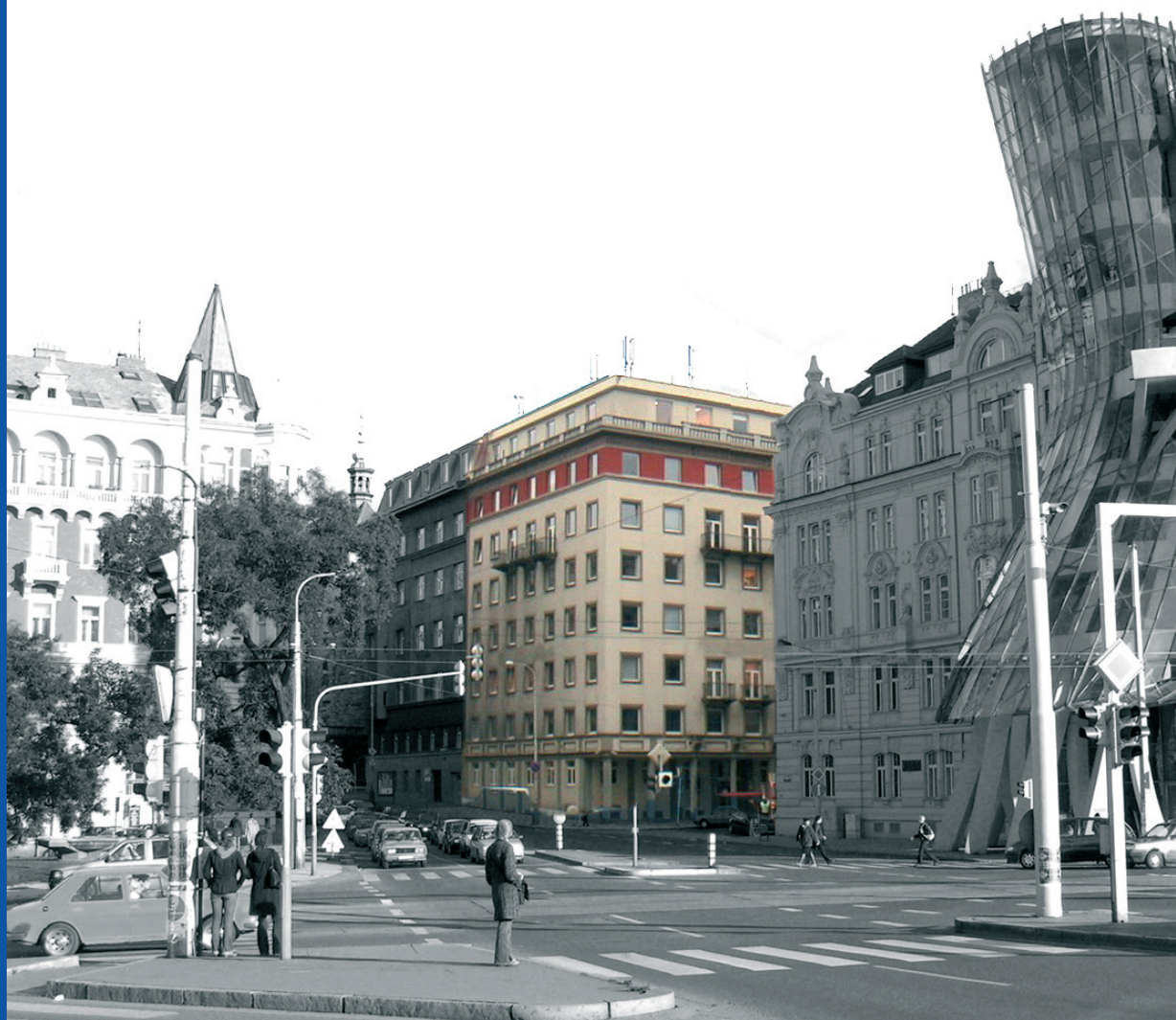
ÚNMZ

SBORNÍKY TECHNICKÉ HARMONIZACE
2006

WELMEC GUIDE 7.1
(druhé vydání – překlad)



Vážení čtenáři a kolegové,

od r. 1996 vydával Úřad pro technickou normalizaci, metrologii a státní zkušebnictví edici nazvanou „K vnitřnímu trhu Evropské unie“. Většina svazků se těšila zcela mimořádné pozornosti a zájmu.

