registra elektromeru.....	58
4.5.3 Odčítanie záťažového profilu elektromera.....	58
4.5.4 Zápis a odčítanie času elektromera	60
4.5.5 Nastavenia aplikácie.....	60
4.6 TESTOVANIE APLIKÁCIE	62
ZÁVER	67
ZOZNAM POUŽITEJ LITERATÚRY	68
ZOZNAM POUŽITÝCH SYMBOLOV A SKRATIEK.....	70
ZOZNAM OBRÁZKOV	71
ZOZNAM TABULIEK	73

ÚVOD

Meranie elektrickej energie začalo už v počiatkoch rozvoja energetiky a elektrotechnického priemyslu. Elektromery, tak ako aj iné zariadenia, prešli rôznymi etapami prirodzeného vývoja od jednoduchých mechanických elektromerov cez statické až po najnovší dnešný trend, ktorým sú jednoznačne smart elektromery. Smart elektromery majú podporiť príchod efektívnejších „smart grid“ energetických sietí. Výrobcovia elektromerov majú v ponuke aj podporné príslušenstvo, ako sú modemy na prenos údajov z elektromera na centrálu či špeciálne určené priemyselné počítače. Tieto zariadenia však často krát viacnásobne prevyšujú cenu elektromera a ich použitie sa stáva na menej dôležitých či dočasných odberných miestach nerentabilné. Takisto parametrizačný softvér servisných technikov pracujúcich v energetike je spravidla určený pre operačné systémy Windows a pri jeho použití je teda nutné mať pri sebe notebook. Softvér k elektromerom je navyše viazaný ku konkrétnemu výrobcovi. Keďže chytré mobilné telefóny sú už dnes prirodzenou súčasťou nášho života, je veľká škoda, že sa žiadny z výrobcov neodhodlal k vytvoreniu aplikácie pre chytrý telefón.

Cieľom tejto diplomovej práce je preto návrh aplikáciu pre chytré telefóny, ktorá by dokázala uľahčiť prácu technickým zamestnancom energetických spoločností a zároveň by bola schopná čiastočne nahradiť v súčasnosti používané priemyselné počítače v nerentabilných nasadeniach. Aplikácia by sa mala vyvíjať pre najčastejšie používaný mobilný operačný systém Android a ako vývojové prostredie by malo byť použité IDE Android Studio. Aplikácia by mala obsahovať základné najpoužívanějšíe funkcie pri servise elektromerov, ako sú odpočet registra elektromeru, dát zátťažových profilov či odpočet a synchronizáciu času elektromera. Ďalej by mala zvládať automatický odpočet dát elektromera v prednastavenom čase a ich uloženie na vzdialený server. Keďže prenos dát bude prebiehať cez mobilné siete, aplikácia by mala byť testovaná na pohyblivom odbernom mieste, konkrétne pôjde o železničné koľajové vozidlo značky Škoda.

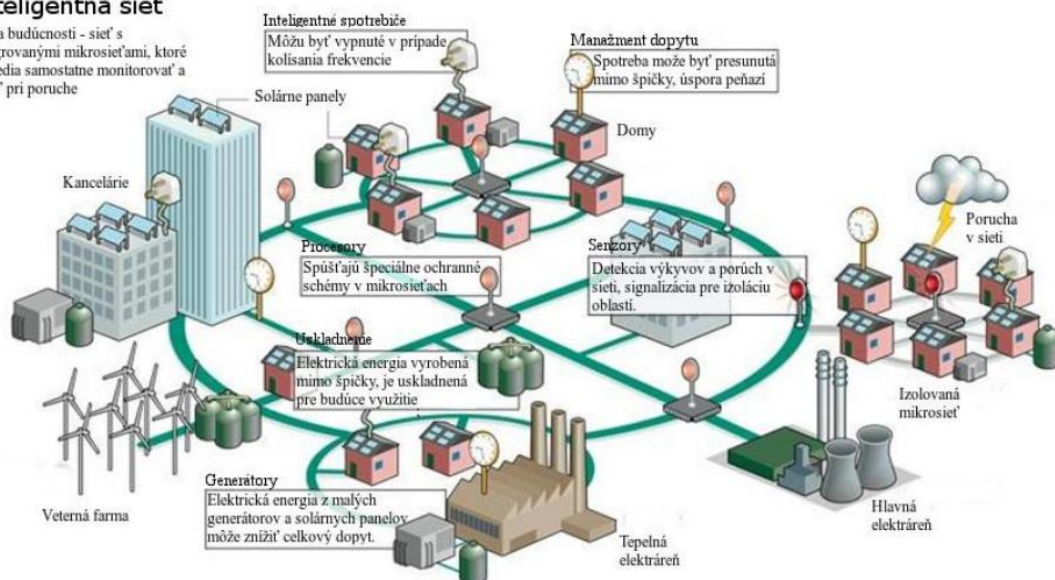
I. TEORETICKÁ ČASŤ

1 Smart Grid

Smart grid (v preklade aj inteligentná sieť) je vlastne prirodzená reakcia odbornej spoločnosti na nárast spotreby elektriny, ktorá je dnes už všade prítomná a jej využitie má stále tendenciu stúpať. Keďže technológie idú dopredu vo všetkých oblastiach a slovo smart je skloňované čoraz častejšie, napríklad v komunikačných technológiách sú to smartfóny, či smart systémy v automobiloch a v iných oblastiach. Je preto zrejmé že tento trend neobíde ani odvetvie energetiky, meracích systémov či systémov orientujúcich sa na optimalizáciu spotreby elektriny. Dôvody zavedenia smart grid technológií do bežného života ľudí je hneď niekoľko. Napríklad z hľadiska ľudí je to nárast populácie, koncentrácia obyvateľstva do miest, zvyšovanie životnej úrovne, väčšie nároky na komfort, nárast energetickej náročnosti, problémy s vodou, odpadmi, parkovaním, ekonomické a iné krízy. Z hľadiska ekológie zasa znečisťovanie životného prostredia, vysoká produkcia CO₂, smog v mestách a priemyselných zónach, výfukové plyny automobilov, globálne otepľovanie, riziká jadrových elektrární, vysoká závislosť od zdrojov fosílnych palív a od ich vlastníkov.

Inteligentná sieť

Vízia budúcnosti - sieť s integrovanými mikrosieťami, ktoré sa vedú samostatne monitorovať a liečiť pri poruche



Obrázok 1 Inteligentná sieť [12]

S súčasnosti sú používané veľké centrálné zdroje a diaľkové energetické vedenia kde vzniká aj riziko znefunkčnenia veľkých území pri výpadkoch nazývaných aj „Blackout“. Takisto je smart grid veľmi prospešný a priam nevyhnutný pri zavádzaní nových technológií ako sú elektromobily, obnoviteľné zdroje a eliminácia ich nežiaducich

následkov na kvalitu elektrickej energie v sieťach. Potreba Inteligentnej siete - Smart Grid je pre budúcnosť priam nevyhnutná na zvládnutie a optimalizáciu energetiky. Inteligentné siete sa plánujú využívať v rôznych odvetviach ako je „Smart Production“ (inteligentná výroba), decentralizácia a mix zdrojov „Smart Distribution“ (inteligentná distribúcia), optimalizácia prenosu elektriny, zlepšenie kvality dodávky elektriny, zníženie poruchovosti, zníženie energetických strát, zvýšenie efektívnosti „Smart Consumption“ (inteligentná spotreba), „Smart Home“, „Smart Cities“ a iné. Základom pre všetky spomenuté odvetvia sú dáta z IMS (inteligentného meracieho systému) – „Smart Metering“, ktorý je základným kameňom optimalizácie týchto procesov. [10]

1.1 SMART GRID v legislatíve

Snaha Európskej únie o prijatie rozumných opatrení na zefektívnenie energetických procesov s ohľadom na životné prostredie, bezpečnosť a stabilitu prenosu, schopnosť regulácie, minimalizáciu nákladov na prevádzku, efektívnosť a spokojnosť zákazníkov docielila, že sa technológie inteligentných sietí presadili aj v zákonoch európskej únie:

- Smernica Európskeho parlamentu a Rady 2009/ 72/ES o spoločných pravidlách pre vnútorný trh s elektrinou, prijatá v rámci 3. energetického balíčka
- Smernica 2012/27/EU o energetickej efektívnosti
- Odporúčanie komisie 2012/148/EU o prípravách na zavádzanie IMS
- V SR Vyhláška č. 358/2013 Z. z. Ministerstva hospodárstva Slovenskej republiky, ktorou sa ustanovuje postup a podmienky v oblasti zavádzania a prevádzky inteligentných meracích systémov v elektroenergetike
- V ČR Národní akční plán pro chytré sítě (NAP SG)

Ministerstvo hospodárstva Slovenskej republiky v spolupráci s Úradom pre reguláciu sieťových odvetví na základe analýz v septembri 2012 rozhodlo, že v súčasných podmienkach bude ekonomicky efektívna realizácia inštalácie inteligentných meracích systémov pre všetky odberné miesta koncových odberateľov elektriny s ročnou spotrebou elektrickej energie viac ako 4 000 kWh, čo skutočne ukazuje že trend zavádzania IMS sa netýka už iba industriálnych veľkoodberov ale hranica klesá až na bežné domácnosti z vyššou spotrebou. Spoločný ročný odber týchto odberných miest je približne 53 % spotrebovanej elektriny na napäťovej úrovni nn a ich počet predstavuje cca 23 % z celkového počtu nn odberných miest, čo činí spolu cca 620 000.

Česká republika v oblasti celoplošného zavedenia inteligentného merania a inteligentných sietí postupovala opatrnejšie. Vláda ČR sa rozhodla zatiaľ neprispôsobovať sa odporúčaniam EU a ponechala zavádzanie inteligentných sietí a meracích systémov na báze dobrovoľnosti distribučných spoločností. Je to hlavne kvôli obavám že celoplošné zavedenie by bolo príliš nákladné a previedlo by sa tým pádom do ceny elektriny koncového zákazníka. Taktiež energetické distribučné spoločnosti argumentujú používaním HDO (hromadného diaľkového ovládania), ktoré by podľa nich tiež malo patriť do funkcionality smart grid. Napriek tomu už teraz je v ČR zavedených a neustále sa zavádza stále väčšie množstvo pilotných projektov v štátnom aj súkromnom sektore.[6]

1.1.1 Vízie smart grid v EU a ich podpora členskými štátmi

Na Slovensku i v Českej republike sa globálne ešte držíme v starom systéme naopak, Nemecko sa pridalo k Rakúsku a do desiatich rokov plánujú odstaviť všetky jadrové reaktory a prejsť na obnoviteľné zdroje. V roku 2020 dosiahnu 35-percentný a v roku 2050 až 80-percentný podiel na celkovej energetickej produkcii. Ako uvádza energetické združenie BDEW, aktuálne sú na 25 percentách.

Keďže vietor a slnko je sa ovládať nedá a produkcia ovplyvnená počasím spôsobuje viacero problémov. Veterné elektrárne na severe Nemecka pätinu času nečinne stoja, pretože nadprodukciou by zaťažili existujúcu sieť. Pomohli by vysokonapäťové prenosné vedenia zo severu na juh s tradičným centrami priemyslu, no stále ich nie je dosť. Aj to je dôvod, prečo sa spomalia stavba veľkých veterných parkov v Severnom mori, hoci sú kľúčovou časťou prechodu na obnoviteľné zdroje. Prebytky prúdu zaťažujú siete v Česku, Poľsku, kritický „červený stav“ ohlásil aj Slovenský operátor. Česká elektrizačná sústava dokonca uvažuje o zariadeniach, ktoré by znížili kapacitu pre prenos elektriny na hranici s Nemeckom, aby ochránila svoju sieť pred preťažením. Bohužiaľ časť nákladov na „zelenú“ energiu, ktoré do konca desaťročia dosiahnu 175 miliárd eur, presúva nemecká vláda na spotrebiteľov. Bežná domácnosť zaplatí ročne za elektrinu o desať eur navyše a povinná podpora obnoviteľných zdrojov zvýši do roku 2020 cenu elektriny pre obyvateľov o tridsať percent. Aj toto je dôvodom prečo sa Česká a Slovenská republika zatiaľ výstavbe inteligentných sietí v takom rozsahu nevenuje. Nemecko a Rakúsko sa vlastne rozhodli prestavať celý svoj energetický potenciál na väčšinovú produkciu zelenej energie a tá sa bez celoplošného zavedenia inteligentných sietí nezaobíde. Na svete už je spustených veľa projektov využívania inteligentných sietí. Sú dokonca aj celé mestá, ktoré vo veľkom

rozsahu tieto siete využívajú. V Zavádzaní IMS môže byť vzorom napríklad Taliansko, ktoré už inteligentné meranie zavádza aj do bežných domácností s nižším odberom. Avšak prestavbe celej štátnej infraštruktúry a vyradeniu atómových elektrární sa okrem týchto krajín v takom rozsahu nikto neodhodlal. Projekt obnoviteľných zdrojov prinesie transformáciu celej energetiky. Tá už 120 rokov ráta so stabilnými zdrojmi a kontrolovanou výrobou. Spotreba je síce nevypočítateľná, no produkcia sa jej vie prispôbiť. Obnoviteľné zdroje tento pomer otočia. Nepredvídateľnou sa stáva časť výroby, preto treba optimalizovať podstatnú časť spotreby. Väčšinu elektriny nebude produkovať niekoľko veľkých elektrární, ale množstvo malých a stredných zdrojov. Z dnešných pasívnych odberateľov sa stanú aktívni hráči – aj mnohé budovy začnú vyrábať a dodávať vlastnú elektrinu do siete. Do hry vstúpia domácnosti so solárnymi článkami a kombinovanými kotlami. Tento „chaos“ dokáže zvládnuť len nová infraštruktúra – smart grids, inteligentné siete. Z jednosmerových sa rozvodné siete zmenia na obojsmerné magistrály, schopné ovládať premenné veličiny v reálnom čase. Ako vraví Jürgen Knaak, šéf energetickej firmy zo švajčiarskeho mestečka Arbon, ide o „ozajstnú revolúciu, porovnateľnú s mobilnými telefónmi či internetom“. Zmena paradigmy vyžaduje nové technológie. Nielen samotné výrobné zariadenia, ako vysoko spoľahlivé veterné elektrárne, ktoré sú, zakotvené ďaleko od pobrežia a zaobídu sa bez údržby či slnečné elektrárne schopné produkovať elektrinu aj v noci. Je nutné nájsť spôsoby, ako uskladniť nevyužitú elektrinu – od stlačeného vzduchu cez prečerpávacie nádrže po výrobu vodíka z vody. Ako najelegantnejšie riešenie sa ukazujú batérie v elektromobiloch, čo bude silný impulz pre elektromobilitu. Úplne sa zmenia aj domácnosti. Dnes presne vieme, koľko nás stál hovor cez mobilný telefón, koľko dát sme stiahli z internetu, ale zistiť dennú spotrebu elektriny v domácnosti a podiel jednotlivých spotrebičov je skoro nemožné. Inteligentné merače nám poskytnú okamžitú odpoveď a napríklad šikovný ohrievač vody sa sám zapne vtedy, keď je prúd najlacnejší. A k tomu je možné mať elektromobil v garáži, bez obáv z ďalšieho rastu cien benzínu a nafty. Nemecko a ďalšie vyspelé krajiny si dnes vytvárajú základ pre čistú, sebestačnú a inteligentnú energetiku. Nejde to bez problémov i nákladov. Ale „smart grids“ predstavujú budúcnosť a kto zvládne potrebné technológie, má šancu na prosperitu.

[15]

1.2 Definícia smart grid

Definícia „inteligentná sieť“ smart grid znamená „zdokonalenú energetickú sieť, ku ktorej bola pridaná obojsmerná digitálna komunikácia medzi dodávateľom a spotrebiteľom, inteligentné meranie, monitoring a riadiace systémy“. Jej základným predpokladom sú správne a komplexné dáta z meraní - zo systému smart metering. Definícia „inteligentný merací systém“ smart metering znamená „elektronický systém, ktorý je schopný merať spotrebu energie a pridávať k tomu viac informácií než konvenčné meradlo (napr. meranie ďalších výkonových a kvalitatívnych parametrov elektriny) a ktorý je schopný vysielat' a prijímať dáta s využitím niektorej formy elektronickej komunikácie“. [10]

1.3 Definícia inteligentného meracieho systému

Podľa vyhlášky 358 Ministerstva hospodárstva Slovenskej republiky z 28. októbra 2013 sa rozumie funkciou inteligentného meracieho systému ďalej len „IMS“ každá vlastnosť, ktorá umožňuje vykonávanie niektorej z činností týkajúcich sa merania odberu a dodávky elektriny a ďalších technických parametrov odberu a dodávky elektriny, prenosu údajov, komunikácie medzi prvkami inteligentného meracieho systému, spracovania a poskytovania nameraných údajov. Požadované parametre na elektromery IMS sú: „3 fázový činný výkon P udávaný v kWh, 3 fázový jalový výkon Q udávaný v kVAr, 3 fázový zdanlivý výkon S udávaný v kVA ktorého súčasťou je okrem činného a jalového výkonu aj deformačný výkon D spôsobený vyššími harmonickými napätia a prúdu“. Centrálou inteligentného meracieho systému je informačný systém prevádzkovateľa distribučnej sústavy, ktorý zabezpečuje zber, zabezpečený prenos, spracovanie a správu nameraných údajov a poskytovanie nameraných údajov informačným systémom prevádzkovateľa distribučnej sústavy. Koncentrátorom inteligentného meracieho systému je zariadenie prevádzkovateľa distribučnej sústavy zabezpečujúce zber nameraných údajov v určenej lokalite prostredníctvom lokálnej dátovej komunikácie. Koncentrátor zabezpečuje súčasne správu komunikačných jednotiek určených meradiel, základné spracovanie dát a odosiela ich do centrály inteligentného meracieho systému priamou komunikáciou prostredníctvom diaľkovej počítačovej siete. [9]

1.3.1 Základné technické parametre Inteligentného meracieho systému

Základnú funkcionality tvorí obojsmerná komunikácia s meracím miestom, priebehové meranie činnnej energie A_p v 15 min intervale, možnosť monitoringu odberu odberateľom, registrácia odberu a dodávky vo viacerých sadzbách, registrácia udalostí neštandardných a poruchových stavov, odpočet a spracovanie údajov najmenej 1x za mesiac. **Pokročilá funkcionality** navyše k základnej pridáva priebehové štvorkvadrantné meranie odberu a dodávky jalovej energie A_q v 15 min intervale, meranie efektívnych hodnôt napätia a prúdu po fázach, spracovanie údajov 1x za deň, možnosť diaľkového odpojenia a pripojenia odberného miesta, prúdové a výkonové obmedzenie odoberaného výkonu, vyhodnocovanie účinníka A_p a $A_q(\cos \phi)$, alarmy poruchy a napadnutia a modulárny komunikačný modul. **Špeciálna funkcionality** k pokročilej pridáva ešte priebehové meranie zdanlivej energie A_s , meranie kvalitatívnych parametrov elektriny a výkonov - aritmetického zdanlivého S , správneho zdanlivého S_r , deformačného D , výkonu nesymetrie N , vyhodnocovanie účinníka P/S a P/S_r , rozhranie na komunikáciu s dispečerským riadiacim systémom. [8]

1.4 Elektromery používané v inteligentných meracích systémoch

Inteligentné elektromery sú vlastne digitálne elektronické zariadenia, ktoré pracujú na statickom princípe. Sú menšie a ľahšie ako bežné mechanické elektromery a ich funkcionality je omnoho rozšírenejšia. Štandardne sa vyrábajú v prevedeniach v plastovom obale a s integrovaným displejom. Na displeji je možné zobrazit' základné údaje o spotrebe, vnútornom nastavení času a dátumu prístroja.



Obrázok 2 Inteligentné elektromery

Na obširnejšiu komunikáciu s elektromerom je však nutné sa pripojiť pomocou PC so špeciálnym prevodníkom. Inteligentné elektromery spravidla obsahujú optické komunikačné rozhranie na ktoré je potrebné mať špeciálny prevodník nazývaný „optická hlava“. Toto rozhranie slúži skôr na parametrizačné a servisné účely. Na komunikáciu medzi elektromerom a koncentrátorom údajov, ktorý ich ďalej zasiela na centrálu distribútora elektriny sú väčšinou elektromery vybavené sériovým komunikačným rozhraním, spravidla priemyselným sériovým rozhraním RS 485. Funkcionalita elektromera je veľmi rozsiahla a závisí od typu, výrobcu a konkrétnych požiadaviek. Najvyššie modely elektromerov spravidla obsahujú rozsiahlu možnosť konfigurácie tarifného nastavenia, meranie 3 fázového výkonu, jalového, činného, zdanlivého, deformačného a to v oboch smeroch pre prípad rekuperácie (výroba elektriny pri brzdení vozidla), či spätnej dodávky elektriny (napríklad pri použití fotovoltaiických panelov v rodinných domoch). Elektromery sú spravidla vyrábané na meranie striedavého prúdu a napätia, no môžu byť použité aj špeciálne elektromery na meranie jednosmerného výkonu. Tie sú spravidla používané v železničných aplikáciách na lokomotívach. Jednosmerná trakčná napájacia sústava prevláda hlavne v štátoch východnej Európy. Ďalšou pridanou vlastnosťou je možnosť merania kvality elektrickej energie. Teda, rôzne výpadky, nesymetrie, rušenia, či záznam vyšších harmonických zložiek v sieti. Inteligentné elektromery majú procesor ktorý dokáže všetky tieto veličiny vypočítať, previesť a zaznamenať v internej pamäti zariadenia. Dokáže zaznamenávať profil spotreby či kvality elektriny v ľubovoľných časových intervaloch a doba záznamu je spravidla viac

ako 3 mesiace. Najväčšou výhodou týchto elektromerov je diaľkový prenos dát, či možnosť rôznych diaľkových nastavení. Obrovskou výhodou je práve možnosť diaľkového odpojenia či zapnutia elektrického odberu zákazníka. Napríklad v prípade neuhradenia faktúry je možné zákazníka odpojiť z centrály a zasa po úhrade faktúry ho zapnúť, čím distribútorovi odpadá povinnosť vyslať zamestnanca na dané miesto, zvlášť keď je odberné miesto neprístupné. Najväčšia výhoda je, že k dátam distribútora elektriny sa môže dostať aj sám zákazník, napríklad cez webový portál distribútora a tým prispieť k optimalizácii spotreby vo svojej domácnosti.

1.5 Existujúce meracie systémy na hnacích dráhových vozidlách

Zavádzanie IMS na verejné odbery, veľké, stredné a malé podniky či domácnosti je dnes už bežná prax. V železničných aplikáciách sa však IMS začal nasadzovať iba nedávno. Hoci isté hnacie dráhové vozidlá ďalej len HDV, meracím systémom vybavené boli už skôr, tento systém slúžil hlavne na rozúčtovanie spotreby elektriny pri prechode do železničnej infraštruktúry iného štátu a teda vybavené nim boli iba určité HDV ktoré za hranicami operovali pravidelne. Zavedenie celoplošného IMS však má v železničnej infraštruktúre veľmi pozitívne prínosy na zníženie celoplošného odberu elektriny na celej železničnej sieti. V celoplošnom merítku bolo nasadené napríklad v Nemecku a skúšobná prevádzka je nasadená aj na Slovensku. Princíp spočíva v tom, že momentálne sa medzi rôzne dopravné spoločnosti rozúčtováva všetka spotreba v rátane strát v sieti, podľa váhy vlaku a prejdenej kilometrov. Tento systém nie je však príliš spravodlivý. Takisto v sieti vznikajú obrovské problémy keď sa v jednom okamihu rozbieha viac ťažkých vlakov. Vedenie je preťažené a železničným spoločnostiam sú účtované pokuty za prekročenie maximálnej rezervovanej kapacity. Vízia využitia IMS je okrem spravodlivého rozúčtovania, kde sa spotreba účtuje priamo za to čo HDV reálne spotrebovalo aj v predikcii preťaženia siete. Podľa dát z IMS je možné informovať dopravných dispečerov o aktuálnom preťažení v danej oblasti a tým sa vyhnúť poruchám na vedení a nemalým sankciám. V budúcnosti by sa IMS dal využiť aj na priamu reguláciu spotreby. Nové HDV sú už vybavené rekuperačným brzdným systémom, ktorý pri brzdení vyrába elektrinu a teda žiaduce v sieti je synchronizovanie rozbehu jedného vlaku a brzdenia druhého. Práve k tomuto by mohol dopomôcť IMS v železničnej oblasti. Aktuálne je napríklad na Slovensku a v Českej republike používaný systém obsahujúci Univerzálny elektromer

ktorý meria ako striedavú, tak jednosmernú časť spotreby. Ten je prepojený s GPS anténou, pretože okrem spotreby elektrickej energie je nutné mať aj informáciu o polohe aby bolo jednoznačne určené s ktorej transformačnej stanice práve dané HDV odoberá elektrinu. V momentálne nasadených aplikáciách sa stará o zber dát s elektromerom a GPS antény špeciálny priemyselný počítač s dátovou zbernicou RS485, ktorý má v sebe integrovaný GPRS modem a pomocou neho odosiela zozbierané dáta na centrálu distribútora. Nasadenie tohto priemyselného PC je však finančne veľmi náročné a často nerentabilné. Tento PC často krát prevyšuje viacnásobne cenu elektromera a na jeho parametrizáciu treba odborné vedomosti. Taktiež, má zariadenie malú pamäť a je pomalé. Práve toto zariadenie a jeho základnú funkcionálnosť sa táto diplomová práca snaží nahradiť mobilným telefónom. Dnes mobilné telefóny takmer nahrádzajú svojim výkonom bežne stolové počítače. Mobilný telefón má v sebe operačný systém, integrovaný dátový modem, je rýchly a má veľkokapacitnú pamäť. Má USB port cez ktorý sa dá pomocou prevodníka USB/RS 485, odčítavať elektromer a popríklad iné zariadenia. Operačný systém Android, má veľmi dobrú podporu pre tvorbu aplikácií, pri ktorých sa na programovanie používa všeobecne známy programovací jazyk Java. Najväčšia výhoda spočíva v tom že telefóny so systémom Android sú k dispozícii za niekoľko násobne nižšie ceny ako momentálne používaný priemyselný počítač.

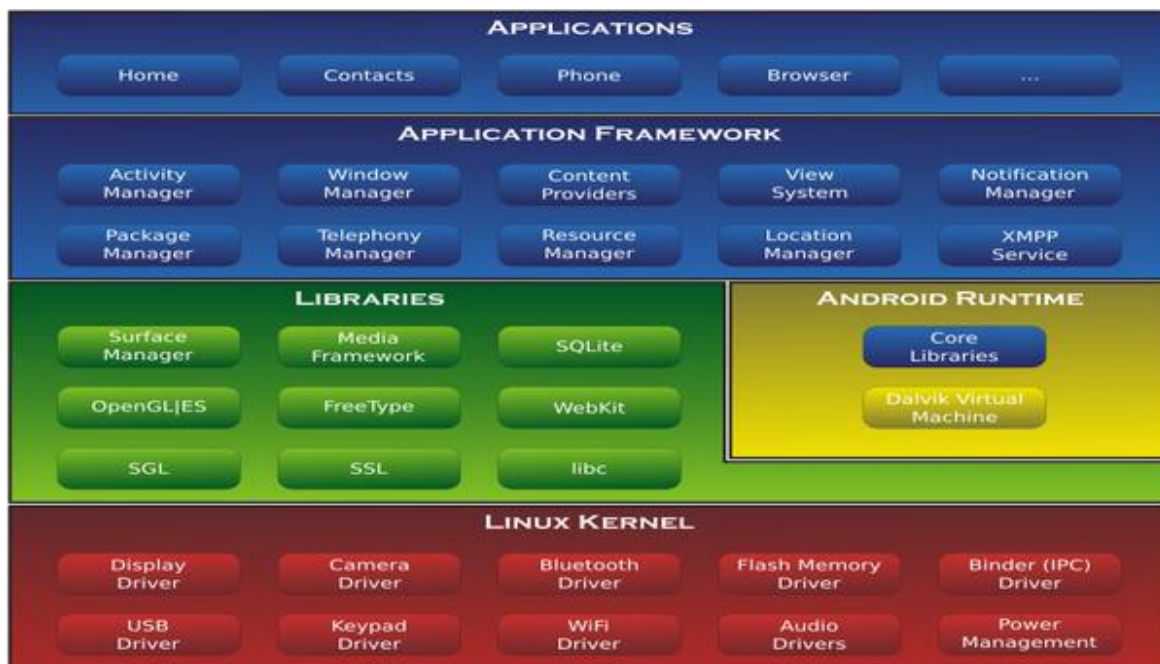
2 Použité technológie

Táto kapitola opisuje použité technológie, protokoly, programovacie jazyky a operačné systémy ktoré boli v práci použité.

2.1 Operačný systém Android

Operačný systém Android je prvou bezplatnou, otvorenou, mobilnou platformou, ktorá je určená najmä pre mobilné zariadenia (inteligentné mobilné telefóny, tablety a PDA). Táto platforma obsahuje operačný systém (založený na jadre systému Linux), middleware, užívateľské rozhranie a aplikácie. Za vývojom stojí konzorcium Open Handset Alliance (čo je skupina 84 technologických a mobilných spoločností, medzi ktorých patria giganti ako Google, HTC, Intel, LG, Motorola, NVidia, Qualcomm, Samsung, Texas Instruments a ďalšie). Cieľom tejto skupiny je progresívny rozvoj mobilných technológií, ktoré budú mať výrazne nižšie náklady na vývoj, distribúciu a zároveň spotrebiteľom prinesú inovatívne užívateľsky prívetivé prostredie. Pri vývoji systému sa do úvahy zobrali obmedzenia, ktoré sú príznačné pre klasické mobilné zariadenia. Napríklad výdrž batérie, menšia výkonnosť a obmedzené množstvo dostupnej pamäte. Zároveň bolo jadro systému Android navrhnuté pre beh na rôznom hardvéri. Systém možno teda používať bez ohľadu na použitý chipset, veľkosť či rozlíšenie obrazovky. Samotná mobilná platforma Android dáva k dispozícii nielen operačný systém s užívateľským prostredím pre koncových užívateľov, ale aj kompletné riešenie nasadenia operačného systému (špecifikácia ovládačov atď.) Pre mobilných operátorov, výrobcov zariadení a v neposlednom rade pre vývojárov aplikácií poskytuje efektívne nástroje pre ich vývoj - SDK (Software Development Kit). Najnižšia vrstva architektúry je jadro operačného systému, ktoré tvoria abstraktné vrstvy medzi používaným hardvérom a zvyškom softvéru vo vyšších vrstvách. Jadro mobilného operačného systému Android je postavené na platforme Linux vo verzii 2.6. Využíva celej rady jeho vlastností, napríklad podpory správy pamäte, správy sietí, zabudovaných ovládačov alebo aj správy procesov, súbežného behu aplikácií, ktoré bežia ako samostatné procesy s oprávnenie definovaným systémom, čo prispieva k jeho stabilite. Naopak systém Android nepodporuje grafické užívateľské rozhranie „X Window System“ a ani úplnú sadu GNU knižníc. Dôvodom použitia jadra systému Linux bola tiež vlastnosť pomerne jednoduchého zostavenia na rôznych zariadeniach a tým zaručená prenositeľnosť. Ďalšou vrstvou sú knižnice, ktoré sú

napísané v jazyku C / C ++, a ktoré využívajú rôzne komponenty systému. Tieto funkcie sú vývojárom poskytnuté prostredníctvom „Android Application Framework“. Architektúra operačného systému Android je rozdelená do piatich vrstiev: jadro, knižnice, aplikačný framework, runtime prostredie a aplikácie. Každá vrstva má svoj účel a nemusí byť priamo oddelená od ostatných vrstiev.



Obrázok 3 Architektúra OS Android [1]

Android Runtime vrstva obsahuje virtuálny stroj označovaný ako Dalvik, ktorý bol vyvíjaný od roku 2005 špeciálne pre platformu Android, tímom v spoločnosti Google pod vedením Dana Bornstein. Dalvík Virtual Machine (DVM) je orientovaná architektúra, využíva základných vlastností linuxového jadra, napríklad správu pamäte alebo prácu s vláknami. Vznik nového virtuálneho stroja bol iniciovaný z dvoch dôvodov. Prvým dôvodom boli licenčné práva, kedy jazyk Java a jeho knižnice sú voľne šíriteľné a Java virtual machine nie. Ďalším dôvodom bola optimalizácia virtuálneho stroja pre mobilné zariadenia a to predovšetkým v oblasti pomeru úspory energie a výkonu. V tejto vrstve sú tiež obsiahnuté základné knižnice programovacieho jazyka Java. Knižnice sa svojím obsahom blížia platforme Java Standard Edition. Hlavný rozdiel je v neprítomnosti knižníc pre užívateľské rozhranie (AWT a Swing), ktoré boli nahradené knižnicami užívateľského rozhrania pre Android alebo prídanie knižnice Apache pre prácu so sieťou. Preklad aplikácií napísaných pre Android prebieha skompilovaním zdrojového kódu v jazyku Java do Java byte kódu pomocou rovnakého kompilátora, ako je používaný v prípade preklade Java aplikácií. Potom sa prekompiluje Java byte kód pomocou Dalvik kompilátora a

výsledný Dalvik byte kód je spustený na DVM. Každá spustená Android aplikácia beží vo svojom vlastnom procese s vlastnou inštanciou DVM. Skupinu prvkov View je možné používať pre nastavenie užívateľského rozhrania aplikácií, ide napríklad o zoznamy, textové polia, tlačidlá, zaškrŕavacie políčka a ďalšie. Content providers umožňuje aplikáciám prístup k dátam iných aplikácií (napríklad Kontakty) alebo zdieľanie vlastných dát. Resource manager poskytuje prístup neprogramovým zdrojom, ako sú lokalizačné reťazce, grafika a súbory dizajnu. Notification manager umožňuje všetkým aplikáciám zobrazit' vlastné upozornenie v stavovom riadku. Activity manager riadi životný cyklus aplikácií a poskytuje orientáciu v zásobníku s aplikáciami. Verzie systému Android:

- Android 1.5 Cupcake (Prvý OS Android. Položil základ pre všetky ostatné verzie.)
- Android 1.6 Donut (Zavedenie Android obchodu s aplikáciami a vyhľadávacej lyšty.)
- Android 2.1 Eclair (Google maps, preklad textu do reči, viaceré pracovné plochy.)
- Android 2.2 Froyo (Wifi hot spot, zavedenie Dalvik runtime.)
- Android 2.3 Gingerbread (Zavedenie Google API súborov, podpora NFC.)
- Android 3.0 Honeycomb (Verzia len pre tablety, zavedenie softvérových tlačidiel.)
- Android 4.0 Ice Cream Sandwich (Kontrola spotreby dát, Android BEAM.)
- Android 4.1 Jelly Bean (Google Now, upozornenia, používateľské účty.)
- Android 4.4 KitKat (Full screen režim, inteligentný dialer.)
- Android 5.0 Lollipop (Material dizajn, notifikácie na uzamknutej obrazovke.)
- Android 6.0 Marshmallow (Zatiaľ používaný iba v zariadeniach Nexus.)

[1]

2.2 Programovací jazyk Java

Vývoj jazyka a platformy Java začal ako projekt firmy Sun Microsystems. Jeho vývojári boli sklamaní so stavom ich verzie C++ a aplikačným programovým rozhraním (API) C++ a ich nástrojov. Pokúsili sa vyvinúť programovaciu technológiu novej generácie, a využiť tak nové možnosti. Prvá verejná verzia bola vydaná v roku 1995 ako Java 1.0. V tom roku Bill Gates tvrdil, že ďalší programovací jazyk je zbytočný a predpokladalo sa, že Java zanikne do dvoch rokov. Príkladom tohto veľkého omylu sú internet-banking systémy najväčších bánk, ktoré sú postavené práve na platforme Javy alebo burzové a clearingové systémy väčšinou stojace na Jave a stále rastúca komunita programátorov v tomto jazyku. Java je objektovo orientovaný programovací jazyk, rozsiahla počítačová technológia a počítačová platforma. Tento jazyk syntaxou vychádza z

jazyka C++. Oproti C++ však Java neobsahuje žiadne zložité konštrukcie (ako sú napríklad ukazovatele), ktoré spôsobujú zbytočné problémy. Vďaka automatickej správe pamäti (garbage collector) sa programátor nemusí starať o manuálne čistenie pamäti po nepotrebných objektoch. Nahradením ukazovateľov (pointers) odkazmi (references) je ukončená hrozba zápisu do neplatnej pamäti. Mechanizmus vlákien umožňuje tvorbu prepracovaného grafického užívateľského rozhrania (GUI). Navyše Java obsahuje serializáciu, ktorá prináša elegantné riešenia v prípade, že je potreba ukladania dát do súborov alebo ich prenášanie po sieti. Štandardne dodávané knižnice umožňujú bez ďalších doplnkov okamžite vytvárať užívateľské rozhrania, pracovať so súbormi, textom, databázami, komprimovanými súbormi, predvoľbami a ďalšími časťami počítačového prostredia. Programy napísané v Jave sú prenositeľné na každú platformu (Windows, Unix, Linux alebo Solaris), ktorá obsahuje virtuálny stroj jazyka Java (Java Virtual Machine – JVM). Kód je kompilovaný len raz na bajtový kód (byte code), ktorý je pri každom spustení interpretovaný virtuálnym strojom. [2] Sun Microsystems vytvoril slogan „napíš raz, spusti všade“ (write once, run everywhere), aby demonštroval hlavnú výhodu jazyka Java. Ako už bolo vyššie spomenuté, JVM spracováva (interpretuje) bajtový kód. Dostupnosť JVM na rôznych druhoch hardwarových a softwarových platformách umožňuje, aby bola Java funkčná nielen ako platforma, ale aj ako middleware (middleware pozostáva zo softwarového agenta, ktorý komunikuje ako medzičlánok medzi rôznymi aplikáciami). Týmto spôsobom sa dosiahne prenositeľnosť kódu na systémy s odlišnou architektúrou, čiže na rozdiel od programu, ktorý je napísaný v jazyku C alebo C++, sa nemusí kód kompilovať zvlášť pre každú architektúru. Napriek tomu, že bajtový kód je rovnaký pre všetky platformy, mechanizmus JVM interpretuje a optimalizuje aplikáciu priamo na hostiteľskom počítači. Java je v podstate kombináciou štyroch častí:

- Programovací jazyk Java
- Formát súboru .class
- Java API
- JVM

Vyvíjanie nejakej aplikácie je písanie kódu v programovacom jazyku Java. Ten sa následne skompiluje do súborov s príponou .class, ktoré sú spúšťané v prostredí JVM. Kombinácia JVM s triedami výkonného jadra jazyka Java je známa aj ako prostredie pre spracovanie jazyka Java (Java Runtime Environment – JRE). [4] Java API je dopredu

pripravený kód, ktorý je tematicky usporiadaný do balíčkov (packages) a delí sa na tri základné platformy:

Java Micro Edition (JME) platforma na vývoj aplikácii napr. pre mobilné telefóny, systémy navádzania motorových vozidiel...,

Java Standard Edition (JSE) platforma obsahujúca základnú sadu tried, ktoré sú potrebné pre tvorbu GUI a štandardných aplikácii,

Java Enterprise Edition (JEE) platforma so sadou tried a rozhraní pre vývoj webových aplikácii (napr. Java Server Pages – JSP, Enterprise Java Beans – EJB),

JDK je produkt firmy Sun určený pre programátorov. Je to balíček, ktorý pozostáva z Java kompilátora, množstva vývojových nástrojov a JRE. Niekedy sa označuje aj ako Java SDK (Software Development Kit).

Jednou z najvýznamnejších vlastností Javy je garbage collector. Tento mechanizmus vyhľadáva v oblasti pracovnej (dynamickej) pamäti neplatné alebo nepotrebné objekty, na ktoré napr. už neexistuje platný odkaz alebo sú dlhšiu dobu nepoužívané. Keby táto pamäť nebola uvoľnená správne (načo dopláca veľa programátorov v C++), došlo by k vyčerpaniu pamäťových zdrojov (memory leak).