Cílem vydávání této edice bylo přiblížit technické veřejnosti principy a procedury technické legislativy, zaváděné v souladu s harmonizačními procesy v Evropské unii (EU) i v České republice. I když dnes existují daleko širší zdroje informací, než tomu bylo před několika lety, považujeme za potřebné v této iniciativě pokračovat, neboť jsme přesvědčeni, že napomáhá pochopení právní úpravy v oblastech působnosti ÚNMZ a jejímu správnému uplatňování. Navíc existuje řada dokumentů, které nejsou součástí práva, ale jsou důležité pro praxi. I v mnoha státech EU je technická regulace a harmonizace doprovázena ze strany státních orgánů širokou informační kampaní.

Proto je od roku 2004 vydávána inovovaná edice, přizpůsobená svým zaměřením aktuálnímu vývoji, podmínkám a potřebám. Byl zaveden nový název edice, který zní „Sborníky technické harmonizace ÚNMZ“, nová grafická podoba, i forma distribuce. Edice je k dispozici na stránkách ÚNMZ (www.unmz.cz) a v omezeném počtu nebo na vyžádání je též využívána forma CD-ROM. Je volně dostupná při respektování autorských práv.

Uplynulé dva roky, kdy v této edici bylo vydáno deset Sborníků, zatím prokázaly pokračující zájem odborné veřejnosti a redakční rada má i řadu námětů do blízké budoucnosti.

Věřím, že jak orgány státu, tak soukromá sféra resp. všichni účastníci procesu technické harmonizace a regulace budou v této edici i nadále nacházet užitečný zdroj informací a pomocníka v jejich práci.

Vaše podněty vedoucí k dalšímu zkvalitnění této činnosti ÚNMZ s povděkem uvítáme.

Ing. Alexander Šafařík-Pštrosz,
předseda ÚNMZ

Praha, 2006

OBSAH SBORNÍKU

1 ÚVODNÍ KOMENTÁŘ..... 4
2 WELMEC GUIDE 7.1..... 10

METROLOGICKÉ POŽADAVKY NA ZKOUŠENÍ SOFTWARU

Úvod

Stále se zdokonalující a množící se mikroelektronické komponenty v měřicí technice přinášejí do jisté míry přesun funkčnosti měřicích zařízení z hardwarové do softwarové oblasti. Protože technologie výroby mechanických a elektrických komponent nejrůznějších senzorů je v současné době na velmi vysoké úrovni, dochází často k situacím, kdy jsou funkčnost a metrologické vlastnosti přístrojů dány především jejich programovým vybavením. Nejvýraznější je tento trend u měřicích systémů založených na využití volně programovatelných osobních počítačů.

Moderní programové vybavení měřicí techniky si žádá vývoj nových zkušebních metod, které by mohly být použity k testování metrologických vlastností softwaru. Při testování softwaru je třeba brát ohled na některá specifika, kterými se liší od běžných měřicích zařízení.

a) složitost softwaru – běžné programy mohou vykonávat větší množství rozdílných operací než čistě hardwarová zařízení. Z toho vyplývá nebezpečí vzájemného ovlivňování více programů nebo různých částí jednoho programu. I relativně krátké programy (ve smyslu délky zdrojového kódu) mohou být velmi složité.

b) změny – na rozdíl od fyzického zařízení je software často – a velmi jednoduše – vylepšován pomocí bezpečnostních záplat, nových funkcí a modulů. Takové změny se mohou promítnout do ostatních částí programu. Stejně tak změny operačního systému mohou ovlivnit program, neboť ten využívá volání systémových funkcí.

c) chyby – chyby softwaru nastávají většinou nečekaně, bez možnosti včas detekovat blížící se selhání. Způsob, jakým se software vyrovná s chybou, závisí jen na zodpovědnosti výrobce a kvalitě návrhu a implementace softwaru a operačního systému.

d) standardizace – vývoj softwaru většinou nepodléhá žádným standardům, použité nástroje závisí na výrobci a je těžké (a zpětně nemožné) je ovlivnit. Úroveň dokumentace také závisí jen na výrobci, resp. na požadavcích zákazníka.

Testování softwaru – validaci – provádíme většinou pro účely typového schválení daného měřicího přístroje, pro splnění požadavků na akreditaci laboratoře či pro splnění požadavků kladených z hlediska managementu jakosti dané organizace.

V tomto dokumentu jsou popsány základní koncepty zkoušení softwaru v rámci směrnice Measurement Instruments Directive (MID) a souvisejícího dokumentu WELMEC 7.2.

Požadavky na software v legální metrologii–dokumenty WELMEC WG7

České předpisy pro požadavky na software v přístrojích používaných v oblasti legální metrologie jsou významně ovlivněny členstvím ČR ve sdružení WELMEC (Western European Legal Metrology Cooperation), které se zabývá sjednocováním metrologických předpisů používaných v legální metrologii.