Programy vyvinuté v Jave sa všeobecne rozdeľujú na takzvané applety a štandardné aplikácie. Applety poskytujú interaktívne vlastnosti pri prezeraní webovým aplikáciám, ktoré nemôže zabezpečiť HTML. Sú spúšťané väčšinou internetových prehliadačov v sandboxoch (ochranný mechanizmus pre bezpečné vykonávanie programov), ktoré zabráňujú prístupu k lokálnym dátam. Kód appletu je najskôr stiahnutý z webového serveru a potom prehliadač buď vloží applet do stránky, alebo ho otvorí v novom okne. Medzi štandardné aplikácie patria napr. konzolové alebo GUI aplikácie. [3]

2.3 Jazyk XML

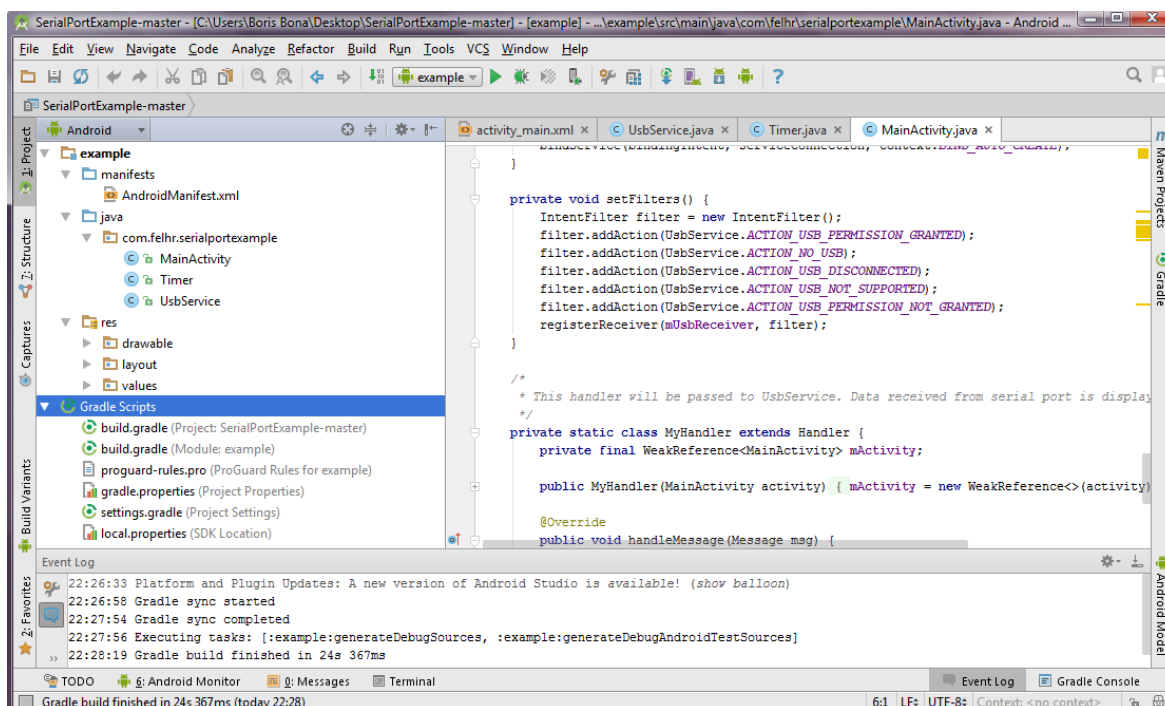
XML je skratka z anglického eXtensible Markup Language, rozšíriteľný značkovací jazyk. V skutočnosti je XML tak zvaný Metajazyk, nadradený značkovací jazyk, v rámci ktorého je možné vytvárať vlastné jazyky (definované pomocou DTD opisu). Takým jazykom je napríklad XHTML, kombinácia XML a HTML. XML je zaujímavý tým, že neobsahuje žiadne konkrétne značky (elementy), ktokoľvek si teda môže vymyslieť vlastné značky, napr. <Farba> zelená </ farba>. To umožňuje veľmi dobre definovať presnú štruktúru každého XML dokumentu podľa aktuálnej potreby. XML je tak

akýmsi predelom medzi databázovou štruktúrou a textovým dokumentom. XML na rozdiel od známejšieho jazyka HTML vôbec neobsahuje informácie o spôsobe zobrazenia, dochádza tak k úplnému oddeleniu formy od obsahu. Vďaka tomu je XML oveľa flexibilnejší, umožňuje vlastnú voľbu zobrazenia každej aplikácii, ktorá s XML dokumentom pracuje. Ak je napriek tomu potreba jednotný vzhľad, možno ho definovať pomocou špeciálnych Stylesheet (kaskádové štýly alebo XSL), ktoré sú pripojené v záhlaví XML dokumentu. XML sa dnes používa predovšetkým pre jednoduchú výmenu informácií (napríklad Výmenu faktúr) a komunikáciu nezávislú na konkrétnej aplikácii či platforme. Hlavnými výhodami XML oproti iným formátom používaným na prenos informácií je jeho nezávislosť, šandardizácia, pomerne malá veľkosť, podpora národných kódovaní a jednoduchý prevod na iné formáty. [1]

2.4 Vývojové prostredie Android Studio

Android Studio je vývojové prostredie IDE (Integrated Development Environment) pre tvorbu aplikácií na operačný systém Android. Bolo predstavené na konferencii I/O v San Franciscu v roku 2013. K dispozícii je už teraz a to v balíku s Android SDK. Android Studio je založené na technológii IntelliJ IDEA a podobne ako populárny IDE Eclipse s pluginom ADT prináša plnohodnotné integrované vývojárske prostredie. Nechýbajú nástroje pre testovanie a ladenie aplikácií a vývojári by podľa zástupcov Google mali oceniť napríklad integrované nástroje pre profiling, testovanie kompatibility, podpisovanie aplikácií alebo wizardy založené na šablónach, ktoré uľahčujú tvorbu aplikácií s bežným dizajnom. Za zmienku stojí aj prepracovaný editor s možnosťou navrhovať užívateľské rozhranie metódou drag-and-drop. Inštalácia Android Studio je možná na platformách: Windows, Linux a MAC OS X. Inštalácia Android štúdia oproti doteraz používanému Eclipse, je veľmi jednoduchá. Stiahnete si inštalčný balík pre svoj systém a JDK.. Pre inštaláciu je obvykle vyžadovaná inštalácia balíka od Java (Oracle). Žiadne ďalšie inštalčné balíky ani inštalácia pluginov sa nevyžaduje. Súčasťou inštalácie je aj Android SDK Tools, kompilátor pre OS Android a základné emulátory s plnohodnotným systémom Android. Medzi najvýznamnejšie funkcie by sa dali zaradiť integrované nástroje pre profiling, testovanie kompatibility, podpisovanie aplikácií, wizardy založené na šablónach a prepracovaný editor s možnosťou navrhovať používateľské rozhranie pomocou drag-and-drop. V Android Studiu možno navrhovať dizajn aplikácie buď v XML alebo v dizajn móde. Keď sa píše priamo v XML móde štúdio

automaticky zobrazuje náhľad vo vybranom rozlíšení zobrazovacieho displeja. Štúdio ponúka aj možnosť "Preview All Screen Sizes", čo zobrazuje náhľady vo všetkých možných rozlíšeníach. Ďalšou možnosťou je Preview Representative Sample, ktorý zobrazí 4 najdôležitejšie. Presnosť v dizajne módu je veľmi vysoká a vkladanie prvkov je veľmi jednoduché. Celé IDE sa prispôsobuje veľkosti okna tzn. náhľady sa zmenšujú / zväčšujú, sú vedľa seba, pod sebou, alebo paleta prvkov, ak má miesto, sa automaticky zobrazí vo viacerých stĺpcoch atď. Android Studio je celé späté s buildovacím ("zostavovacím") nástrojom Gradle. Súčasťou štúdia sú aj emulátory pre Nexus 4, 7 a 10.



Obrázok 4 Vývojové prostredie Android Studio [13]

Emulátor si môžete ďalej nakonfigurovať. Môžete zmeniť Android API, veľkosť RAM v emulátore, veľkosť úložiska a SD karty alebo prednú či zadnú kameru, ktorú je možné buď úplne emulovať alebo ju napojiť na kameru počítača. Nevýhoda emulátorov spočíva v tom, že sú pomalé aj na relatívne rýchlom počítači. Veľkou výhodou je aj integrovaný našeptávač kódu pre Android. Keďže dané vývojové prostredie je modernejšie a má oproti prostrediu Eclipse mnoho výhod bolo použité aj v tejto diplomovej práci. [13]

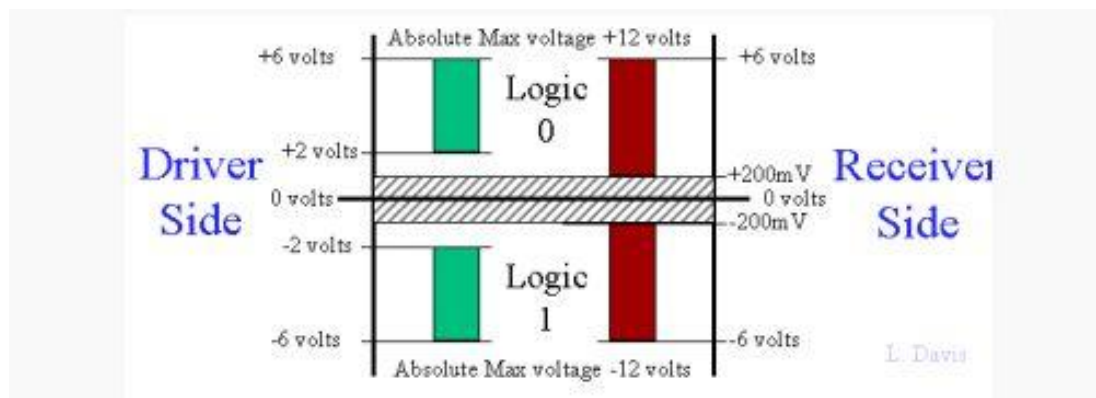
2.5 Komunikačné rozhranie RS 485

Podľa normy STN EN 62056-21 je RS-485 (EIA-485 alebo RS485) špecifikácia dvojdrátového poloduplexného multibodového sériového spojenia, ktorý definuje elektrické charakteristiky vodičov a prijímačov na použitie v digitálne vyvážených systémoch. Prenosový komunikačný štandard RS-485 bol definovaný v roku 1983 inštitúciou EIA (Electronics Industries Association). RS485 sa vyznačuje dvojvodičovým prepojením jednotiek. Tieto vodiče sa označujú písmenami A a B, niekedy sa používa označenie „-“ a „+“ (v praxi často býva vodič A kladný). Maximálna dĺžka zbernice je až 1200 m, maximálny počet vysielateľov a prijímačov je 32 a podľa použitého ovládača môže byť počet účastníkov až 128. Maximálna prenosová rýchlosť je 10 Mb/s (zo stúpajúcou dĺžkou vedenia prenosová rýchlosť klesá). Zbernica musí byť zakončená rezistormi alebo aj takzvanými terminátormi o veľkosti $110\ \Omega$ z každej strany kvôli eliminácii odrazov signálu. Výsledná impedancia vedenia musí byť $R = 55\ \Omega$ ($110\ \Omega \parallel 110\ \Omega$).

RS-485	
Štandard	EIA RS-485
Fyzické Medium	Twisted Pair – Krútená dvojlinka
Topológia siete	Point-to-point, Multi-dropped, Multi-point
Maximum zariadení	32 (32 vodičov a 32 prijímačov)
Maximálna vzdialenosť	1200 metrov
Režim prevádzky	Diferenciálny
Maximálna prenosová rýchlosť	100Kbps - 10 Mb / s
Napäťové úrovne	-5V to +5V (max) -5V až +5 V (max)
Dostupné signály	Tx+, Tx-, Rx+, Rx- (Full Duplex) Tx +, Tx-, Rx +, Rx- (Full Duplex)
Typy konektorov	Nie je uvedené, všeobecne skrutkovacie svorky

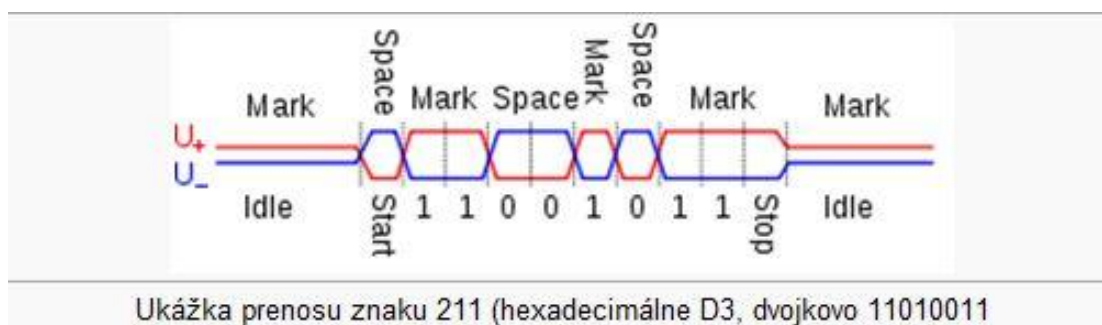
Obrázok 5 Parametre RS 485 [7]

Logické stavy sú reprezentované rozdielnym napätím medzi oboma vodičmi. Je to rozdiel oproti RS-232, kde sa úrovne stavov vzťahujú k referenčnému napätiu, najčastejšie k zemi (GND). Rozdielne napätie medzi oboma vodičmi je výhodné, kvôli eliminácii naindukovaného rušivého napätia vzťahujúceho sa k nulovému potenciálu zeme. Logický stav „1“ označovaný ako „ON“ je reprezentovaný rozdielovým napätím $A - B < -300\text{ mV}$, logický stav „0“ označovaný ako „OFF“ rozdielovým napätím $A - B > +300\text{ mV}$. Správny vysielateľ by mal na výstupe generovať napätie $+2\text{ V}$ (prípadne -2 V), správny prijímač by mal na vstupe rozlíšiť napätie $+200\text{ mV}$ (prípadne -200 mV).



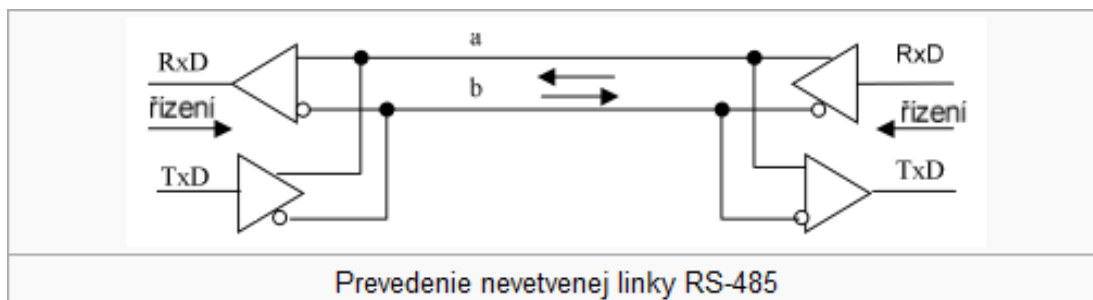
Obrázok 6 Napät'ové úrovne RS 485 [7]

Prenos dát pozostáva s pravidiel nastavenia. Ako prvý je vysielaný štart bit, potom 8 bitou počnúc LSB, bez parity a nakoniec stop bit.



Obrázok 7 Prenos znaku 211 [7]

Dvojvodičová verzia RS 485 je používaná najčastejšie. Prenos dát sa uskutočňuje pomocou 7 alebo 8 bitových rámcov so štart bitom, 1 alebo viacerých stop bitov a prípadne i paritným bitom. Prenos je polo-duplexný preto sa vyžaduje riadenie prenosu dát (smeru komunikácie). Výhodou je, že pomocou dvojvodičovej linky RS485 je možné vytvoriť komunikačnú sieť. Na spoločnú komunikačnú linku môže byť obvykle pripojených bez opakovača až 32 zariadení. V skutočnosti sa pri spojení na dlhšie vzdialenosti musí okrem signálových vodičov (RxTx+ a RxTx-) pripojiť i zem (GND, G) komunikujúcich zariadení. Príčinou je, že vo vzdialených miestach môžu existovať značné rozdiely v potenciály "zeme". Preto je "dvojvodičová" RS-485 v skutočnosti skôr trojvodičová. Pri tejto diplomovej práci je však použité iba krátke vedenie a vodič GND nemusí byť zapojený.



Obrázok 8 Dvojvodičová linka RS 485 [7]

Štvorvodičová verzia RS 485 je menej používaná no v niektorých aplikáciách sa používa. Poskytuje plne-duplexnú (dvojsmernú) komunikáciu a odpadá tak nutnosť riadenia smeru prenosu dát. V podstate ide o dve dvojvodičové linky. Výhodou je logická kompatibilita so základnými signálmi linky RS232. Nevýhodou je, že vykonávané spojenie je typu 1:1 (rovnako ako u RS-232). V praxi sa u štvorvodičovej linky používa aj spojenie 1: N, čo predpokladá že slave zariadenia majú schopnosť odpájať svoj vysielač kanál. Na takejto linke je väčšinou jedno zariadenie typu master, ktoré posiela po vysielačkej linke príkazy a N zariadenia typu slave, ktoré prijímajú príkazy a vysielaajú odpovede. Výhodou je, že master nepotrebuje prepínať smer linky a aj časové požiadavky na prepínanie linky a na vyhodnocovanie prichádzajúcich dát sú u zariadení typu slave miernejšie. Súčasne nehrozí, že by slave zariadenia napríklad v dôsledku chyby softvéru mohli zablokovať príkazový kanál celej zbernice. V skutočnosti sa pri spojení na dlhšie vzdialenosti musí okrem signálových vodičov (Rx +, Rx-, Tx +, Tx-) prepojiť aj zem (GND) komunikujúcich zariadení. Príčinou je tak ako aj v dvojvodičovom prevedení, že vo vzdialených miestach môžu existovať značné rozdiely v potenciály "zeme".

Určitú "dierou" v štandarde RS-485 je, že na rozdiel od RS232 neboli definované štandardné konektory (a teda ani štandardné zapojenie konektorov). Vzhľadom k aplikácii v priemysle, kde sa môžu používať v závislosti na prostredí rôzne systémy konektorov by to bolo pravdepodobne nepraktické. Absencia štandardizácie konektorov pravdepodobne prispela k tomu, že niektorí výrobcovia chybne prehadzujú signály A a B. Ďalším problémom je, že označenie RS-485 a RS-422 bývajú kvôli svojej podobnosti nesprávne používané. Veľká podobnosť RS-485 a RS-422 vedie k tomu, že sú tieto termíny používané ako synonymá, v lepšom prípade sa na RS-422 pozerá ako na podmnožinu RS-485. Kompatibilita týchto štandardov však nie je stopercentné. Komponenty vyhovujúce RS-485 môžu byť väčšinou bez problémov používané v sieťach založených na RS422, naopak to však neplatí. Komponenty RS-422 nemajú signál pre odpojenie výstupu. V

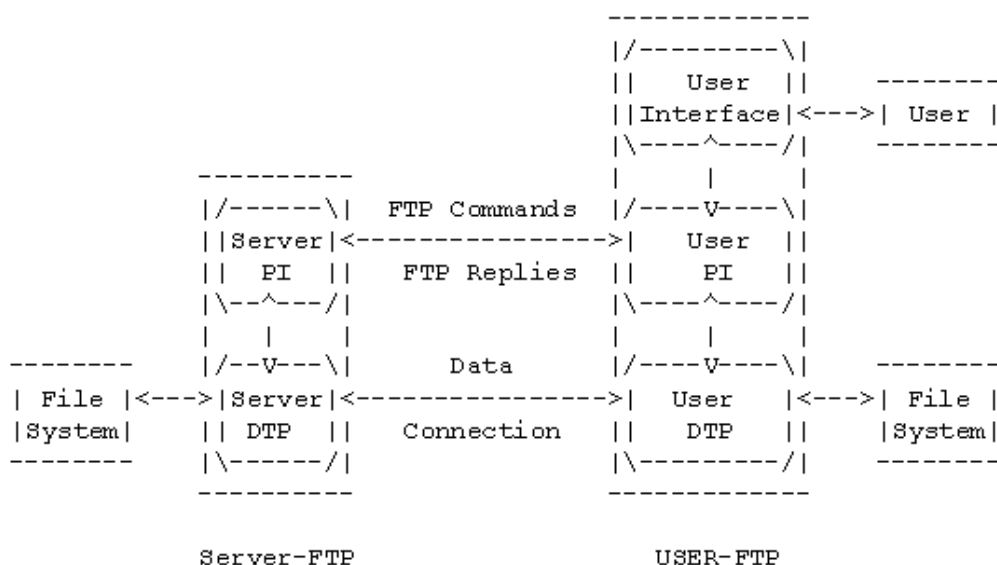
sieťach RS422 môže byť teda použitý iba jediný vysielateľ, komponenty RS422 nie sú odolné voči konfliktu na zbernici a počítajú s jednosmernou linkou, preto majú slabšie budiče, ktoré nezvládnu sieť s dvoma terminátormi. V sieťach RS-485 je možné používať komponenty RS-422 iba ako prijímače, avšak je potrebné vziať do úvahy že je pri vstupnom odpore 4 kOhm musíme 1 jednotku RS-422 počítať za 3 jednotky RS-485. Prakticky sa väčšinou používajú súčiastky RS-485 kompatibilné s oboma štandardmi. Príkladom toho je aj elektromer LEM používaný v železničných aplikáciách. [7]

2.6 FTP protokol

Na prenos zozbieraných údajov medzi telefónom a centrálou distribútora bude v tejto diplomovej práci použitý FTP (File Transfer Protocol). Je to známi protokol, ktorý sa používa na prenos súborov cez Internet. Protokol FTP sa bežne používa pre prístup k súborom na prevzatie pre iné osoby, môže sa však používať aj na odovzdávanie webových stránok pri vytváraní webových lokalít alebo na umiestnenie digitálnych fotografií. Sú podporované dva typy prenosu: ASCII a binárny. Prvý sa používa pri prenose textových súborov, druhý pri prenose programov, údajových a skomprimovaných súborov. Služba je poskytovaná mechanizmom klient - server. Celý prenos sa uskutočňuje interaktívne a je ovládaný používateľom na lokálnom počítači pomocou jednotlivých príkazov klient programu FTP. Na komunikáciu sa využívajú dva porty - 20 a 21. Port 20 slúži na prenos dát a port 21 slúži na kontrolu dát a ftp príkazy. FTP je spoznatelný tak, že vidíme namiesto „HTTP“ na začiatku adresy „FTP“, takže nejde o www-server ale o FTP-server. Napríklad: ftp://meno:heslo@host:port/adresar .

Na začiatku je ftp, potom nasleduje prihlasovacie meno, heslo, host - IP adresa servera alebo napr. ftp.server.com prípadne www.server.com, na konci môže byť cesta do daného adresára. Na serveri je možné vytvoriť viacej používateľských kont, nastaviť práva pre používateľov. Niektoré FTP sú anonymné a tak nie je nutné zadávať meno a heslo, postačí len adresa servera. FTP využíva na prenos súborov netransparentný spôsob, ktorý si používateľ musí uvedomovať. Používateľ musí sám podniknúť určité akcie, aby získal prístup k vzdialeným súborom. Ak chce pracovať s nejakým súborom, musí si ho sám a explicitne najprv preniesť na svoj počítač, a potom spracovať rovnakým spôsobom, ako ktorýkoľvek iný miestny súbor. Ide o prenos celých súborov, označuje sa ako file transfer. V rámci TCP/IP je k prenosu určený FTP (File Transfer Protocol) a TFTP (Trivial FTP). FTP umožňuje jednoduchým spôsobom pracovať so súborami na serveri z lokálnej stanice.

Jeho prostredníctvom možno prenášať súbory medzi lokálnymi stanicami a serverom, premenovávať súbory, mazať ich, spravovať adresárovú štruktúru. Protokol dovoľuje sťahovanie (download) a nahrávanie (upload) binárnych či textových súborov na FTP server. Takto umiestnené súbory sú potom dostupné každému (anonymný FTP server) alebo jednému užívateľovi či skupine (FTP server na heslo). Prístup na FTP server sa deje pomocou FTP klienta. Protokol slúži väčšinou ako doplnok k serveru HTTP, kde sa používa na updatovanie a uploadovanie webových stránok. Ďalej sa používa v sharewarových kruhoch ako veľmi populárny spôsob prenášania programov a súborov v sieti Internet. Používa sa aj vo veľkých podnikových sieťach Intranet, kde slúži ako rýchly a lacný prostriedok prenášania dôležitých dokumentov. Pomocou FTP protokolu spolu komunikuje FTP klient a FTP server. FTP klient sa skladá z užívateľského rozhrania (User Interface), užívateľského interpretu protokolu (User PI – Protocol Interpreter) a užívateľského procesu k prenášaní dát (User DTP – Data Transfer Process). Užívateľ komunikuje s klientom pomocou používateľského rozhrania. Dátové spojenie nemusí existovať po celú dobu. [14]



Obrázok 9 Schéma komunikácie FTP protokolu [14]

II. PRAKTICKÁ ČASŤ

3 Technické prostriedky

Táto kapitola sa zaoberá technickým opisom použitých komponentov potrebných pre správnu funkciu vyvinutej aplikácie ELMReader, ktorej cieľom je odpočet a nastavenie základných dát statických smart elektromerov.

3.1 Použitý mobilný telefón

Na prácu bol použitý mobilný telefón Samsung Galaxy Xcover3. Jedná sa o odolný outdoorový telefón. Odolnosť podľa certifikácie IP67 zaisťuje zvýšenú odolnosť proti vode a prachu. Certifikácia zaručuje, že telefón vydrží umiestnenie pod vodou v hĺbke 1 meter po dobu 30 minút. Je taktiež vhodný do silne zaprášeného prostredia. Tento model bol vybraný aj kvôli možnému nasadeniu a testovaniu aplikácie na vonkajších meracích bodoch, ale aj možnému testovaniu v strojovni lokomotívy v riadnej železničnej prevádzke. V strojovni lokomotívy dochádza k veľkým tepelným výkyvom. Prostredie je prašné, hlavne pre rozptýlený uhlíkový prach z trakčných motorov. Taktiež pri nasadení na vonkajších trafostaniciach môže dôjsť k styku s vodou pri atmosférických zrážkach.



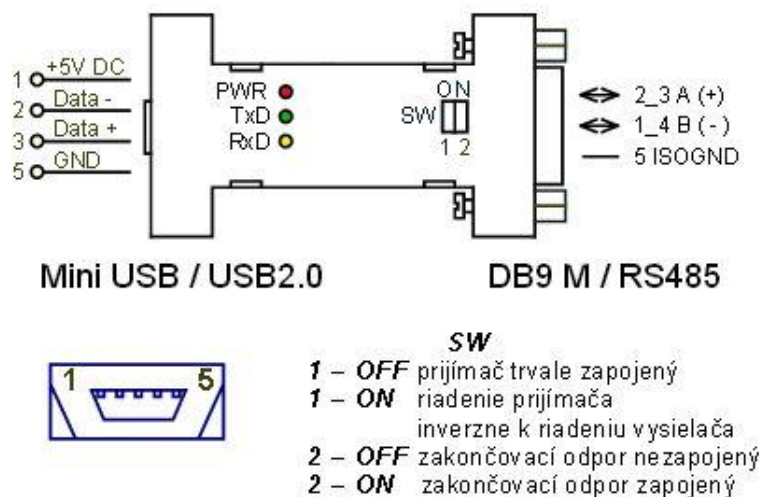
Obrázok 10 Telefón Samsung Galaxy Xcover 3 [16]

Displej telefónu je veľkosti 4,5 palca a typu WVGA s rozlíšením 800×480 pixlov. Telefón má štvorjadrový procesor 1,2 GHz a operačnú pamäť 1,5 GB RAM. Vnútoraná interná pamäť činí 8 GB. Podpora pamäťových kariet až 128 GB. Mobilné dátové pripojenie využíva technológiu LTE. Telefón je vybavený GPS prijímačom a komunikačným rozhraním bluetooth 4.0 a Wi-Fi. Kapacita batérie činí 2 200 mAh, čo je

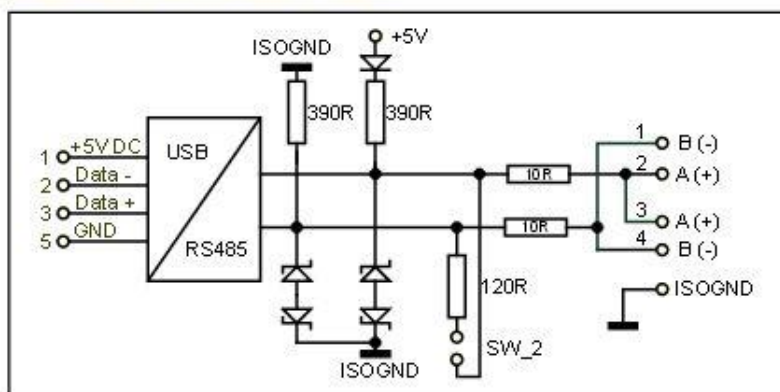
dostatok kapacity na 4 hodiny zasielania meracích dát aj bez pomocného napájania. Jedná sa o ďalšiu výhodu oproti bežnému priemyselnému PC, pri ktorom by na vykrývanie takéhoto výpadku musel byť použitý zdroj nepretržitého napájania UPS. Nevýhoda štandardného mobilného telefónu zasa spočíva v tom, že po vyčerpaní kapacity batérie je nutné telefón manuálne spustiť a zadať PIN kód. Rozmery telefónu sú $132,9 \times 70,1 \times 9,95$ mm a hmotnosť 154 gramov. Ako operačný systém slúži Android Kit Kat vo verzii 4.4.4. Tento operačný systém je výkonný aj na starších zariadeniach a pozitívne pre vývoj aplikácií je, že disponuje už vyšším API 20. Toto ponúka širokú škálu funkcií, ktoré môžu programátori využiť pri tvorbe aplikácií.

3.2 Použitý prevodník USB na RS 485

Keďže mobilné telefóny a v dnešnej dobe už vo väčšine prípadov aj notebooky nedisponujú priamym sériovým rozhraním, je nutné použiť zbernicu USB (Universal Serial Bus). Na smartfónoch je dokonca všeobecne používaný štandard mikro USB, a teda je nutné použiť ešte adaptér mikro USB na klasický USB konektor. Tento konektor je univerzálny a bežne používaný, preto ho nie je potrebné ďalej opisovať. Keďže mobilný telefón disponuje iba univerzálnou sériovou zbernicou a elektromer zasa komunikuje pomocou priemyselného štandardu RS 485, ktorý má iný princíp a iné napäťové úrovne ako USB, je nutné použiť prevodník z USB na RS 485. Existujú výrobcovia čipov ako napríklad Prolific (Prolific technology INC.) alebo FTDI (Future Technology Devices International Ltd.), ktorí vyrábajú mikročipy na tieto účely. Títo výrobcovia sú dosť známi a majú aj dobrú podporu ovládačov pre rôzne operačné systémy. Keďže návrhu a výrobe sériového prevodníka sa táto diplomová práca nevenuje, bol na tieto účely zakúpený už hotový prevodník od slovenskej firmy ComErgon, s.r.o. Spoločnosť ComErgon sa dlhodobo venuje vývoju a výrobe takýchto prevodníkov. Prevodníky USB na RS422 a RS485 sú vyrábané s galvanickým oddelením 1kV, nakoľko linky na strane RS422 a RS485 môžu byť vedené aj na veľké vzdialenosti a prípadné prepätie by mohlo poškodiť koncové zariadenia. Pre túto prácu bol použitý typ prevodníka MINI 5.6/485 s galvanickým oddelením. Tento prevodník využíva čip firmy FTDI. Je kompatibilný s USB 1.1 a USB2.0.



KONFIGURÁCIA LINKY RS485



Obrázok 11 Popis prevodníka USB/RS485 [11]

Priamu podporu ani ovládače k operačnému systému Android firma síce neponúka, no pri použití čipu FTDI je kompatibilný s univerzálnymi ovládačmi FTDI pre OS Android. Mód prenosu je RS485 - half duplex, maximálna prenosová rýchlosť 920kb/s, galvanické oddelenie o pevnosti 1 kV, prepäťová (soft) ochrana 300W 8/20us, voliteľný terminátor 120R, prenášané signály RxD, TxD, optická indikácia (PWR, RxD, TxD), ktorá je veľmi užitočná pri ladení aplikácie. Konektor RS485 - D SUB 9pin M, konektor USB - mini USB 5pin, zdroj (+5V DC z USB), spotreba maximálne 1,2W a rozmery 63 x 17 x 34mm. [11]

3.3 Použitý elektromer pre vývoj a testovanie aplikácie

Pre testovanie a vývoj aplikácie v laboratórnych podmienkach bol použitý elektromer najvyššej rady od nemeckej firmy EMH (EMH metering GmbH & Co. KG). Typ elektromera je **LZQJ S1F4** 3x58/100V-240/415 V-5A. Jedná sa o nepriamy elektromer, čo znamená, že je určený na meranie vysokých výkonov napríklad na veľkých

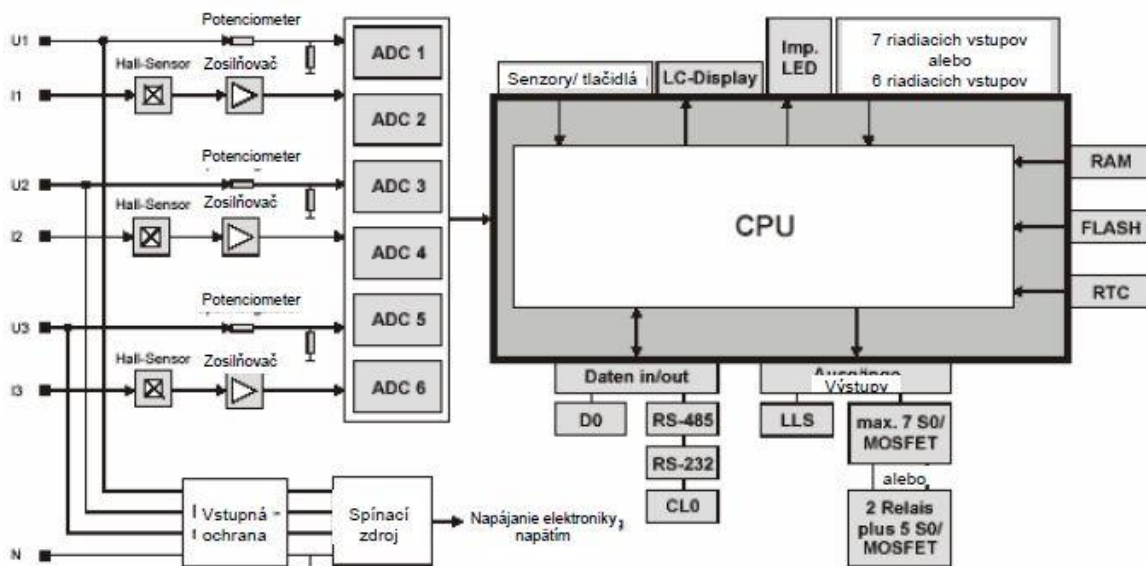
transformačných stanicích, kde tečú veľké prúdy a napätie dosahuje 22 000, 25 000 alebo aj 110 000 voltov. Pri takýchto prúdoch a napätiach je samozrejme nutné k elektromeru pripojiť špeciálne meracie transformátory, ktoré pomerovo s vysokou presnosťou (rádovo 0,5 % až 0,2% maximálnej chyby) premieňajú napríklad napätie z 25 000 V na 100 V. To je už pre elektromer prijateľné, rovnako premieňajú i prúd napríklad z 600 A na prijateľných 5 A. Aplikácia v tejto diplomovej práci sa testuje aj na hnacom dráhovom vozidle Škoda. Hnacie dráhové vozidlá sú napájané z trakčnej sústavy, ktorá dosahuje napätie 25 000 V striedavých a odber lokomotívy činí pri tomto napätí až 200 ampérov. Preto je použitý práve tento typ nepriameho elektromera. Prístroj je vyrábaný ako moderný elektronický kombinovaný (takzvaný 4-kvadrantný) elektromer na meranie činnnej a jalovej energie v režimoch odber aj dodávka. Disponuje veľkým množstvom štandardných a prídavných funkcií. Primárne je určený na meranie a vyhodnocovanie spotreby a dodávky elektrickej energie podľa platnej legislatívy, dokáže však oveľa viac. Prístroj má integrované dve komunikačné rozhrania. Optické servisné komunikačné rozhranie, na ktoré sa servisný technik pripája pomocou špeciálneho prevodníka USB/Optická hlava a externé, respektíve pracovné komunikačné rozhranie RS485, pomocou ktorého odčítava údaje priemyselný počítač alebo iná komunikačná jednotka (Master). V našom prípade mobilný telefón Samsung Galaxi Xcover3. Tým, že je elektromer 4-kvadrantný, nahradzuje meracie sady, ktoré boli tvorené doteraz klasicky zo 4 Ferrarisových (mechanických) elektromerov, po jednom činnom elektromere pre odber a dodávku plus po jednom jalovom elektromere pre odber a dodávku. Pokiaľ ide o systém OBIS-ukazovateľov, je 4-kvadrantný elektromer schopný zobrazit' namerané hodnoty podľa kódov 1.8.0, 2.8.0, 3.8.0, 4.8.0.

Merací systém	<ul style="list-style-type: none"> Napätie Elektrický prúd Dalšie napätia a prúdy Frekvencia Meracia presnosť (štandard) Meracia presnosť (opcia) Konštanta elektromera (LED) 	4L: 3x58/100V ... 3x240/415V 3L: 3x100V ... 3x415V 1A, 1(2)A, 5II1A, 5A na požiadanie 50 Hz, 60 Hz ($\pm 2\%$) činná energia Cl. 0,2 (IEC 62053-22) jalová energia 0,5% činná energia Cl. 0.5 (IEC 62053-22) jalová energia 1% parametrizovateľná
Tarifný systém	<ul style="list-style-type: none"> Zaznamenanie energie Zaznamenanie maxima Meracia perióda Pamäť profilu zaťaženia (konfigurovateľná) Tarifná forma Uchovanie údajov 	32 registrov + 8 bez tarify s max. 15 údajmi 32 registrov s max. 15 údajmi 1, 2, 3, 5, 10, 15, 30, 60 minút (konfigurovateľná) 1... 32 kanálov, hĺbka pamäti: pozri kapitolu 4.6.2.6 Profil záťaže konfigurovateľná špecificky pre zákazníka > 10 rokov
Displej	<ul style="list-style-type: none"> LC-displej 	Rozmery: 84 x 24 mm Rozsah hodnôt: výška 8 mm rozsah OBIS-Code: výška 6 mm alternatívne: 4-riadkový displej po 20 znakov
Tarifné spínacie hodiny, reálne hodiny	<ul style="list-style-type: none"> Nastaviteľné prostredníctvom Presnosť chodu Rezerva chodu so SuperCap Rezerva chodu s batériou (lítium) Nastaviteľné prostredníctvom Protokoly Frekvencia, funkčné napätie Výstupy 	D0- alebo elektrické rozhranie do ± 5 ppm 150 hod > 20 rokov D0- alebo elektrické rozhranie pozri 4,8. Kruhový riadiaci prijímač (RSE) nastaviteľný 6 kanálov
Rozhranie	<ul style="list-style-type: none"> Výstup údajov, konfigurácia 	optické rozhranie D0 podľa IEC 62056-21, elektrické rozhranie RS232, RS485 alebo CL0 (20mA-dvojdrôt)
Vstupy	<ul style="list-style-type: none"> 7 ríadiacich vstupov 	Systémové napätie, bezpotenciálové (opcionálne jedno prevedenie ako S0- vstup, beznapätový)
Výstupy	<ul style="list-style-type: none"> Pre výstup rôznych spínacích stavov, napr.: energetické impulzy, meracia perióda, tarifné stavy, špeciálne zákaznické kontakty atď. Svetlovodné rozhranie pre 	max. 7 x S0 alebo MOSFET, z toho alternatívne max. 2 ako relé S0 max. 27V DC, 27mA MOSFET max. 250V AC/DC, 100mA relé max. 250V AC/DC, 100mA pripoj boxu oddeľovacieho relé svetlovodu LTR
Spínací zdroj	<ul style="list-style-type: none"> Premostenie výpadku siete 	> 500ms
Sieťový diel s externým pomocným napätím	<ul style="list-style-type: none"> Sieťový diel s veľkým rozsahom 	pozri 4.4.1.1. Napájanie pomocným napätím
Vlastná spotreba, napätový obvod	<ul style="list-style-type: none"> s pomocným napätím bez pomocného napätia 	0,02VA ... 0,1VA na fázu (v závislosti od meracieho napätia) max. 1,7VA ... 3,5VA na fázu (v závislosti od meracieho napätia)
Vlastná spotreba, prúdový obvod		< 0,2VA na fázu
Izolačná pevnosť	<ul style="list-style-type: none"> Striedavé napätie Rázové napätie 	4 kV, 50Hz, 1min. 8kV, impulz 1,2/50 μ s
Mechanika	<ul style="list-style-type: none"> Rozmery telesa Hmotnosť Trieda ochrany Materiál telesa 	178 x 327 x 60 (Š x V x H) mm 1,6 kg Ochranné izolované, trieda 2 polykarbonát
Rozsah teploty	<ul style="list-style-type: none"> Prevádzkový rozsah / hraničný prevádzkový rozsah Uloženie a preprava (max.) 	-25°C ... +55°C / -40°C ... +70°C -40°C ... +70°C
Vlhkosť vzduchu	<ul style="list-style-type: none"> Prevádzka 	90% pri 40°C, bez kondenzácie

Obrázok 12 Technické parametre elektromera

Elektromer je trojfázový, teda disponuje tromi vstupmi pre napätia, tromi vstupmi pre prúdy, vstupom pre nulovací vodič, spomínaným výstupom pre komunikačné rozhranie RS 485, ktoré obsahuje výstup pre vodič A, výstup pre vodič B, uzemnenie GND a komunikačné rozhranie RS 232. Elektromer tiež disponuje špeciálnymi, tzv. impulznými výstupmi, ktoré slúžia na indikáciu prenosu výkonu v jednotlivých výkonových kvadrantoch. Princíp funkčnosti spočíva v priamej úmere meraného výkonu s frekvenciou

impulzov. Náš model elektromeru je tiež vybavený svorkami tzv. „pomocného napájania“, ktoré sa pri testovaní používalo ako hlavný zdroj prívodu napájacieho napätia. Bežne nie sú elektromery vybavené týmito svorkami a napájanie vlastnej spotreby sa zabezpečuje z meranej fázy. Elektromer tiež disponuje prijímačom HDO (hromadné diaľkové ovládanie) na prepínanie rôznych fakturačných taríf. Ako napríklad denná a nočná tarifa. Tarifné spínacie hodiny sú integrované v elektromere. Sú to hodiny reálneho času (Real Time Clock), riadené systémom quarz, zálohované z batérie alebo kondenzátora, ktoré poskytujú časovú informáciu (dátum, deň v týždni, čas) v sekundovej kadencii. Spínacie doby je možné konfigurovať špecificky pre zákazníka prostredníctvom sezónnych tabuliek, denných prepnutí alebo definícií sviatkov a voľných dní v týždni. Je nutné, aby boli vnútorné hodiny reálneho času vždy presne nastavené, inak to môže viesť k nesprávnej fakturácii za odobranú elektrinu, napríklad posunutím meracieho intervalu v profiloch elektromera či posunutím času zopnutia inej tarify. Preto je nutné čas elektromeru pravidelne zosynchronizovať, čo rieši aj aplikácia ELMReader v tejto diplomovej práci. Technik má možnosť odčítať čas a dátum reálnych hodín elektromera a v prípade, že nie je správny, môže čas a dátum nastaviť.