Jak v rámci typového schvalování na základě směrnice MID (Measurement Instruments Directive), tak v rámci vývoje jiných předpisů je vycházeno především z dokumentu Welmec Software Guide, který je v tomto sborníku prezentován ve své české i anglické verzi.

Dokument WELMEC 7.2 popisuje jednotlivé požadavky na validaci softwaru s tím, že zavádí různé třídy rizika (risk classes) a k nim odpovídající požadavky. Je definováno celkem šest tříd rizika (A–F), přičemž naprostá většina přístrojů je zařazena do tříd B–D. Toto zařazení je výsledkem práce technických komisí WELMEC, které se zabývají jednotlivými měřicími přístroji. Třídy rizika i požadavky kladené na software závisí zejména na tom, zda se jedná o samostatný jednoúčelový přístroj (typ P instrument), nebo o přístroj využívající osobní počítač (typ U instrument). Pro přístroje typu U jsou požadavky obecně vyšší, neboť jsou vystaveny většímu nebezpečí ze strany uživatele.

Každý zkoušený přístroj musí být jednoznačně zařazen do některé z uvedených tříd rizika, aby bylo možné na něj aplikovat požadavky vyplývající z kapitol 5–10. Každá třída rizika odpovídá různým úrovním ochrany, zkoušení a shody, přičemž každý z těchto faktorů může mít tři stupně: nízký, střední a vysoký.

Požadavky na software v legální metrologii uvedené v dokumentu WELMEC 7.2 je přitom možné přeneseně aplikovat (po volbě vhodné třídy rizika) i na další software.

V souvislosti s přehledem požadavků na software je v tomto sborníku prezentován také informativní dokument WELMEC 7.1, který lze po revizích považovat za předchůdce dokumentu WELMEC 7.2. V tomto dokumentu jsou blíže rozvedeny některé koncepty tříd rizika a slouží zároveň jako reference pro zajištění kontinuity se staršími předpisy na validaci softwaru (které právě z dokumentu WELMEC 7.1 často vycházely).

Metody zkoušení softwaru

Vlastní testování závisí na charakteru softwaru a na cíli validace. Pokud provádíme validaci softwaru v legální metrologii, zaměřujeme se zejména na některé parametry zabezpečení programu a dat a na shodu funkčnosti s dokumentací, v souladu s doporučeními Welmec 7.2. V dokumentu samotném je podrobný návod, podle kterého je možné postupovat, z pohledu validace v oboru legální metrologie je tedy dostatečně zajištěna jednotnost přístupu při kontrole softwaru.

Praktické metody validace softwaru můžeme rozdělit do dvou skupin – na analýzu statickou a dynamickou. K těmto metodám souvisejícím s funkčností softwaru se v rámci přístupu dokumentu WELMEC 7.2 připojuje metoda třetí – kontrola dokumentace dodané výrobcem.

Jak vyplývá z požadavků dokumentu WELMEC 7.2, pro nižší třídy rizika ve většině případů aplikujeme pouze kontrolu dokumentace spojenou s dynamickou analýzou. Důraz je však kladen na kontrolu dokumentace, zahrnující kromě dokumentace uživatelské také deklarace výrobce speciálně vydané pro účely validace. Při přechodu do vyšších tříd rizika stoupá důraz na dynamickou analýzu a u nejvyšších tříd i statickou analýzu spolu se zvyšujícími se nároky na software.

V praxi jsou jednotlivé metody zkoušení softwaru prováděny například následujícím způsobem:

Statická analýza: cílem je kontrola kódu se zřetelem na jeho správnost a přehlednost. Je také kontrolován tok dat v programu a jsou hledány případné chyby v zacházení s daty a s pamětí. Postupujeme následujícím způsobem:

1. zběžná kontrola kódu, ověření jeho úplnosti, pokud je to možné, jeho překlad s přidáním informací pro debugger a kontrolou všech varování kompilátoru. Je vhodné použít kompilátor, který je použit v praxi (pokud nejde o software distribuovaný ve formě zdrojových souborů, který může být přeložen libovolným kompilátorem),
2. kontrola přehlednosti zdrojového kódu a jeho vhodného dělení do funkcí a modulů,
3. vyhledání toku dat v programu a kontrola jeho oddělení od ostatních částí programu (zejména se týká metrologických aplikací). Kontrola všech větví a cyklů souvisejících s tokem dat a kontrola zabezpečení dat (zejména při validaci zabezpečení). Můžeme definovat následující metriky pokrytí (např. 100% pokrytí větví pak znamená, že všechny větve byly kontrolovány a při dynamické analýze pak spuštěny):
   - pokrytí příkazů – všechny příkazy (ve zdrojovém kódu) jsou testovány a spuštěny,
   - pokrytí větví – všechny větve programu (ve zdrojovém kódu) jsou testovány a spuštěny,
   - pokrytí podmínek – všechny podmínky (např. ve větvení programu, při kontrole dat atd.) jsou testovány a spuštěny,
   - pokrytí cyklů – všechny cykly jsou spuštěny jednou, dvakrát a mnohokrát, je také kontrolováno, zda program pracuje dobře, pokud neproběhnou vůbec,
   - pokrytí cest – jsou kontrolovány všechny možné cesty programu (zahrnuje všechny větve, podmínky a jejich kombinace),
4. kontrola modularity programu (oddělení funkcí zabývajících se různými formami zpracování dat do samostatných modulů); kontrola, zda moduly vzájemně interagují jen očekávaným způsobem,
5. kontrola modulů – oddělení samostatných částí kódu zabývajících se zpracováním dat a statistické simulování vstupů a výstupů pro tyto části programu,
6. kontrola ošetření výjimek a nestandardních situací v programu,
7. testování vhodnosti – zahrnuje především testy určené uživatelem (z hlediska jeho požadavků), testy rychlosti, využití zdrojů, implementace zálohování, zaznamenání dat a podobných funkcí, které souvisí s uživatelským rozhraním. Testujeme, zda jsou funkce implementovány podle požadavků uživatele,

8. kontrola zabezpečení – sleduje se tok dat v programu, implementaci jednotlivých algoritmů zabezpečení (digitální podpisy, šifrování, kontrolní součty, správa uživatelů a jejich hesel atd.). Kontrolujeme, nakolik je program schopen sám detekovat nežádoucí změny a vytvářet svou identifikaci, zda je možné jej spustit s pozměněnými daty či knihovnami.

Pro účely statické analýzy je vhodné použít softwarové nástroje pro sledování toku dat v programu, debuggery a programy kontrolující práci s pamětí.

Dynamická analýza: cílem je ověřit funkčnost a stabilitu programu a jeho shodu s dokumentací (případně dokumentaci nepopsaných či špatně popsaných částí na místě vytvořit). Test musí být přednostně proveden přímo v prostředí, ve kterém bude program používán, případně v simulovaném podobném prostředí (operační systém, hardware). Postupujeme následujícím způsobem:

1. identifikace binární verze programu, např. pomocí cyklického součtu a zálohování kopie programu jako reference aktuální validované verze,
2. předvedení typického užití programu zákazníkem s důrazem na běžně prováděné operace. V případě, že jde o program, který je určen k další distribuci, je testována také instalace programu a jeho inicializace,
3. kontrola všech větví a cyklů programu (pokud je to z technického hlediska možné) alespoň jedním průchodem,
4. oddělené vyhodnocení co možná nejširší množiny vstupů a kontrola výstupů. V ideálním případě (nezávislost na hardwaru) statistické testování velkým množstvím náhodě generovaných dat,
5. vyvolání kritických situací (chybějící hardware, špatná vstupní data) a kontrola výstupů, kontrola, zda program zaznamenává kritické situace a zda je možné vzniklé záznamy modifikovat,
6. kontrola dokumentace programu a její shody se skutečným stavem.

Kontrola dokumentace

Dokumentací výrobce dodanou pro účely validace rozumíme většinou uživatelské manuály, specifikace měřicího přístroje a sadu deklarací a výčtů funkcí měřidla zhotovenou podle požadavků v příslušné třídě rizika (např. kompletní výčet všech příkazů komunikačního rozhraní měřidla a deklaraci úplnosti tohoto výčtu). Součástí dokumentace mohou být také popisy algoritmů (pro některé třídy rizika je postačující kontrola popisu algoritmů, bez statické či dynamické analýzy).

Konkrétní průběh validace závisí na typu softwaru, který je zkoušen, dostupnosti jeho zdrojových kódů a na účelu, pro který se validace provádí. Výstupem validace je tzv. *protokol o validaci*, ve kterém jsou popsány jednotlivé provedené testy a jejich výsledky. Nejedná se tedy jen o stanovisko vyhověl–nevyhověl, ale o celkový popis zkoušek, které byly na konkrétním programovém vybavení provedeny. Podle doporučení Welmec 7.2 je forma tohoto protokolu pevně dána specifikací jeho nezbytných součástí.

Závěr

Validace softwaru je soubor úkonů, které by měly zajistit správnou a bezpečnou funkcionalitu programového vybavení používaného v oblasti metrologie. Pro účely schvalování měřidel podle směrnice Measurement Instruments Directive (MID) je v tomto sborníku uveden dokument WELMEC 7.2, který kompletně popisuje požadavky na vývoj a zkoušení softwaru v měřicích přístrojích pokrytých touto směrnicí.