Obrázok 13 Bloková schéma zapojenia elektromera LZQJ

Elektromer disponuje 4 tabuľkami, ktoré sa nazývajú aj registre elektromeru. Jedná sa o textové tabuľky, ktoré nesú v jednotlivých riadkoch rôzne informácie o elektromere, aktuálne odoberanom a dodávanom výkone, kvalite meranej siete, prúde, napätí a iných dôležitých informáciách. Tieto tabuľky sú používateľsky voliteľné v parametrizácii

elektromera, no v energetickej praxi je zaužívané ich dohodnuté nastavenie. Tabuľka T1 je hlavná tabuľka, nazývaná býva aj register elektromeru.

/PMH5\@011ZQJC0014F	
C.1.0(02208889)	výrobné číslo
0.0.1(00000000)	identifikačné číslo 1
0.0.2(00000000)	identifikačné číslo 2
0.0.3(00000000)	identifikačné číslo 3
0.1.0(11)	počet kumulácií
0.1.2&11(160503213900)	Časová pečiatka nulovania 11.Predhodnota
0.1.2*10(160501074641)	Časová pečiatka nulovania 10.Predhodnota
0.1.2&09(160403145008)	Časová pečiatka nulovania 09.Predhodnota
0.9.1(150927)	denný čas
0.9.2(160509)	dátum
1.2.0(000.013*kW)	pozitívne činný výkon kumulatívny register tarifa 0
1.2.2(000.013*kW)	pozitívne činný výkon kumulatívny register tarifa 2
1.2.3(000.000*kW)	pozitívne činný výkon kumulatívny register tarifa 3
1.6.0(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 0
1.6.0&11(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 0 11.Predhodnota
1.6.0*10(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 0 10.Predhodnota
1.6.0&09(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 0 09.Predhodnota
1.6.2(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 2
1.6.2&11(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 2 11.Predhodnota
1.6.2*10(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 2 10.Predhodnota
1.6.2&09(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 2 09.Predhodnota
1.6.3(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 3
1.6.3&11(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 3 11.Predhodnota
1.6.3*10(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 3 10.Predhodnota
1.6.3&09(0.000*kW)(000000000000)	pozitívne činný výkon maximum tarifa 3 09.Predhodnota
1.8.0(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 0
1.8.0&11(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 0 11.Predhodnota
1.8.0*10(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 0 10.Predhodnota
1.8.0&09(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 0 09.Predhodnota
1.8.2(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 2
1.8.2&11(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 2 11.Predhodnota
1.8.2*10(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 2 10.Predhodnota
1.8.2&09(00000.058*kWh)	pozitívne činný výkon Register energie tarifa 2 09.Predhodnota
1.8.3(00000.000*kWh)	pozitívne činný výkon Register energie tarifa 3
1.8.3&11(00000.000*kWh)	pozitívne činný výkon Register energie tarifa 3 11.Predhodnota
1.8.3*10(00000.000*kWh)	pozitívne činný výkon Register energie tarifa 3 10.Predhodnota
1.8.3&09(00000.000*kWh)	pozitívne činný výkon Register energie tarifa 3 09.Predhodnota
2.2.0(000.000*kW)	negatívne činný výkon kumulatívny register tarifa 0
2.2.2(000.000*kW)	negatívne činný výkon kumulatívny register tarifa 2
2.2.3(000.000*kW)	negatívne činný výkon kumulatívny register tarifa 3
2.6.0(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 0
2.6.0&11(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 0 11.Predhodnota
2.6.0*10(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 0 10.Predhodnota
2.6.0&09(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 0 09.Predhodnota
2.6.2(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 2
2.6.2&11(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 2 11.Predhodnota
2.6.2*10(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 2 10.Predhodnota
2.6.2&09(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 2 09.Predhodnota
2.6.3(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 3
2.6.3&11(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 3 11.Predhodnota
2.6.3*10(0.000*kW)(000000000000)	negatívne činný výkon maximum tarifa 3 10.Predhodnota

Obrázok 14 Popis registru elektromera (Časť 1.)

2.6.3&09(0.000*kW) (0000000000000)	negatívne	činný výkon maximum	tarifa 3	09.Predhodnota
2.8.0 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 0	
2.8.0&11 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 0	11.Predhodnota
2.8.0*10 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 0	10.Predhodnota
2.8.0&09 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 0	09.Predhodnota
2.8.2 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 2	
2.8.2&11 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 2	11.Predhodnota
2.8.2*10 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 2	10.Predhodnota
2.8.2&09 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 2	09.Predhodnota
2.8.3 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 3	
2.8.3&11 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 3	11.Predhodnota
2.8.3*10 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 3	10.Predhodnota
2.8.3&09 (00000.000*kWh)	negatívne	činný výkon Register	energie tarifa 3	09.Predhodnota
3.8.0 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 0	
3.8.0&11 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 0	11.Predhodnota
3.8.0*10 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 0	10.Predhodnota
3.8.0&09 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 0	09.Predhodnota
3.8.2 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 2	
3.8.2&11 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 2	11.Predhodnota
3.8.2*10 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 2	10.Predhodnota
3.8.2&09 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 2	09.Predhodnota
3.8.3 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 3	
3.8.3&11 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 3	11.Predhodnota
3.8.3*10 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 3	10.Predhodnota
3.8.3&09 (00000.000*kvarh)	pozitívne	Jalový výkon Register	energie tarifa 3	09.Predhodnota
4.8.0 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 0	
4.8.0&11 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 0	11.Predhodnota
4.8.0*10 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 0	10.Predhodnota
4.8.0&09 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 0	09.Predhodnota
4.8.2 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 2	
4.8.2&11 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 2	11.Predhodnota
4.8.2*10 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 2	10.Predhodnota
4.8.2&09 (00000.182*kvarh)	negatívne	Jalový výkon Register	energie tarifa 2	09.Predhodnota
4.8.3 (00000.000*kvarh)	negatívne	Jalový výkon Register	energie tarifa 3	
4.8.3&11 (00000.000*kvarh)	negatívne	Jalový výkon Register	energie tarifa 3	11.Predhodnota
4.8.3*10 (00000.000*kvarh)	negatívne	Jalový výkon Register	energie tarifa 3	10.Predhodnota
4.8.3&09 (00000.000*kvarh)	negatívne	Jalový výkon Register	energie tarifa 3	09.Predhodnota
F.F (00000000)		register chýb		

Obrázok 15 Popis registru elektromera (Časť 2.)

Register obsahujú dodávané elektromery všetkých výrobcov na európskom trhu a jeho prítomnosť v elektromeroch je dohodnutá ako štandard. Odčítavaniu tohto registra sa venuje aj táto diplomová práca a funkcia jeho odčítania je implementovaná v aplikácii ELMReader ako manuálna funkcia pre potreby technikov či v automatickom režime diaľkového odpočtu elektromera. Na základe registra elektromera a časovej pečiatky dokáže distribútor vyhodnotiť aktuálne nameranú elektrickú prácu a fakturovať ju bez potreby fyzického odčítania na mieste. Ďalšou dôležitou funkciou, ktorú podporuje aj aplikácia ELMReader, je odčítanie výkonového profilu LP1(Load profile 01). Jedná sa o záťažový profil, konkrétne v tomto prípade s 15-minútovým odpočtovým intervalom, ktorý zaznamenáva 15-minútový priemer elektrického výkonu v štyroch meraciach kvadrantoch. Ide konkrétne o činný výkon odoberaný, ktorý reprezentuje identifikačný OBIS kód (1.5.0), činný výkon dodávaný (2.5.0), jalový výkon odoberaný (3.5.0) a jalový výkon dodávaný (4.5.0). Profily sa používajú najmä na fakturáciu veľkých odberov, kde sa fakturuje odoberaný výkon v 15-minútovom intervale a kontroluje sa jeho prekročenie. V prípade, ak by zákazník prekročil dohodnutý priemerný odoberaný výkon v danom 15-

minútovom intervale, hrozí mu pokuta od distribučnej spoločnosti za prekročenie maximálnej rezervovanej kapacity. Inak povedané za nepovolené preťaženie elektrického vedenia. Hoci nie každý riadok má pridelenú časovú pečiatku, je softvérovo veľmi jednoduché ju vyrátať podľa počtu riadkov krát 15 minút od poslednej časovej pečiatky. Spracovávaníu profilových a registrových dát sa však už venuje softvérová centrála na servery distribučnej spoločnosti. Úlohou priemyselných počítačov, a teda aj aplikácie ELMReader, je iba požadované dáta odčítať a prípadne ich odoslať cez mobilné dáta na server distribútora.

```
P.01(160505001500)(00000000)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
P.01(160505004500)(00000080)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
P.01(160505113000)(00000040)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
P.01(160505114500)(00000000)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
P.01(160505143000)(00000080)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
P.01(160505220000)(00000040)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
P.01(160505221500)(00000000)(15)(4)(1.5)(kW)(2.5)(kW)(3.5)(kvar)(4.5)(kvar)
(0.000)(0.000)(0.000)(0.000)
(0.000)(0.000)(0.000)(0.000)
```

Obrázok 16 Ukážka profilového záznamu elektromera

Elektromer LZQJ ako elektromer najvyššej rady disponuje ešte ďalšou veľkou množinou zvláštnych a doplnkových funkcionalít, meraním rôznych parametrov kvality siete, integrovanou kontrolou zapojenia, konfiguráciami tarifných pásiem a mnohými ďalšími, ktorými sa však táto práca podrobnejšie nezaobera a je možné ich nájsť v manuáli výrobcu.



Obrázok 17 Elektromer EMH LZQJ XC

3.4 Použitý elektromer pri testovaní na HDV Škoda

Hoci princíp funkčnosti, parametre a častokrát aj funkcionalita rôznych výrobcov sú podobné, nie každý výrobca sa venuje aj výrobe špeciálnych elektromerov, ako sú napríklad elektromery určené pre železničnú prevádzku. Každé zariadenie použité v lokomotíve potrebuje špeciálne schválenie železničného úradu danej krajiny. Elektronické zariadenia v lokomotívach sú vystavované úplne iným a často omnoho náročnejším podmienkam ako zariadenia inštalované v budovách. Elektromery určené pre železničnú prevádzku musia znášať vysoké výkyvy prevádzkových teplôt, sú vystavené vysokej prašnosti, vlhkosti, otrasom, pôsobeniu chemických látok, ako sú napríklad uhlíkový prach, častice oleja a iné nečistoty. Taktiež v železničnej prevádzke sa môže lokomotíva pohybovať vo viacerých trakčných napäťových úrovniach, načo musí elektromer vedieť reagovať. Napríklad v SR aj ČR máme 2 trakčné systémy. Striedavý 25 000 V a 50 Hz a jednosmerný 3 000 V. Pokiaľ lokomotíva premáva aj v západnej Európe, môže už napríklad v Rakúsku či Nemecku prísť do styku s 16 000 V sústavou.

Najväčším problémom je, že fakturačné elektromery pre meranie jednosmerného výkonu sa takmer nevyrábajú. Železnice si preto dávajú vyrobiť takéto zariadenia na zákazku. Na Železničiach Slovenskej republiky a takisto aj v Českej republike SŽDC (Správa železniční a dopravní cesty) sa používa najčastejšie univerzálny železničný elektromer Švajčiarskeho výrobcu LEM EM4T. Jedná sa o elektromer, ktorý je od vyššie uvedeného typu funkcionálne jednoduchší, keďže sa jedná o zariadenie vyvinuté na konkrétny účel. Absentujú napríklad možnosti tarifných nastavení, prijímač HDO, kvalita siete a iné nadštandardné funkcionality. Elektromer je však veľmi robustnej a odolnej konštrukcie, je certifikovaný pre použitie na hnacích dráhových vozidlách, dokáže sám rozoznávať napäťovú úroveň železničnej trakcie a meria aj jednosmerný výkon. Elektromer je jednofázový a jeho vstupy sú zvlášť pre jednosmerné napätie a prúd a zvlášť pre striedavé napätie a prúd. Taktiež disponuje svorkami pre pomocné napájanie, avšak toto činí 24 až 48 V, čo je vlastne napäťová hladina bežne používaná v lokomotívach. Elektromer obsahuje štandardne servisné optické komunikačné rozhranie a externý sériový port RS422, ktorý je vlastne podmnožinou rozhrania RS485.



Obrázok 18 Železničný elektromer LEM

Jeho hlavný rozdiel spočíva v možnosti plne duplexného sériového prenosu, ktorý ale v aplikácii ELMReader nie je používaný. Teda vodiče R+, T+ a vodiče R- a T- boli spojené, čím sa vlastne vytvoril štandardný polo duplex RS485. Elektromer dokáže automaticky prijímať a posielat' dáta v oboch režimoch a riadi sa štandardne príkazmi a popismi hodnôt pomocou kódu EDIS. Komunikácia aj funkcionálna sa riadi podľa európskeho štandardu IEC 62 056, tak ako aj vyššie uvádzaný elektromer EMH na ktorom bola aplikácia testovaná. Tento štandard popisuje všetky veličiny v elektromeroch špeciálnym kódom. Kód EDIS je teda vlastne podmnožinou univerzálneho kódu OBIS slúžiaceho na popis dát pri meraní vody, plynu, elektriny a iných veličín. Podmnožina EDIS sa zaoberá čisto iba kódmi pre meranie elektrickej energie. Elektromer rovnako disponuje registrom, profilmi a používa rovnaké príkazy pre odčítanie aj nastavovanie parametrov ako napríklad nastavenie času. Keďže aplikácia ELMReader ktorej sa venuje táto diplomová práca sa riadi týmto štandardom jej použitie by malo byť univerzálne a teda aj funkčné na všetkých elektromeroch, ktoré podporujú štandard IEC. Register elektromeru je oproti predošlému komerčne nasadzovanému modelu EMH zúžený. Obsahuje iba základné informácie a disponuje iba jedným registrom. Profilové dáta sú rovnaké avšak v železničnej prevádzke dochádza ku častým zmenám v napájaní lokomotívy a preto je časový interval záznamu profilu skrátený z 15 minút na jednu minútu, čo značne zväčšuje aj nároky na prenos dát.

4 Aplikácia ELM-Reader

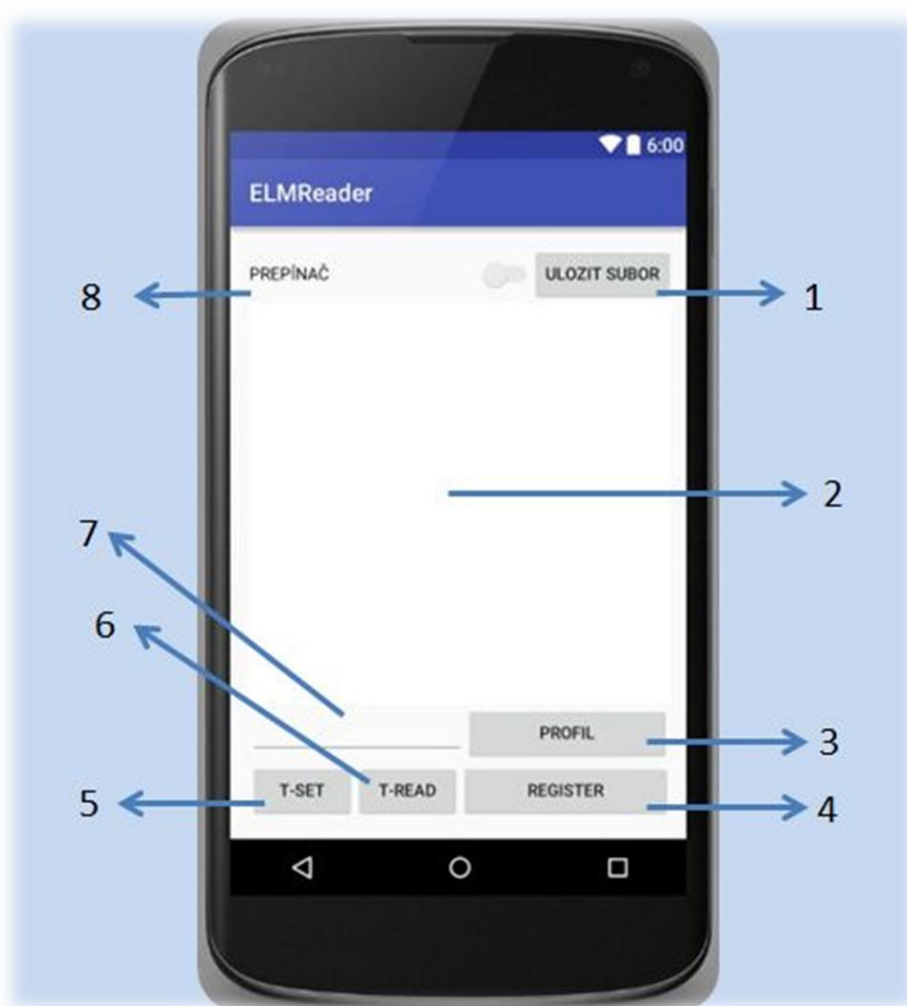
V súčasnosti je veľa výrobcov elektromerov a ich príslušenstva ako jednouúčelové priemyselné počítače a iné podporné zariadenia. Výrobcovia dodávajú k daným produktom väčšinou aj parametrizačný či používateľský softvér, ktorý spravidla funguje iba na operačnom systéme Windows. Je teda nutné, aby technický pracovník mal pri sebe štandardný notebook s operačným systémom Windows a mal nainštalovanú konkrétnu aplikáciu od konkrétneho výrobcu elektromera, ktorá funguje len na danom výrobku. Technici distribučných spoločností nenosia na servisné výjazdy vždy kompletnú servisnú sadu náradia a pomôcok ako napríklad notebook. Občas sa však aj pri servisnej činnosti iného typu vyskytne potreba kontroly dát či nastavení, spravidla v praxi hlavne kontrola a synchronizácia času elektromera či kontrola registra. V tomto prípade môže absencia notebooku či konkrétneho softvéru k danému elektromeru spôsobiť nemalé problémy. Jedna z myšlienok vývoja tejto aplikácie je, že v dnešnej dobe používa mobilný telefón takmer každý, ale žiadna zo spoločností vyrábajúcich elektromery a ich príslušenstvo sa nepustila do vývoja mobilného softvéru. Všeobecne sa vývoj aplikácií v smartfónoch ubera skôr v komerčných sférach a aplikácie pre industriálne využitie sa vyvíjajú iba veľmi zriedka. Aj podpora a ponuka knižníc pre priemyselný štandard RS 485 je pre operačný systém Android veľmi malá až žiadna, čo značne sťažovalo aj vývoj tejto aplikácie. Práve kvôli riešeniu týchto praktických problémov bola táto aplikácia vyvinutá. Smartfón s operačným systémom Android je veľmi rozšírený a s použitím malého ľahkého prevodníku je možné odčítať dáta či vykonať nastavenia v teréne priamo z telefónu. Keďže sa aplikácia riadi normou IEC a jej funkcionálna zahŕňa základné najčastejšie používané funkcie, stáva sa tým pádom univerzálnou a dokáže obsluhovať rôzne elektromery od rôznych výrobcov.

Druhý dôvod pre vývoj aplikácie ELMReader, ako sa spomína aj vyššie v teoretickej časti, je nerentabilnosť nasadenia drahých priemyselných počítačov na odpočet častokrát iba jedného elektromera. Ten je vo väčšine prípadov rádovo lacnejší než samotný priemyselný PC, a preto je jeho nasadenie odkladané alebo nerealizované. Vyskytujú sa aj prípady, kedy je potreba diaľkového odpočtu dát iba dočasná (napríklad stavby diaľnic a nasadenie dočasných transformačných staníc). Práve v týchto prípadoch by mohla byť využitá namiesto nasadenia kompletného finančne náročného hardvéru táto aplikácia. Mobilný telefón obsahuje hardvérovo aj softvérovo všetko potrebné pre lokálny sériový

zber a odosielanie dát po sieti internet. Preto bola implementovaná funkcionálna automatického odčítavania registrov a ich zasielanie na FTP server distribútora aj v tejto aplikácii. Veľmi užitočné použitie mobilného zasielania dát sa javí práve v železničnom sektore na lokomotívach, ktoré nemajú inú možnosť dátovej komunikácie.

4.1 Grafický návrh aplikácie

Pridaná hodnota tejto aplikácie spočíva v automatickom móde, kde hlavná funkcia aplikácie beží na pozadí bez prítomnosti používateľa. Cieľom aplikácie nebol zložitý vzhľad s množstvom doplnkových funkcií, ktoré je nutné si naštudovať a naučiť sa ich používať, ale naopak jednoduchosť a vybraná bola v praxi najčastejšie používaná funkcionálna. Tomuto zámeru bol prispôsobený aj grafický návrh aplikácie. Keďže sa jedná o aplikáciu pre industriálne využitie, je predpoklad, že s ňou budú pracovať odborníci na danú oblasť. Aj preto bola volená skôr jednoduchšia grafická architektúra.



Obrázok 19 Popis grafického návrhu aplikácie

Vysvetlivky k obrázku číslo 19. Popis grafického návrhu aplikácie:

1. Tlačidlo pre uloženie načítaných dát do pamäte telefónu
2. Zobrazovací rolovateľný prvok „TextView“ pre zobrazenie dát z elektromera
3. Tlačidlo pre odčítanie záťažového profilu
4. Tlačidlo pre odčítanie registra
5. Tlačidlo pre nastavenie času elektromera
6. Tlačidlo pre odčítanie času elektromera
7. Textové pole pre zadávanie počiatočného dátumu a času pre začiatok profilu
8. Prepínač medzi manuálnym a automatickým režimom programu

V prvej verzii boli naprogramované samostatné aktivity, napríklad pre zadanie rozsahu odpočtu profilov či formulár pre zadávanie najdôležitejších nastavení. Po konzultácii s odbornými pracovníkmi energetiky však boli tieto aktivity odstránené a nastavenia programu premiestnené do parametrizačného súboru. Ten je možné bežne editovať priamo v telefóne. Aplikácia sa teda skladá z jednej hlavnej aktivity, ktorá je uvedená na obrázku číslo 19. V ľavej hornej časti obrazovky je umiestnený prvok „switch“, teda prepínač pomocou ktorého sa aplikácia prepína z manuálneho módu do automatického. V manuálnom móde aplikácia slúži na lokálne odčítavanie údajov používateľom programu. V automatickom sú všetky tlačidlá zablokované a aplikácia na pozadí automaticky odčítava v prednastavených časových intervaloch register elektromera a zasiela ho na FTP server. Túto činnosť zároveň zobrazuje na prvku „TextView“, ktorý je v strede obrazovky a slúži ako zobrazovač odčítaných dát, popřípadе informácií o stave a činnosti aplikácie. V pravom hornom rohu vedľa prvku „switch“ je umiestnené tlačidlo „ULOZIT SUBOR“. Po stlačení tlačidla sa vykoná uloženie dát zobrazených na displeji do textového súboru s koncovkou .abl. Ten je následne uložený do koreňového adresára pamäte telefónu. V pravom dolnom rohu obrazovky sa nachádza tlačidlo „REGISTER“, po ktorého stlačení sa vypíše na obrazovku kompletný register elektromera. Vedľa neho v strednej dolnej časti je tlačidlo „T-READ“. Po jeho stlačení sa na displej vypíše dátum a čas elektromera aj s príslušnými OBIS kódmi. V ľavom spodnom rohu je tlačidlo „T-SET“, ktoré po stlačení nastaví dátum a čas elektromera pomocou dátumu a času telefónu. O priebehu nastavenia informuje aplikácia výpisom príkazov na displej. Hore nad týmto tlačidlom sa nachádza textové pole „EditText“, do ktorého sa zapisuje dátum a čas v tvare YYMMDDHHMM. Ten slúži ako počiatočný dátum a čas pre začiatok odčítavania

zátťažového profilu. Vedľa tohto prvku na pravej strane je tlačidlo „PROFIL“, ktoré spustí odčítavanie zátťažového profilu od zadaného dátumu a času až po aktuálny čas nastavený v telefóne. Ako základný podklad pre jednotlivé prvky v danej aktivite slúži „Relative Layout“. Na ňom je umiestnený „Linear Layout Vertical“, v ktorom sú uložené ďalšie 3 prvky „Linear Layout Horizontal“, v ktorých sú umiestnené koncové prvky ako tlačidlá a iné. Rozmiestňovanie prvkov prebieha pomocou XML kódu.

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="16dp"
    android:screenOrientation="portrait"
    tools:context="zsr.andrejpecimuth.dp.elmreader.MainActivity">

    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true"
        android:id="@+id/linearLayout">

        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T-Set"
            android:id="@+id/Tset"
            android:layout_alignParentBottom="true"
            android:layout_gravity="bottom" />
```

Obrázok 20 Ukážka XML kódu grafického návrhu

Každý prvok na obrazovke má svoj blok medzi značkami jazyka XML, kde sú opísané jeho vlastnosti, veľkosť, farba, štýl a iné informácie. Pri aplikácii bolo nežiaduce, aby sa natáčala obrazovka. Aj táto vlastnosť sa nastavuje pomocou XML kódu v danej aktivite. Pre zakázanie rotácie obrazovky je použitý príkaz `android:screenOrientation="portrait"`, čo značí, že obrazovka zostáva vždy v režime na stojato. Pri rotácii obrazovky sa vytvárali udalosti, ktoré rušili odpočty elektromera a dochádzalo k strate zobrazených dát na prvku „TextView“. Hornú lištu aplikácie popisuje IDE automaticky. Jedná sa o zobrazený názov projektu. V tomto prípade „ELMReader“.

Pri zadávaní dátumu a času do prvku číslo 7 „EditText“ sa automaticky otvorí číselná klávesnica s potvrdzovacím tlačidlom a tlačidlom mazania.

4.2 Nastavenie AndroidManifest

Každý balíček aplikácie Android musí mať vo svojom koreňovom adresári zložku AndroidManifest.xml (iné pomenovanie tejto zložky nie je prípustné). Android-Manifest poskytuje nevyhnutné informácie o aplikácii systému Android a informácie, ktoré systém potrebuje predtým, ako môže spustiť nejaký kód tejto aplikácie. Okrem toho popisuje komponenty (activities, services, broadcast receivers, content providers a iné), z ktorých aplikácia pozostáva. Ak sa popisy všetkých komponentov nenachádzajú v AndroidManifeste, aplikáciu nebude možné na zariadení spustiť. Preto ak chceme vkladať do APK súboru vlastné komponenty a nie len obohacovať existujúce o nejaký kód, musíme ich uviesť v AndroidManifeste.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="zsr.andrejpecimuth.dp.elmreader">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-feature android:name="android.hardware.usb.host"
        android:required="true"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="ELMReader"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <service
            android:name="zsr.andrejpecimuth.dp.elmreader.UsbService"
            android:enabled="true">
        </service>
    </application>
</manifest>
```

Obrázok 21 Nastavenia v súbore AndroidManifest.xml

Súbor AndroidManifest.xml vygeneruje IDE Android Studio automaticky pri vytvorení projektu. Do tohto súboru bolo nutné pridať napríklad oprávnenia, ktoré bude mať aplikácia po inštalácii. V našom prípade sa jednalo o prístup do externej pamäte telefónu, čo reprezentuje príkaz "android.permission.WRITE_EXTERNAL_STORAGE" pre zápis do pamäte a "android.permission.READ_EXTERNAL_STORAGE" pre čítanie z pamäte. Keďže aplikácia posielala dáta na FTP server, bolo nutné nastaviť aj povolenie prístupu na internet príkazom "android.permission.INTERNET" a v neposlednom rade prístup k USB zbernici príkazom "android.hardware.usb.host". Pri použití tohto príkazu sa pri pripojenom USB zariadení operačný systém pýta, či má použiť pripojené USB v tejto aplikácii. Ďalej je nastavená v tagu „Service“ Trieda USBservice, v ktorej sú naprogramované obslužné algoritmy pre chod USB. Ostatné nastavenia sú už bežné nastavenia aplikácie, ako napríklad meno či téma. Časť týchto nastavení je vygenerovaná automaticky po vytvorení projektu.

4.3 Ovládanie USB

Aplikácia ELMReader zbiera dáta z elektromera cez sériový port, konkrétne v našom prípade cez mikro USB port telefónu Samsung. Telefón beží v móde „USB Host“, čo znamená, že je v úlohe „Master“ a zabezpečuje aj napájanie zbernice a riadenie komunikácie, výpočet a adresovanie všetkých pripojených USB zariadení. Režim USB host módu je podporovaný v OS Android od verzie 3.1 a vyššie, teda aj v našom prípade. Pre vývojárov ovládačov má Android už predpripravený balíček tried s funkciami na prácu s USB portom, ktorý sa volá „android.hardware.usb“. Jeho triedy sú označené v tabuľke č. 1. Vo väčšine prípadov je ale nutné použiť pre plnohodnotnú komunikáciu všetky tieto triedy.

Tabuľka 1 Popis tried android.hardware.usb

UsbManager	Umožňuje vytvoriť zoznam a komunikovať s pripojenými zariadeniami USB.
UsbDevice	Predstavuje pripojené USB zariadenie a obsahuje metódy pre prístup k jeho identifikačným informáciám ako rozhranie a koncové body.
UsbInterface	Predstavuje rozhranie USB zariadenia, ktoré definuje sadu funkcií pre dané zariadenie. Zariadenie môže mať jedno alebo

	viac rozhraní, na ktorých dokáže komunikovať.
UsbEndpoint	Predstavuje koncový bod rozhrania, čo tvorí vlastne komunikačný kanál pre toto rozhranie. Rozhranie môže mať jeden alebo viac koncových bodov a zvyčajne má vstupné a výstupné koncové body pre obojsmernú komunikáciu s prístrojom.
UsbDeviceConnection	Predstavuje pripojenie k zariadeniu, ktoré prenáša dáta na koncových bodoch. Táto trieda umožňuje posielat' dáta tam a späť synchrónne alebo asynchrónne.
UsbRequest	Predstavuje asynchrónnu požiadavku na komunikáciu so zariadením prostredníctvom UsbDeviceConnection .
UsbConstants	Definuje USB konštanty, ktoré spĺňajú definíciu linux / USB / ch9.h linuxového jadra.

Pre používanie USB host módu je nutné deklarovať do súboru AndroidManifest.xml používanie "android.hardware.usb.host". Je možné aj filtrovať pripojené USB zariadenia pomocou filtru <usb-device> ktorý po pripojení zariadenie rozpozná na základe atribútov, ako sú vendor-id, product-id, Class, subclass a protocol. Je to výhodné pri rozpoznávaní pripojených zariadení a ich kategórií ako veľkokapacitné pamäťové zariadenie (napr. USB kľúč), digitálny fotoaparát či iné. Táto možnosť filtrácie pripojených zariadení však nebola v našej aplikácii implementovaná z dôvodu možného použitia rôznych USB/RS485 prevodníkov. Aplikácia ELMReader si po pripojení USB zariadenia vyžiada od USB manažéra OS Android zoznam všetkých pripojených USB zariadení „device-list“ s výnimkou koreňového rozbočovača USB. Ovládanie USB rozbočovača nie je pre našu aplikáciu potrebné. Je predpoklad, že pri odčítavaní elektromera nebudú pripojené na USB port viaceré zariadenia pomocou rozbočovača. Všeobecne je používanie tohto prvku v mobilných telefónoch takmer nepoužívané. Po pripojení USB prevodníka a spustení aplikácie je používateľ vyzvaný dialógovým oknom na potvrdenie, či má systém povoliť aplikácii prístup k tomuto USB zariadeniu. Používateľ musí túto akciu potvrdiť voľbou „OK“, inak aplikácia nebude fungovať správne. Naša aplikácia spracováva oznámenia a povolenia operačného systému v bloku uvedenom na obrázku č. 22. Funkcia „BroadcastReciver“, ktorá sleduje správy od systému pri každej

udalosti, odchyťáva akcie, ktoré sa s USB portom dejú. Ak bolo USB zariadenie operačným systémom nájdené a povolené, nastaví sa premenná boolean na hodnotu TRUE a v podmienke sa pokúsi otvoriť zariadenie ako štandardný sériový port. Pokiaľ používateľ nepotvrdil povolenie, port sa neotvorí. Pokiaľ bolo USB zariadenie odpojené, odošle sa informácia do hlavnej triedy MainActivity a tam ho spracuje príslušný „BroadcastReceiver“, ktorý zobrazí na displeji oznámenie typu „Toast“ o tom, že USB zariadenie bolo odpojené.

```
private final BroadcastReceiver usbReceiver = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context arg0, Intent arg1) {  
        if (arg1.getAction().equals(ACTION_USB_PERMISSION)) {  
            boolean granted = arg1.getExtras().getBoolean(UsbManager.EXTRA_PERMISSION_GRANTED);  
            if (granted) // Užívateľ prijal pripojenie USB. Pokusi otvoriť zariadenie ako sériový port.  
            {  
                Intent intent = new Intent(ACTION_USB_PERMISSION_GRANTED);  
                arg0.sendBroadcast(intent);  
                connection = usbManager.openDevice(device);  
                serialPortConnected = true;  
                new ConnectionThread().run();  
            } else // Používateľ neprijal USB. Informácia sa odošle do activity_main.  
            {  
                Intent intent = new Intent(ACTION_USB_PERMISSION_NOT_GRANTED);  
                arg0.sendBroadcast(intent);  
            }  
        } else if (arg1.getAction().equals(ACTION_USB_ATTACHED)) {  
            if (!serialPortConnected)  
                findSerialPortDevice(); // Bolo pripojené zariadenie. Pokusi sa ho otvoriť ako štandardný sériový port  
        } else if (arg1.getAction().equals(ACTION_USB_DETACHED)) {  
            // USB bolo odpojené, posle sa do Main Activity.  
            Intent intent = new Intent(ACTION_USB_DISCONNECTED);  
            arg0.sendBroadcast(intent);  
            serialPortConnected = false;  
            serialPort.close();  
        }  
    }  
};
```

Obrázok 22 Spracovávanie oznámení OS Android

Rovnako sa zobrazujú iné stavové informácie, ako „USB Nepripojené“, „USB nie je podporované“ a iné zobrazené na obrázku číslo 23. Používateľ je o každej zmene stavu informovaný bublinou „Toast“, ktorá sa zobrazí po dobu niekoľko sekúnd na displeji telefónu.

```
// Oznámenia o stave pripojenia USB
private final BroadcastReceiver mUsbReceiver = (context, intent) -> {
    switch (intent.getAction()) {
        case UsbService.ACTION_USB_PERMISSION_GRANTED:
            Toast.makeText(context, "USB Pripravene", Toast.LENGTH_SHORT).show();
            break;
        case UsbService.ACTION_USB_PERMISSION_NOT_GRANTED:
            Toast.makeText(context, "USB Nie je garantovane!", Toast.LENGTH_SHORT).show();
            break;
        case UsbService.ACTION_NO_USB:
            Toast.makeText(context, "USB Nepripojene!", Toast.LENGTH_SHORT).show();
            break;
        case UsbService.ACTION_USB_DISCONNECTED:
            Toast.makeText(context, "USB Bolo odpojené!", Toast.LENGTH_SHORT).show();
            break;
        case UsbService.ACTION_USB_NOT_SUPPORTED:
            Toast.makeText(context, "USB Zariadenie nie je podporovane!", Toast.LENGTH_SHORT).show();
            break;
    }
};
```

Obrázok 23 Spracovanie oznámení o stave USB

O nájdenie zariadenia a vytvorenie sériového pripojenia sa v našej aplikácii stará funkcia „findSerialPortDevice()“. Po jej zavolaní sa vyčíta z operačného systému zoznam pripojených zariadení pomocou funkcie „usbManager.getDeviceList()“ a ak nie je tento zoznam prázdny, načítajú sa údaje o vendor id, product id a otvorí sa štandardný sériový port pre komunikáciu so zariadením. Pokiaľ zariadenie nie je podporované alebo neexistuje, odošle sa správa hlavnej aktivite a tá následne vypíše túto informáciu na display. Samotná komunikácia po sériovom porte je nastavovaná a spúšťaná v samostatnom vlákne. Je to nutné kvôli tomu, aby aplikácia pri čítaní a prijímaní dát zo sériovej linky nezamrzala.

```

private class ConnectionThread extends Thread {
    @Override
    public void run() {
        serialPort = UsbSerialDevice.createUsbSerialDevice(device, connection);
        // Nacitanie BaudRate zo suboru
        File cardR = Environment.getExternalStorageDirectory();
        File dirR = new File(cardR.getAbsolutePath());
        File fileR = new File(dirR, "ELMSetings.txt");
        try {
            BufferedReader br = new BufferedReader(new FileReader(fileR));
            String line;
            if ((line = br.readLine()) != null) {BAUD_RATE = Integer.parseInt(line);}
            br.close();
        }
        catch (IOException e) {Toast.makeText(UsbService.this, "Nepodarilo sa nacist BAUD RATE!",
        // Nacitanie BaudRate zo suboru

        if (serialPort != null) {
            if (serialPort.open()) {
                serialPort.setBaudRate(BAUD_RATE);
                serialPort.setDataBits(UsbSerialInterface.DATA_BITS_7);
                serialPort.setStopBits(UsbSerialInterface.STOP_BITS_1);
                serialPort.setParity(UsbSerialInterface.PARITY_EVEN);
                serialPort.setFlowControl(UsbSerialInterface.FLOW_CONTROL_OFF);
                serialPort.read(mCallback);

                // Ak všetko ide ako ma posle sa Intent do MainActivity
                Intent intent = new Intent(ACTION_USB_READY);
                context.sendBroadcast(intent);
            } else {
                // Seriový port nejde otvoriť. Sposobene možno chybou I/O alebo ak bol vybraný CI
                // Posle Intent do Main Activity
                if (serialPort instanceof CDCSerialDevice) {
                    Intent intent = new Intent(ACTION_CDC_DRIVER_NOT_WORKING);
                    context.sendBroadcast(intent);
                } else {
                    Intent intent = new Intent(ACTION_USB_DEVICE_NOT_WORKING);
                    context.sendBroadcast(intent);
                }
            }
        }
    }
}

```

Obrázok 24 Inicializácia sériového portu

Ako je vidieť na obrázku 24, po otvorení sériového portu sa otvorí súbor „ELMSetings.txt“, z ktorého sa načíta nastavenie prenosovej rýchlosti. Defaultne je nastavené na rýchlosť 300 Baud Rate, čo je štandardná inicializačná rýchlosť pre optické komunikačné hlavy používané servisnými technikmi. Pri použití linky RS 485 je však odporúčaná rýchlosť 9600. Tá sa načíta z prvého riadka parametrizačného súboru. Zvyšné nastavenia sériovej komunikácie zostávajú nezmenné, pretože aplikácia je určená výlučne na odčítanie elektromerov. Tie sú nastavené podľa normy na 7-bitovú dĺžku, jeden stop bit, parity event a hardvérová kontrola toku dát sa pri zbernici RS 485 ani optických hlavách nepoužíva. Po úspešnej inicializácii je sériový port pripravený prijímať a odosielať bity. Na odosielanie slúži funkcia `public void write(byte[] data) { if (serialPort != null) serialPort.write(data); }`, ktorá je volaná z hlavného vlákna pri stlačení nejakého tlačidla. Na odchytyvanie bajtového prúdu zo sériového portu slúži funkcia

„UsbSerialInterface.UsbReadCallback“, ktorá odchyťí bajtový prúd, prevedie ho na reťazec a odošle hlavnej aktivite. Tam je následne spracovávaný a odosielaný do premennej ABLfile a na zobrazovací prvok display, ktorý tvorí komponent „TextView“. Následne je už na používateľovi, či mu stačí dáta iba zobrazit' alebo ich chce aj uložiť do pamäte v textovom súbore. Dáta sa ukladajú do textového súboru zo špeciálnou príponou „.abl“. Jedná sa o typ súboru, ktorý je daný normou STN EN 62056-21.

```
private UsbSerialInterface.UsbReadCallback mCallback = new UsbSerialInterface.UsbReadCallback() {  
    @Override  
    public void onReceivedData(byte[] arg0) {  
        try {  
            String data = new String(arg0, "UTF-8");  
            if (mHandler != null)  
                mHandler.obtainMessage(MESSAGE_FROM_SERIAL_PORT, data).sendToTarget();  
        } catch (UnsupportedEncodingException e) {  
            e.printStackTrace();  
        } catch (IOException e) {  
        }  
    }  
}
```

Obrázok 25 Zber prichádzajúcich údajov zo sériového portu

4.3.1 Knižnica pre sériovú komunikáciu com.felhr.usbserial

Komunikácia s elektromerom prebieha po sériovej linke podľa priemyselného štandardu RS 485. Samotné hardvérové porty sériovej linky RS 485 na bežných PC alebo notebookoch neboli nikdy samozrejmosťou. Na väčšine starších zariadení býval bežne dostupný sériový port RS 232. Ten sa hardvérovo odlišuje od RS 485 hlavne napätovou hladinou a počtom vodičov. Zo softvérovej strany je však sériová komunikácia takmer rovnaká. S univerzálnou sériovou zbernicou USB je ale situácia o niečo zložitejšia. Kým štandard RS 232 ráta iba s dvoma komunikujúcimi jednotkami „master“ a „slave“, USB môže mať cez rozbočovače pripojených až 127 zariadení typu slave. Na premostenie linky USB s linkou RS 485 bol použitý vyššie spomínaný prevodník MINI 5.6. Ten však vystupuje pre USB zbernicu ako klasické USB zariadenie s rozhraním a koncovými bodmi. Pre toto zariadenie by bolo nutné naprogramovať kompletný ovládač pre OS Android. Aj kvôli tomuto problému a aj kvôli zjednodušeniu ovládania bola použitá jednoduchá knižnica zameraná na sériovú komunikáciu „com.felhr.usbserial“. Táto knižnica obsahuje súbor užitočných funkcií a tried, ktoré značne zjednodušili prácu s USB portom, a taktiež obsahuje okrem iných ovládačov, ako napríklad všeobecne známe Arduino či Prolific, aj ovládač pre čip FTDI. Na čipe FTDI je založený aj prevodník MINI 5.6. S knižnicou boli

použité triedy „UsbSerialDevice“ na inicializáciu a nastavenie sériovej komunikácie, „UsbSerialInterface“ na zápis a čítanie dát cez rozhranie a „CDCSerialDevice“ ako ovládač pre zariadenia. Knižnica je bezplatná s otvorenou licenciou a je zverejnená na portáli github.com.