WELMEC GUIDE 7.1

Vývoj požadavků na software

Zkoušení softwaru na základě směrnice MID

(druhé vydání – překlad)

WELMEC je kooperace mezi autoritami na poli legální metrologie ze zemí Evropské unie a sdružení EFTA. Tento dokument je jedním z návodů, jejichž cílem je vytvořit vodítko pro výrobce a notifikované osoby pracující na poli zajištění shody. Tyto návody jsou čistě doporučujícího charakteru a samy o sobě nepřinášejí žádné další požadavky, které by byly nad rámec direktiv Evropské komise. Problémy popisované v těchto dokumentech mohou mít alternativní přijatelná řešení a zde uvedená řešení je třeba brát jen jako dobré příklady vyhovující daným požadavkům.

Přeloženo z anglického originálu:
WELMEC 7.1 Issue 2
Informative Document Development of Software Requirements,
May 2005

Vydaného:
WELMEC Secretariat, BEV,
Arltgasse 35, A-1160 Vídeň, Rakousko
Tel: +43 1 21176 3608, Fax: +43 1 49 20 875
E-mail: welmec@metrologie.at
Web: www.welmec.org

Přeložil: Mgr. Petr Klapetek, Ph.D.

Vydal Úřad pro technickou normalizaci, metrologii a státní zkušebnictví se souhlasem WELMEC a při zachování všech práv WELMEC. Originál dokumentu v anglickém jazyce je umístěn na výše uvedených webových stránkách WELMEC.

OBSAH WELMEC GUIDE 7.1

PŘEDMLUVA 13
1 ÚVOD....14
1.1 Motivace....14
1.2 Koncepce....14
2 TERMINOLOGIE....16
2.1 Kód programu....16
2.2 Software relevantní z pohledu legální metrologie....16
2.3 Změny softwaru....19
2.3.1 Neúmyslné změny....19
2.3.2 Úmyslné změny jednoduchými nástroji....19
2.3.3 Úmyslné změny sofistikovanými nástroji....19
2.4 Ochrana softwaru....21
2.4.1 Chráněný software....21
2.4.2 Kontrolní čítač....21
2.5 Rozhraní....21
2.5.1 Hardwarové rozhraní....21
2.5.2 Ochranné rozhraní....21
2.5.3 Softwarové rozhraní....22
2.5.4 Ochranné softwarové rozhraní....22
2.6 Bezpečnost dat....22
3 ZÁKLADNÍ POŽADAVKY NA SOFTWARE....25
4 DEFINICE ÚROVNÍ....28
4.1 Úroveň ochrany softwaru....29
4.2 Úroveň zkoušení softwaru....29
4.3 Úroveň shody softwaru....30
5 TECHNICKÉ VLASTNOSTI MĚŘICÍCH PŘÍSTROJŮ A SYSTÉMŮ....32
5.1 Hardwarová konfigurace....32
5.2 Uživatelské rozhraní....34
5.3 Nahrávání softwaru....34
5.4 Struktura softwaru....34
5.5 Prostředí....34
5.6 Detekce chyb....35
5.7 Dlouhodobé uchovávání naměřených hodnot....35
5.8 Měřicí princip....35

5.8.1 Časová závislost....35
5.8.2 Opakovatelnost....35
5.8.3 Složitost....35
6 INTERPRETACE ZÁKLADNÍCH POŽADAVKŮ PRO VYBRANÁ MĚŘIDLA A SYSTÉMY....36
6.1 Příklad A: Jednoduchý jednoúčelový měřicí přístroj....37
6.1.1 Popis přístroje....37
6.1.2 Klasifikace z pohledu legální metrologie....38
6.1.3 Technická klasifikace....39
6.1.4 Interpretace základních požadavků....39
6.2 Příklad B: Složitý měřicí systém využívající PC....45
6.2.1 Popis přístroje....45
6.2.2 Klasifikace z pohledu legální metrologie....47
6.2.3 Technická klasifikace....47
6.2.4 Interpretace základních požadavků....48
7 LITERATURA....65
8 REVIZE TOHOTO DOKUMENTU....66

PŘEDMLUVA

Cílem tohoto revidovaného dokumentu je poskytnout informace týkající se vývoje softwarových požadavků založených na směrnici MID (Measuring Instruments Directive). Až do vydání směrnice MID byl tento dokument v některých zemích používán jako základ národních předpisů pro schvalování typu. Skupina WELMEC 7 (Software) proto rozhodla, že bude nadále ponechán jako informativní dokument doplněný referencemi vztahujícími se ke směrnici MID a k dokumentu WELMEC 7.2.

Pro testování softwaru u přístrojů, kterých se dotýká směrnice MID, by měl být použit dokument WELMEC 7.2.

1 ÚVOD

1.1 Motivace

Ve směrnici MID (Measurement Instruments Directive) [1] jsou uvedeny „základní požadavky" pro měřidla používaná v oblasti legální metrologie. Některé z těchto základních požadavků mohou být přímo aplikovány na software ovládající tyto přístroje, další jak na software, tak na hardware obsažený v přístrojích.

V průběhu vývoje směrnice MID bylo zjištěno, že již zmíněné „požadavky na software" jsou nezbytné, pokud má být zamezeno rozdílnému zacházení se softwarem v měřicích přístrojích ze strany různých notifikovaných osob.

Po publikování dokumentu WELMEC 7.2 má tento dokument pouze informativní charakter. Byl revidován tak, aby souhlasil s konečnou podobou směrnice MID a s výsledky projektu MID-Software. Seznam změn je uveden v kapitole 8.

Smyslem tohoto dokumentu (stejně jako dokumentu WELMEC 7.2) je přesvědčit čtenáře, že při současném stavu měřicí techniky, kdy jsou přístroje založené na využití mikroprocesorů nebo dokonce osobních počítačů, již není dostačující zkoušet přístroje bez speciálního zřetele na software, který obsahují a který zásadní měrou ovlivňuje jejich metrologické vlastnosti. Vzhledem k tomu, že tyto dokumenty popisují velké množství typů přístrojů, jsou jednotlivé požadavky definovány co možná nejobecněji.

Smyslem tohoto dokumentu je podpořit rozvoj testování software tak, aby byl prováděn v celé Evropě jednotně a průhledně. Nejedná se ovšem o závazný dokument, a to ani pro přístroje zahrnuté do směrnice MID.

1.2 Koncepce

Kapitola 2 shrnuje základní terminologii.

V kapitole 3 jsou uvedeny „základní požadavky na software" odvozené ze směrnice MID, přílohy I. Tyto požadavky úzce souvisejí se základními požadavky směrnice MID. Pro praktické aplikace pro jednotlivé konkrétní přístroje je nezbytné je dále rozvést, zejména na základě specifických požadavků na přístroje (hardwaro-

vé a softwarové konfigurace), uvedených v jednotlivých přílohách směrnice MID, týkajících se konkrétních přístrojů.

V praxi schvalování typu bylo zjištěno, že požadavky na jednotlivé typy měřicích přístrojů se v některých případech liší i v rámci jednoho státu, a ještě častěji se bez jakýchkoliv objektivních důvodů liší v rámci jednotlivých zemí Evropské unie. Proto jsou v kapitole 4 zmíněna fakta a kritéria, ve kterých dochází k neshodám, a to zejména:

- úroveň ochrany softwaru proti změnám,
- intenzita zkoušení softwaru v rámci schvalování typu,
- úroveň shody mezi schváleným softwarem a softwarem v jednotlivých přístrojích.

Kritéria jsou přehledně seřazena do skupin měřicích přístrojů, které by mohly poskytnout nezávazné vodítko při vytváření požadavků na zkoušení softwaru.

V kapitole 5 jsou popsány technické aspekty měřicích přístrojů a systémů, které musí být brány v potaz při zkoušení softwaru. V kapitole 6 jsou uvedeny příklady zkoušení dvou typických měřicích přístrojů:

- jednoduchý jednoúčelový přístroj s ochranným rozhraním
- složitý modulární systém využívající PC

Příklady obsahují technický popis přístrojů i jejich popis z pohledu legální metrologie a související interpretaci požadavků na zkoušení softwaru s příslušnými komentáři. Uvedena je i požadovaná dokumentace. Z výše zmíněných důvodů tento dokument neobsahuje úplný přehled požadavků na jednotlivé druhy měřicích přístrojů.

Kapitola 7 obsahuje seznam literatury a dalších odkazů k rozšiřujícímu studiu.

2 TERMINOLOGIE

Poznámka: ***pokud se některá definice liší od definice z dokumentu WELMEC 7.2, je upřednostňováno znění z dokumentu WELMEC 7.2.***

2.1 Kód programu

zdrojový kód: kód programu v čitelné podobě (například textovým editorem) [14]

kód spustitelného souboru: sekvence binárních čísel, kterou čte a interpretuje procesor. Není čitelný bez pomoci speciálních nástrojů (debugger, disassembler atd.) [14]

2.2 Software relevantní z pohledu legální metrologie

Software, který realizuje měřicí funkci ve smyslu článku 1 dokumentu MID. Obsahuje části programu a data, které jsou předmětem kontroly z pohledu legální metrologie.