4.4 FTP Pripojenie

Aplikácia ELMReader obsahuje dva módy. Automatický a manuálny. Pri automatickom móde aplikácia v prednastavených časových intervaloch odčítava z elektromera dátový register a ten sa po uložení do súboru automaticky po desiatich sekundách zasiela na zvolený FTP server. Pre túto funkcionality bola použitá knižnica Apache Commons Net TM. Táto Knižnica podporuje na strane klienta množstvo základných internetových protokolov. Účelom knižnice je poskytnúť základný prístup k protokolu, nie však abstrakcie na vyššej úrovni. Preto niektoré z návrhov porušujú objektovo orientované princípy. Hlavnou filozofiou knižnice je, aby boli prístupné globálne funkcie protokolu napríklad TFTP odoslať súbor alebo prijať súbor atď. Hoci je knižnica rozsiahla, v našej práci boli použité iba triedy poskytujúce FTP pripojenie, a to „org.apache.commons.net.ftp.FTP“ a „org.apache.commons.net.ftp.FTPClient“. V podstate bolo nutné naprogramovať FTP klienta s funkciou, ktorá bude odosielať uložené súbory v pamäti telefónu na zvolený FTP server. Pre túto úlohu sme vytvorili v našej aplikácii separátnu triedu „FtpAsyncTask“, ktorá obsahuje všetky potrebné metódy na vytvorenie FTP pripojenia. Trieda využíva ešte ďalšie dve pomocné triedy, a to „FtpAction“ pre zadanie údajov o prenášanom súbore a „FtpConection“ pre zadanie údajov o vytvorení FTP spojení. V tejto triede sú taktiež naprogramované oznámenia typu toast, napríklad o úspešnom či neúspešnom uložení súboru na FTP server a „ProgressDialog“ zobrazujúci informáciu o tom, že odosielanie súboru práve prebieha. Hlavnú úlohu zohráva funkcia „doInBackground“ volaná z hlavnej triedy „MainActivity“.


```
protected Boolean doInBackground(FtpAction... actions) {  
    FTPClient ftp = new FTPClient();  
    FtpAction a = actions[0];  
    try {  
        ftp.connect(a.connection.server, a.connection.port);  
        ftp.login(a.connection.username, a.connection.password);  
        File car = Environment.getExternalStorageDirectory();  
        File di = new File(car.getAbsolutePath());  
        File file1 = new File(di, a.remoteFilename);  
        boolean success = false;  
        FileInputStream inputStream = new FileInputStream(file1);  
        ftp.setFileType(FTP.LOCAL_FILE_TYPE);  
        ftp.enterLocalPassiveMode();  
        success = ftp.storeFile(a.remoteFilename, inputStream);  
        if (success == true) {file1.delete();}  
        // inputStream.close();  
        a.success = success;  
        ftp.disconnect();  
        if (success == true) { file1.delete();}  
        return success;  
    } catch (Exception x) {  
        this.exception = x;  
  
        a.success = false;  
  
        return false;  
    }  
}
```

Obrázok 26 Funkcia pre odosielanie súboru na FTP server

Po inicializácii pomocných tried sa nastaví príkazom `ftp.connect` URL či IP adresa požadovaného FTP servera a jeho komunikačný port, ktorý je defaultne prednastavený na 21. Príkaz `ftp.login` odošle serveru meno a heslo používateľa, nasleduje vytvorenie absolútnej adresy odosielaného súboru v pamäti telefónu, nastavenie typu odosielaného súboru, pasívny FTP mód prenosu, samotný príkaz na uloženie `ftp.storeFile` a nakoniec príkaz ukončenia prenosu `ftp.disconnect`. Nastavenie všetkých FTP parametrov prebieha v hlavnej triede v časovom intervale alarm manažéra.

```

if (pocitadlo==2){
    FtpConnection cc = new FtpConnection();
    cc.password = FTPheslo;
    cc.port = FTPport;
    cc.server = FTPserver;
    cc.username = FTPmeno;

    FtpAction a = new FtpAction();
    a.upload = true;
    a.connection = cc;
    a.remoteFilename = Autosubor;
    new FtpAsyncTask(MainActivity.this).execute(a);
    SimpleDateFormat dateFormatAutoWW = new SimpleDateFormat("yy.MM.dd-HH:mm:ss");
    display.append(" \nStart odosielania suboru na FTP "+dateFormatAutoWW.format(new Date())+"\n");
    am.set(AlarmManager.ELAPSED_REALTIME_WAKEUP, SystemClock.elapsedRealtime() + Interval, pi);
}

```

Obrázok 27 Nastavenie parametrov FTP prenosu v MainActivity

Po spustení aplikácie prebehne načítanie údajov o pripojení z parametrizačného súboru do premenných, pomocou ktorých sa nastaví atribúty FTP pripojenia, ako je vidieť na obrázku 27. Po nastavení parametrov sa spustí prenos príkazom „new FtpAsyncTask(MainActivity.this).execute(a)“ a odošle sa informácia o začiatku prenosu na displej. K tejto informácii je pripojená aj časová pečiatka získaná z hodín OS Android.

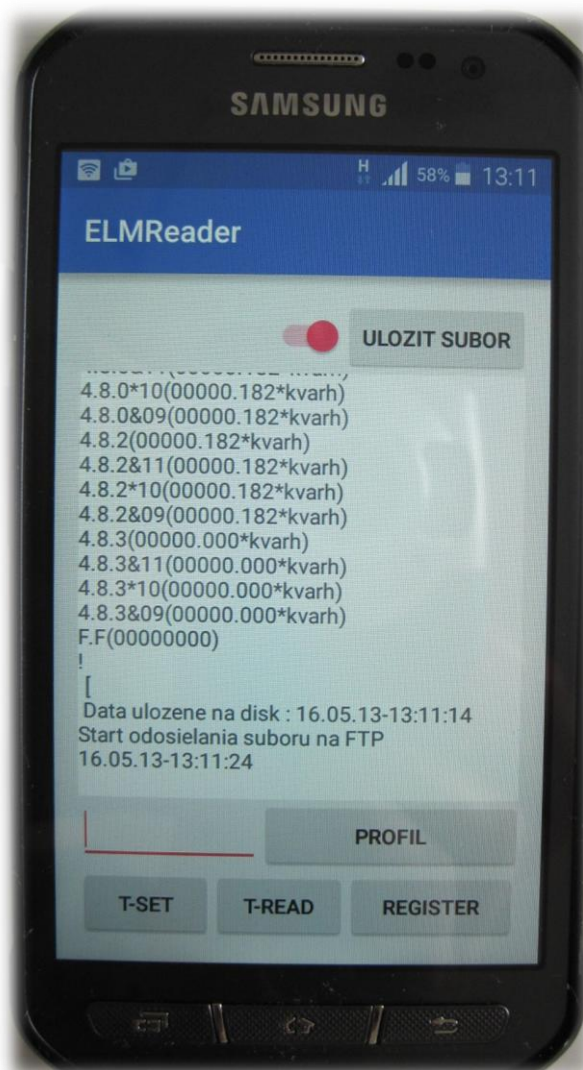
4.5 Popis funkcií aplikácie

Aplikácia funguje v dvoch základných režimoch, a to v manuálnom, v ktorom sú k dispozícii všetky manuálne funkcie, ktoré už boli popísané v grafickom návrhu, a v automatickom, v ktorom sú tlačidlá zablokované a aplikácia na pozadí odčítava v časových intervaloch register elektromeru a posiela ho na FTP server. O tom, v ktorom režime aplikácia je, informuje prvok „switch“, pomocou ktorého sa aj aplikácia prepína z jedného módu na druhý. „Listener“ stlačenia každého tlačidla je ošetrený podmienkou, že ak je switch prepnutý v polohe Auto, žiadna funkcia sa nevykoná, ale používateľ je informovaný bublinou typu toast o tom, že musí najprv vypnúť automatický režim. Podobne je ošetrený automatický režim v časovom cykle, ak sa prepne switch do režimu manuál, nič sa nevykoná a ďalší časový cyklus alarm manažéra už nebude nastavený.

4.5.1 Automatický režim aplikácie

V automatickom režime aplikácia využíva triedu „AlarmManager“, ktorá je už štandardne implementovaná v operačnom systéme Android. Úkony, ktoré musia bežať v časových intervaloch, sú zaradené do void funkcie alarm manažéra s názvom „setup“ a z listenerov jednotlivých tlačidiel sa volá už iba inicializácia alarm manažéra „am.set()“.

V tejto inicializácii je možné nastaviť rôznu dĺžku časového intervalu. Pri auto móde funkcia vyvolaná v čase volá zase samu seba v nekonečnom cykle, pokiaľ je splnená podmienka, že prepínač je zapnutý v polohe „Auto“. Pri prepnutí prepínača do automatického režimu sa odošle na sériový port príkaz na odčítanie registra elektromeru. Po odčítaní sa nastaví alarm manažér na 10 sekúnd a inkrementuje sa premenná „počítadlo“, pomocou ktorej sa určuje, ktorá funkcia sa má v danom časovom intervale vykonať. Ako ďalšiu voláme funkciu „Datsender“ nachádzajúcu sa v triede UsbService, ktorá registrované dáta uložené v premennej ABLFile zapíše do súboru .abl v koreňovom adresári pamäte telefónu. Aby bolo možné na strane servera súbory lepšie identifikovať, pridáva sa do súboru okrem názvu aj číslo elektromera a časová pečiatka získaná z operačného systému pomocou triedy „SimpleDateFormat“. Po uložení súboru do pamäte sa opäť nastaví čas na 10 sekúnd a inkrementuje sa počítadlo.



Obrázok 28 Automatický odpočet elektromera

Nasleduje inicializácia, nastavenie atribútov a spustenie FTP pripojenia, čo je aj zaznamenané na displeji telefónu spolu s časovou pečiatkou. V prípade úspešného odoslania súboru sa nastaví čas na používateľom predvolený časový interval odpočtu (defaultne 5 minút) a proces sa opakuje do nekonečna, pokiaľ je zapnutý prepínač v automatickom móde. V prípade neúspešného pripojenia súbor zostáva uložený v pamäti a proces pokračuje ďalej. Pri problémoch s internetovým pripojením telefónu sa namerané dáta nestratia a je možné ich získať manuálne. Pri veľkosti rádovo v jednotkách kilobajtov by trvalo roky kým by súbory zaplnili pamäť telefónu, takže daný scenár zatiaľ nebolo potrebné programátorsky ošetrovať.

4.5.2 Odčítanie registra elektromeru

Pri komunikácii s elektromerom sa riadi aplikácia podľa normy STN EN 62056-21, ktorá hovorí o výmene údajov na odčítanie elektromerov, sadzieb a zaťaženia. Register elektromera je základná tabuľka každého elektronického elektromera, a preto sa aj podľa danej normy posiela register elektromeru ako automatická odpoveď na ohlásenie.

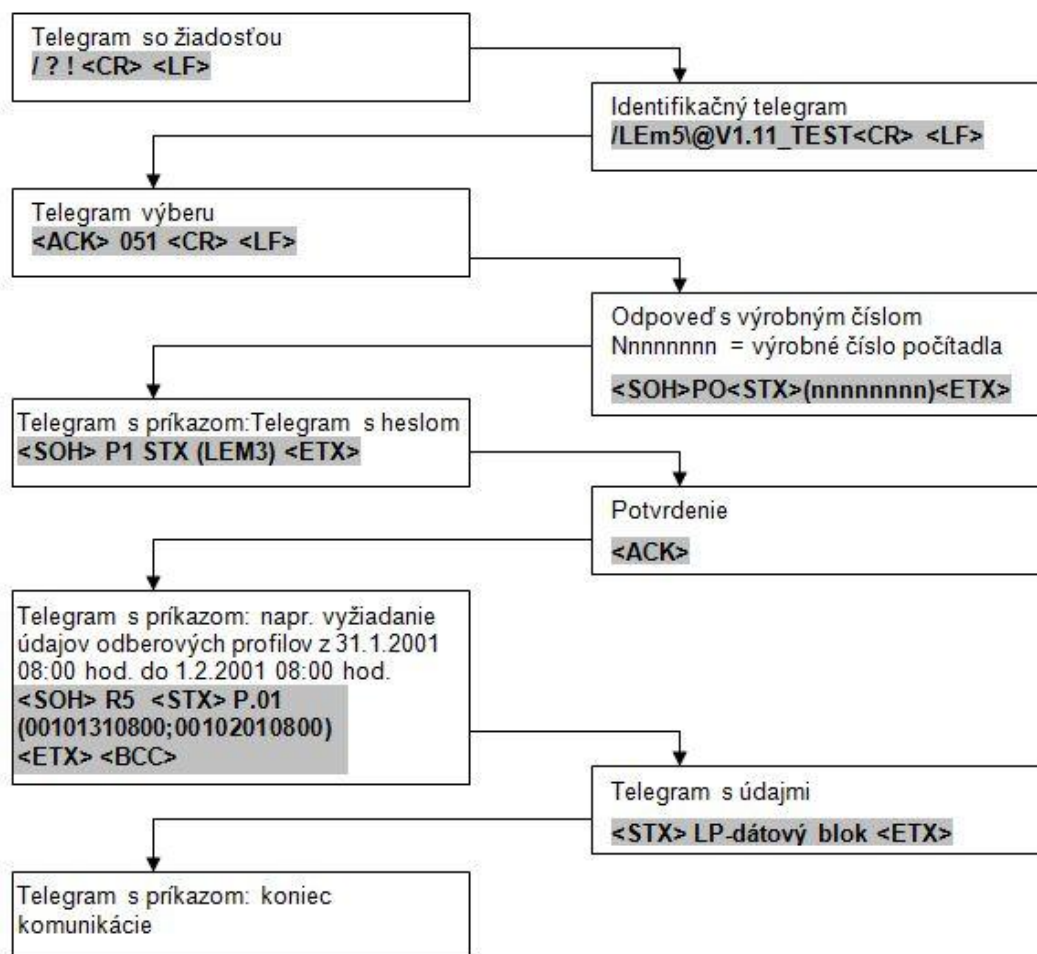
Tabuľka 2 Veta na oslovenie elektromera [7]

/	?	Adresa prístroja	!	CR	LF
---	---	------------------	---	----	----

Podľa danej tabuľky sa odošle na sériovú linku daná sekvencia znakov, značka CR+LF znamená posunutie kurzora na začiatok riadku a nový riadok, teda vlastne klávesu „Enter“. Po prijatí zašle elektromer odpoveď v tvare „/EMH5\@01LZQJC0014F“ a počká 1,5 až 5 sekúnd (podľa nastavenia elektromera). Následne zašle kompletný register. Dáta prichádzajúce z elektromera sa zobrazujú na displeji telefónu. V prípade potreby je možné dáta uložiť do súboru pomocou tlačidla uložiť súbor.

4.5.3 Odčítanie záťažového profilu elektromera

Odčítanie záťažového profilu je náročnejšie z pohľadu riadenia komunikácie na sériovej zbernici. Keďže knižnice RS 485 pre OS Android neexistujú a často používaná knižnica pre jazyk Java RXTX nie je primárne určená pre telefóny, bolo nutné ovládanie komunikácii po zbernici naprogramovať. Pri programovaní sa využila interná trieda alarm manažér a na riadenie časových ofsetov odpovedí elektromeru. Podľa normy je maximálny čas odpovede na požiadavku elektromeru 1500 milisekúnd. V našej aplikácii bol zvolený ofset 900 milisekúnd, ktorý bez problémov postačoval aj pri najnižšej rýchlosti 300 bd.



Obrázok 29 Komunikácia medzi telefónom (vľavo) a elektromerom (vpravo)

Diagram komunikácie na obrázku 29 zobrazuje oslovenie elektromera LEM používaného v lokomotívach. Pred stlačením tlačidla „PROFIL“ je nutné zadať do textového poľa dátum a čas začiatku odpočtu v tvare „yymmddhhmm“. Koncový dátum a čas odpočtu je vždy aktuálny čas telefónu a je generovaný automaticky. Rovnako ako je tomu pri odpočte registra začína sa zo základnou ohlasovacou vetou, ktorá je tentokrát bez čísla elektromera iba s otáznikom. Pokiaľ je na zbernici RS 485 pripojený iba jeden elektromer, jeho číslo sa zadávať nemusí. Nasleduje odpoveď elektromera. Po uplynutí 900 milisekúnd zasiela aplikácia telegram s kódom 051 výberu rýchlosti a protokolu. Číslo 5 znamená maximálnu povolenú rýchlosť 9600 bd. V prípade čísla 4 by sa jednalo o rýchlosť 4800 bd. Po odoslaní elektromer zašle správu, v ktorej požaduje heslo. Po 900 milisekundách mu zašle aplikácia telegram s heslom v zátvorke. V tomto prípade je heslo „LEM3“. Elektromer posiela potvrdzovací znak a aplikácia odpovedá znakovou vetou pre odčítanie záťažového profilu. Znak „R5“ hovorí o tom, že sa bude jednať o čítanie dát, znak kódu EDIS „P.01“ hovorí, že sa jedná o záťažový profil, a časové pečiatky v zátvorke

označujú časový rozsah požadovaných dát. Znak <SOH>, <STX>, <BCC> sú pomocné komunikačné znaky a určujú napríklad začiatok a koniec správy či potrebu pridania znaku kontrolného súčtu. Znak kontrolného súčtu sa pridáva na koniec správy vtedy, ak je správa premenná, ako napríklad v tomto prípade, keď pri každom odpočte zadávame inú časovú pečiatku. Je to vlastne hashovací odtlačok, podľa ktorého si komunikujúce strany overujú celistvosť správy. V našom programe generovanie znaku kontrolného súčtu zabezpečuje funkcia „GetBCC“ umiestnená v triede UsbService. Po odoslaní správy elektromer zašle požadované profilové dáta a v prípade, že je na zbernici viac elektromerov, zasiela jednotka master ukončujúci príkaz B0. Ak ho master nezašle, elektromer sám ukončí komunikáciu do 5 sekúnd. Načítané dáta aplikácia zobrazí na displeji a je možné ich uložiť do súboru.

4.5.4 Zápis a odčítanie času elektromera

Po stlačení tlačidla „T-READ“ sa spustí rovnaký proces správ ako pri odčítaní profilu s tým rozdielom, že pri poslednej správe odčítaní času sa zmení EDIS kód z P.01 na kód 0.9.1 a zátvorka zostane prázdna. Elektromer zašle späť správu s rovnakým kódom v tvare 0.9.1(163621), kde je v zátvorke čas elektromera vo formáte (hhmmss). Po doručení tejto správy aplikácia po 900 milisekundách opäť zašle správu s kódom 0.9.2, čo značí EDIS kód pre dátum. Elektromer odpovedá správou 0.9.2(160511), kde v zátvorke je dátum vo formáte (yymmdd). Po stlačení tlačidla „T-SET“ prebieha všetko v rovnakej sekvencii s tým rozdielom, že príznak čítania R5 sa zmení na príznak programovacieho režimu W5, teda „write“ s rýchlosťou 5, ktorá činí 9600 bd. Do zátvoriek nastaví aplikácia časové pečiatky aktuálneho dátumu a času, ktorý používa telefón. Pri potrebe nastaviť elektromeru iný čas než je aktuálny, je nutné zmeniť čas v telefóne.