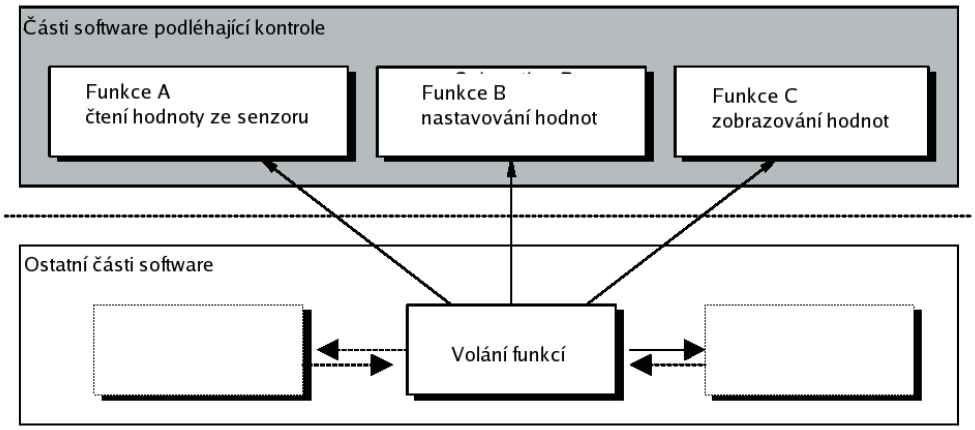
Části programu relevantní z pohledu legální metrologie

Části programu vykonávající funkce, které jsou předmětem kontroly z pohledu legální metrologie. Na obrázku 2.1 nad čarou jsou znázorněny tři takové části softwaru, pod čarou jsou znázorněny tři části softwaru, které nejsou předmětem kontroly. Šipky ukazují, jak se funkce navzájem volají. Namísto o části jednoho počítačového programu by se mohlo jednat taktéž o zcela samostatně spustitelné soubory.

Poznámky:

a) tato struktura softwaru je pouze doporučením, nicméně poskytuje velké množství výhod a souvisí se základními koncepcemi programování, jako je „modulární programování" a „objektově orientované programování", a je podporována řadou programovacích jazyků (C/C++, Java, Delphi, Visual Basic...),

b) pro „střední" úroveň shody se může část softwaru, která je předmětem kontroly z pohledu legální metrologie, skládat z více programů, z nichž některé mohou být neměnné (viz odstavec 4.3).



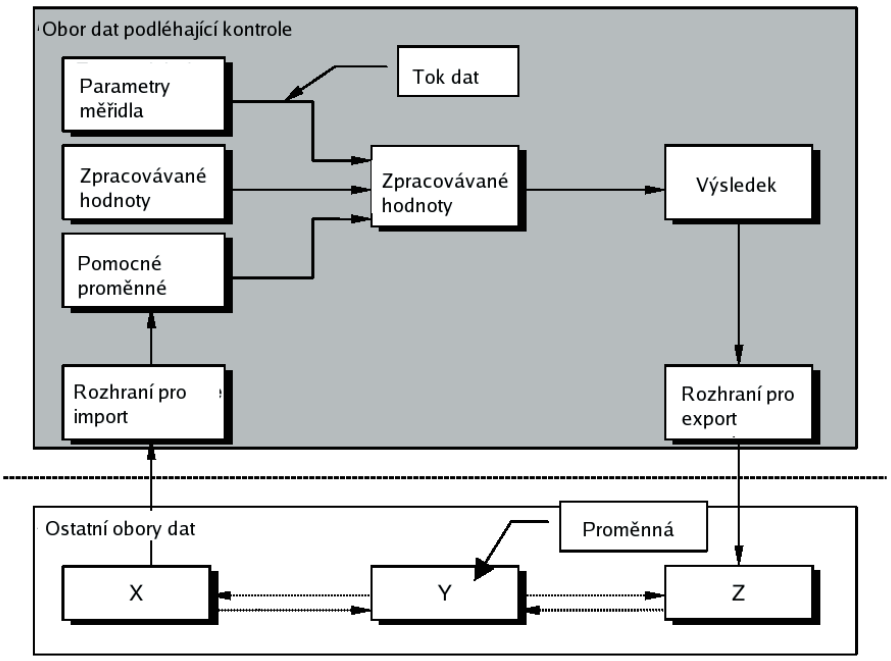
Obr. 2.1: Příklad rozdělení softwaru na části podléhající a nepodléhající kontrole

Data relevantní z pohledu legální metrologie

Tato data mohou být rozdělena do následujících skupin:

- parametry specifické pro typ přístroje jsou dány při schválení typu a většinou jsou v praxi součástí kódu programu,
- *specifické parametry přístroje: jsou různé pro každý jednotlivý přístroj, jedná se zejména o nastavení přístroje (citlivost, korekce) a další metrologické parametry, jako je konfigurace (měřicí rozsah, jednotky měření, rozlišení čtení).*
  Poznámka: *tyto parametry musí být většinou zabezpečeny.*
- nastavitelné parametry: data, která mohou být měněna uživatelem a zadávána v provozu,
- proměnné: jsou to zpracovávaná data a zpracované výsledky měření, pomocná data, jako jsou čítače, nastavení toku dat, obecně tedy proměnné z oboru dat podléhajících kontrole z pohledu legální metrologie.