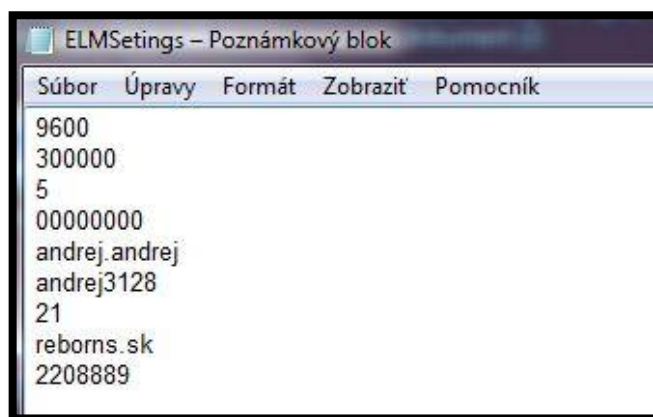
4.5.5 Nastavenia aplikácie

Keďže nie všetky elektromery pracujú na rovnakej komunikačnej rýchlosti a takisto nastavenia FTP servera je nutné zmeniť podľa používateľa, je možnosť nastaviť aplikáciu pomocou parametrizačného súboru. Do koreňového adresára telefónu je nutné pridať textový súbor s názvom ELMSetings.txt, ktorý musí mať dodržanú presnú štruktúru dát.

Tabuľka 3 Popis nastavení aplikácie

1.	Komunikačná rýchlosť Baud Rate
2.	Časový interval automatického odčítania registra
3.	Maximálna povolená rýchlosť komunikácie
4.	Heslo elektromera
5.	Prihlasovacie meno na FTP server
6.	Prihlasovacie heslo na FTP server
7.	Port FTP (defaultne 21)
8.	URL alebo IP adresa FTP servera
9.	Číslo elektromera

Načítanie súboru s nastaveniami je volané hneď po štarte aplikácie pomocou funkcie „onCreate“ v hlavnej triede. Trieda UsbService načítava svoje nastavenia ako napríklad komunikačnú rýchlosť pri nastavovaní atribútov sériovej komunikácie.



Obrázok 30 Ukážka nastavení

V prípade, že súbor s nastaveniami neexistuje alebo sa nedá načítať, aplikácia funguje v základných nastaveniach. Dôležité je dodržať presný formát a poradie nastavení v súbore, ako ukazuje obrázok 30.

4.6 Testovanie aplikácie

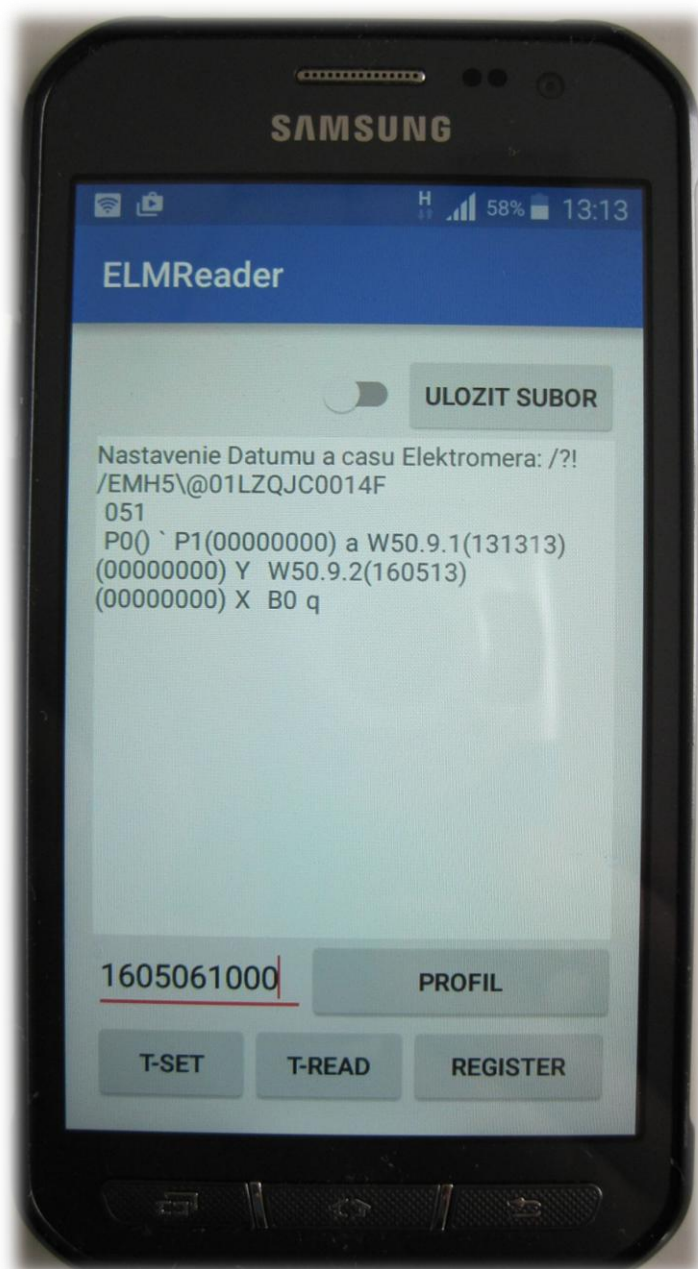
Testovanie našej aplikácie ELMReader prebiehalo v dvoch etapách. Prvá etapa spočívala v testovaní v laboratórnych podmienkach, druhá etapa už prebiehala v reálnom nasadení na mobilnom odbernom mieste železničnej lokomotívy radu 671. Pri testovaní v laboratórnych podmienkach bol vyskladaný merací model z mobilného telefónu Samsung Galaxi Xcover3, sériového prevodníka z USB na RS 485 a elektromera EMH LZQJ S1F4.



Obrázok 31 Testovanie aplikácie s elektromerom EMH LZQJ

Pri tomto modeli nebola meracia časť elektromera zapojená čo znamená, že elektromer nemeral žiadnu spotrebu a jeho namerané dáta obsahovali iba nuly. Hoci tento stav nebol úplne ideálny na testovanie správnej funkcionality našej aplikácie to bolo dostatočné. Elektromer mal pripojené pomocné napájacie napätie 230 V a prevodník MINI 5.6 aj so sériovou linkou RS 485 bol napájaný s telefónu Samsung. Ten vystupoval v danom zapojení v úlohe „Mastra“, ktorý napája sériovú linku a riadi na nej komunikáciu. Pri testovaní v tomto zapojení bol skúšaný odpočet registra elektromeru, ktorý prebehol úspešne a bez akýchkoľvek problémov, tiež odpočet záťažového profilu, pri ktorom nastal

problém pri výpočte kontrolného súčtu správy. Takzvaného kontrolného znaku BCC, ktorý určuje, že správa prišla po transporte sériovou linkou nepozmenená. V pôvodnej verzii aplikácie sa s počítaním tohto súčtu neuvažovalo, avšak bolo zistené, že elektromer vyžaduje výpočet kontrolného znaku pri každej správe ktorá sa mení. V našom prípade komunikácia zlyhávala pri správe s časovými pečiatkami odpočtu záťažového profilu. Po doprogramovaní výpočtu kontrolného znaku BCC podľa normy STN EN 62056-21 už komunikácia prebiehala bez problémov. Ďalej bol testovaný odpočet času elektromera a aj zápis času elektromera, ktorý prebehol bez problémov.



Obrázok 32 Nastavenie času Elektromera EMH LZQJ

Na obrázku 32 je vidieť na displeji kompletnú výmenu správ medzi elektromerom a telefónom, ktorá prebehla celá úspešne až po ukončovací znak komunikácie B0. Následne bol úspešne testovaný zápis nameraných dát do súboru a automatický režim aplikácie. Pri testovaní automatického režimu bol interval opakovania odpočtu nastavený na 120 000 milisekúnd, čo predstavuje 2 minúty. Aplikácia vykonala odpočet registra elektromeru, uloženie odpočtu do súboru v pamäti telefónu a súbor odoslala na vzdialený FTP server. Pri tomto režime je dôležité mať zapnuté mobilné dáta aby mohlo byť FTP spojenie vytvorené. Po zapnutí mobilných dát prebiehal automatický odpočet bez problémov. Bolo vykonaných 5 cyklov odpočtu. Všetky prebehli úspešne.

Ako druhé testovanie bolo zvolené testovanie v ostrej prevádzke na reálnom meracom systéme. Keďže aplikácia komunikuje s vzdialeným FTP serverom cez mobilné dáta jej výhoda spočíva v tom, že odberné miesto nemusí byť pripojené na LAN sieť pomocou kábla. Práve pre túto vlastnosť bolo aplikácia testovaná na železničnej lokomotive Škoda radu 671. Testovanie prebiehalo za prítomnosti strojvodu a zástupcu železničnej spoločnosti Slovensko a.s.. Jednalo sa o novú modernú elektrickú poschodovú jednotku, ktorá mala merací systém umiestnený v zadnej časti strojovne.



Obrázok 33 Lokomotíva elektrickej jednotky Škoda radu 671

Po inštalovaní skúšobnej meracej súpravy a po otestovaní manuálnych funkcií aplikácie bola vykonaná krátka skúšobná jazda. Pri tejto jazde bol spustený automatický režim aplikácie, ktorý prebehol v ôsmich dvojminútových cykloch. Mobilný telefón mal počas testovania SIM kartu distribútora elektriny, ktorá bola zaradená v APN sieti ŽSR. Po ukončení jazdy distribútor ŽSR potvrdil prijatie ôsmich súborov na svojom serveri, ktoré boli úspešne spracované energetickým informačným systémom EDW. To potvrdzuje okrem funkčnosti aplikácie aj správnosť formátu dát a názvu súboru.



Obrázok 34 Skriňa merania Škoda s inštalovaným telefónom Samsung

Pri inštalácii mobilného telefónu do meracej súpravy lokomotívy bol použitý USB/RS485 prevodník MINI 5.6.



Obrázok 35 Inštalácia telefónu do meracej skrine lokomotívy Škoda

Telefón bol v skrini merania dočasne provizórne uchytený a svorky A a B boli privedené na internú sieť RS485. Pri testovaní aplikácie musel byť odpojený priemyselný počítač „Skalar“, ktorý je vidieť na obrázku 35 hneď nad mobilným telefónom. Odpojenie skalára bolo nutné z dôvodu, že dva vysielacie typu „Master“ by sa mohli pri odpočtoch na linke rušiť. Pri testovaní manuálneho módu aplikácie sa vyskytol problém s odpočtom profilových dát. Bolo to z dôvodu príliš malých offsetov pri výmene správ s elektromerom. Kým elektromer EMH na ktorom sa aplikácia vyvíjala mal komunikačnú rýchlosť 9600 bd, elektromer na lokomotíve komunikoval iba na rýchlosti 1200 bd a teda nestíhal na správy zasielané aplikáciu reagovať. Preto bolo nutné hneď na mieste tento problém programátorsky odstrániť. Po prispôbení časových offsetov už elektromer komunikoval bez problémov. Taktiež sa podarilo čas elektromera bez problémov odčítať a aj nastaviť. A teda môžeme konštatovať, že aj v ostrej prevádzke na inom type elektromera sa aplikácia osvedčila ako plne funkčná.

ZÁVER

Cieľom tejto diplomovej práce bolo navrhnúť dvojúčelovú aplikáciu, ktorá by dokázala čiastočne nahradiť momentálne používané priemyselné počítače, takzvané dátové koncentrátory, slúžiace pre zber a odosielanie dát na centrálu energetickej distribučnej spoločnosti. Pre druhý účel využitia aplikácie bolo treba navrhnúť a zostrojiť nástroj z najčastejšie používanými funkciami pri kontrole a servise chytrých elektromerov. Okrem samotnej aplikácie bolo treba zvoliť vhodné komunikačné rozhranie, štruktúru zasielaných dát a protokol prenosu dát po internete. Pre vývoj a testovanie aplikácie bol použitý mobilný telefón Samsung Galaxi Xcover3 a elektromer EMH LZQJ S5F4. Operačný systém, pre ktorý bola aplikácia primárne navrhnutá, bol Android 4.4 Kitkat s verziou API 20. Ako komunikačné rozhranie bolo zvolené sériové rozhranie elektromeru RS 485 a k nemu bol použitý prevodník z RS 485 na USB ComErgon MINI 5.6.. Pre odosielanie odčítaných dát z telefónu na centrálu distribútora elektriny bol zvolený FTP protokol. FTP pripojenie bolo programované pomocou knižnice Apache Commons Net. Inicializáciu a vytvorenie sériovej komunikácie pomáhala zabezpečovať voľne dostupná knižnica com.felhr.usbserial. Riadenie komunikácie a obsluha sériového portu RS485 musela byť programovaná manuálne. Ovládač pre komunikáciu s elektromerom po sériovej linke bol programovaný podľa normy STN EN 62056-21. Vytvorená aplikácia ELMReader je nezávislá od výrobcu či typu elektromera, čo bolo potvrdené testovaním na univerzálnom elektromere EMH LZQJ v laboratórnych podmienkach a zároveň bola aplikácia úspešne otestovaná počas skúšobnej jazdy na železničnom koľajovom vozidle ŠKODA radu 263. Jediné problémy počas testovania na rôznych typoch sa vyskytli s rozdielnym nastavením prenosových rýchlostí. Kvôli tejto skutočnosti musel byť upravený časový offset riadenia komunikácie sériovej linky. V manuálnom móde ponúka aplikácia možnosť odčítavania registrov elektromera, záťažového profilu v ľubovoľnom časovom rozsahu a odčítanie a synchronizáciu interných hodín elektromera. Automatický režim, ktorý bol úspešne otestovaný na koľajovom vozidle ŠKODA, v pravidelných cykloch zbiera registrové dáta elektromera a odosiela ich pomocou mobilnej dátovej siete na FTP server distribútora elektriny, kde môžu byť dáta ďalej spracovávané. Pri testovaní v ostrej prevádzke sa vyskytol problém s dobíjaním batérie telefónu pri pripojenom sériovom prevodníku. Problém je možné riešiť malou úpravou prevodníka s 5-voltovým stabilizátorom napätia. Ten môže byť pripojený na napájacie svorky USB zbernice telefónu, čím sa zabezpečí dobíjanie telefónu a zároveň externé napájanie sériovej linky.

ZOZNAM POUŽITEJ LITERATÚRY

- [1] ALLEN, Grant. *Android 4: průvodce programováním mobilních aplikací*. 1. vyd. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6.
- [2] SCHILDT, Herbert. *Java 7: výukový kurz*. 1. vyd. Brno: Computer Press, 2012. ISBN 978-80-251-3748-2.
- [3] SCHILDT, Herbert. *Mistrovství - Java*. 1. vyd. Brno: Computer Press, 2014. Mistrovství. ISBN 978-80-251-4145-8.
- [4] PITNER, Tomáš. *Java - začínáme programovat: podrobný průvodce začínajícího uživatele*. 1. vyd. Praha: Grada, 2002. ISBN 80-247-0295-9.
- [5] UJBÁNYAI, Miroslav. *Programujeme pro Android*. Vyd. 1. Praha: Grada, 2012. Průvodce (Grada). ISBN 978-80-247-3995-3.
- [6] SMERNICA EURÓPSKEHO PARLAMENTU A RADY 2009/72/ES z 13. júla 2009 o spoločných pravidlách pre vnútorný trh s elektrinou, ktorou sa zrušuje smernica 2003/54/ES. Úradný vestník Európskej únie, 2009.
- [7] STN EN 62056-21. Meranie elektrickej energie. Výmena údajov na odčítanie elektromerov, na ovládanie sadzieb a zaťaženia. Prešov: Križik a.s., 2009.
- [8] Vyhláška č. 358/2013 Z. z. Vyhláška Ministerstva hospodárstva Slovenskej republiky, ktorou sa ustanovuje postup a podmienky v oblasti zavádzania a prevádzky inteligentných meracích systémov v elektroenergetike. Vyhláška Slovenskej republiky, 2013.
- [9] DARBY, Sarah. Smart metering: what potential for householder engagement? *Building Research & Information* [online]. 2010, **38**(5), 442-457 [cit. 2016-04-20]. DOI: 10.1080/09613218.2010.492660. ISSN 0961-3218. Dostupné z: <http://www.tandfonline.com/doi/abs/10.1080/09613218.2010.492660>
- [10] EDITOR, Stuart Borlase. *Smart grids: infrastructure, technology, and solutions*. Boca Raton, FL: Taylor & Francis, 2012. ISBN 1439829055.
- [11] Oddelený RS485 - USB. In: comergon.sk [online]. [cit. 2016-04-15]. Dostupné z: <http://www.comergon.sk/index.php/sk/usb/mini-56485>
- [12] KOLCUN, Michal. Inteligentné siete [online]. [cit. 2016-03-22]. Dostupné z: http://people.tuke.sk/dusan.medved/Porac/2014/Kolcun_Dudiak.pdf

-
- [13] Android Studio. In: Wikipedia [online]. [cit. 2016-03-20]. Dostupné z: https://cs.wikipedia.org/wiki/Android_Studio
- [14] Informační a komunikační sítě. In: vse.cz [online]. [cit. 2016-02-10]. Dostupné z: <http://nb.vse.cz/~svabo/4IZ110/FTP.htm>
- [15] JURINA, Ľubomír. Aj elektrina vie byť inteligentná. In: tyzden.sk [online]. [cit. 2016-02-11]. Dostupné z: <http://www.tyzden.sk/casopis/11989/aj-elektrina-vie-byt-inteligentna/>
- [16] Samsung Galaxy Xcover 3 G388. In: f-mobil.sk [online]. [cit. 2016-03-10]. Dostupné z: <https://www.f-mobil.sk/telefon/samsung/samsung-galaxy-xcover-3-sm-g388f-stribrny>

ZOZNAM POUŽITÝCH SYMBOLOV A SKRATIEK

HDV Hnacie dráhové vozidlo.

Napr. Napríklad.

Tzv. Takzvaný.

Atd'. A tak ďalej.

PC Počítač

APK Aplikácia

IDE Integrated Development Environment

ŽSR Železnice Slovenskej republiky

ZOZNAM OBRÁZKOV

Obrázok 1 Inteligentná sieť [12].....	9
Obrázok 2 Inteligentné elektromery	15
Obrázok 3 Architektúra OS Android [1]	19
Obrázok 4 Vývojové prostredie Android Studio [13]	24
Obrázok 5 Parametre RS 485 [7].....	25
Obrázok 6 Napäťové úrovne RS 485 [7]	26
Obrázok 7 Prenos znaku 211 [7].....	26
Obrázok 8 Dvojvodičová linka RS 485 [7]	27
Obrázok 9 Schéma komunikácie FTP protokolu [14]	29
Obrázok 10 Telefón Samsung Galaxy Xcover 3 [16].....	31
Obrázok 11 Popis prevodníka USB/RS485 [11]	33
Obrázok 12 Technické parametre elektromera	35
Obrázok 13 Bloková schéma zapojenia elektromera LZQJ	36
Obrázok 14 Popis registru elektromera (Časť 1.)	37
Obrázok 15 Popis registru elektromera (Časť 2.)	38
Obrázok 16 Ukážka profilového záznamu elektromera	39
Obrázok 17 Elektromer EMH LZQJ XC.....	40
Obrázok 18 Železničný elektromer LEM	41
Obrázok 19 Popis grafického návrhu aplikácie	44
Obrázok 20 Ukážka XML kódu grafického návrhu	46
Obrázok 21 Nastavenia v súbore AndroidManifest.xml	47
Obrázok 22 Spracovávanie oznámení OS Android	50
Obrázok 23 Spracovanie oznámení o stave USB	51
Obrázok 24 Inicializácia sériového portu	52
Obrázok 25 Zber prichádzajúcich údajov zo sériového portu	53
Obrázok 26 Funkcia pre odosielanie súboru na FTP server	55
Obrázok 27 Nastavenie parametrov FTP prenosu v MainActivity.....	56
Obrázok 28 Automatický odpočet elektromera	57
Obrázok 29 Komunikácia medzi telefónom (vľavo) a elektromerom (vpravo)	59
Obrázok 30 Ukážka nastavení	61
Obrázok 31 Testovanie aplikácie s elektromerom EMH LZQJ	62
Obrázok 32 Nastavenie času Elektromera EMH LZQJ.....	63

Obrázok 33 Lokomotíva elektrickej jednotky Škoda radu 671	64
Obrázok 34 Skriňa merania Škoda s inštalovaným telefónom Samsung	65
Obrázok 35 Inštalácia telefónu do meracej skrine lokomotívy Škoda	66

ZOZNAM TABULIEK

Tabuľka 1 Popis tried android.hardware.usb	48
Tabuľka 2 Veta na oslovenie elektromera [7]	58
Tabuľka 3 Popis nastavení aplikácie	61