Příklady funkcí a dat relevantních z pohledu legální metrologie jsou uvedeny v tabulce 2.1.



Obr. 2.2: Příklad rozdělení oborů dat

Pro programy a funkce můžeme definovat obor dat. Skládá se ze všech proměnných a konstant programu, které může program číst, případně do nich i zapisovat. Může být vlastní jen programu (či funkci) nebo do něj mohou mít přístup i jiné programy.

Obrázek 2.2 zobrazuje obor dat pro část programu relevantní z pohledu legální metrologie (nad čarou) a další části (pod čarou). Sdílení dat mezi jednotlivými částmi je realizováno výlučně přes importní–exportní rozhraní, jiný způsob propojení mezi jednotlivými obory dat není možný.

Poznámky:

a) pokud je program navržen podle obrázku 2.1, musí být obory dat rozděleny z pohledu relevantnosti v legální metrologii,

b) pokud jsou data přenášena nebo ukládána, ocitají se mimo obor dat programu podléhajícího kontrole a musí na ně být aplikovány další požadavky (v závislosti na jejich dalším použití v legální metrologii. Viz odstavec 5.7 a příklad B 6.2.4, ER2.2).

2.3 Změny softwaru

2.3.1 Neúmyslné změny

Změny softwaru nebo dat, které podléhají kontrole z pohledu legální metrologie, způsobené náhodnými a neúmyslnými hardwarovými či softwarovými událostmi (havárie, virové infekce).

2.3.2 Úmyslné změny jednoduchými nástroji

Změny softwaru uskutečněné obecně známými prostředky, jako jsou textové editory, bez zvláštních znalostí. Do této skupiny nepatří sofistikované nástroje, např. debuggery nebo diskové editory.

2.3.3 Úmyslné změny sofistikovanými nástroji

Změny nebo simulace softwaru nebo dat, které podléhají kontrole z pohledu legální metrologie, vytvořené softwarovými nástroji, které nejsou obecně rozšířené a používané a vyžadují zvláštní znalosti. Do této kategorie spadají změny debuggery, diskovými editory, prostředky pro vývoj softwaru, apod.

leg. relevantní funkce	parametry specifické pro typ přístroje	specifické parametry přístroje	nastavitelné parametry	proměnné
algoritmus výpočtu konečné hodnoty	korekce nelinearity	citlivost	tára	konečná hodnota k zobrazení
		jednotky měření		
		digitální rozlišení, interval stupnice		průběžná měřená hodnota
		rozsah měření		
analýza stability měřené hodnoty	časová konstanta			stavové signály (nula, stabilita průměru)
počítání pulzů u kumulativního měření		impulzní faktor		proměnná čítače
výpočet maxima	délka bufferu	doba měření		buffer pro všechny měřené hodnoty
				mezivýsledek
funkce samokontroly	nominální hodnota výsledku[1]		aktivace: na vyžádání, nebo cyklická	stav (OK/ chyba)
výpočet ceny při prodeji			jednotková cena	cena
zaokrouhlování				mezivýsledek

Tabulka 2–1: Příklady funkcí, parametrů a dat relevantních z pohledu legální metrologie

[1] mohla by být i specifickým parametrem přístroje

2.4 Ochrana softwaru

2.4.1 Chráněný software

Software (kód programu a příslušná data), jehož změna není buď možná, nebo je detekována a může tak být odhalena (např. porušením plomby nebo prostřednictvím kontrolního čítače).

2.4.2 Kontrolní čítač

Softwarový čítač a/nebo zapisovač, který zaznamenává změny parametrů specifických pro daný přístroj (případně změny ostatních relevantních dat). Může být realizován prostřednictvím následujících prostředků:

- **čítač událostí:** nenulovatelný čítač, který je navýšen pokaždé, kdy přístroj vstoupí do speciálního provozního módu umožňujícího nastavování těchto parametrů a jeden či více parametrů je nastaveno,
- **zapisovač událostí:** soubor zaznamenávající datum a čas vzniku události (změny příslušných parametrů či dat), identifikaci změněných parametrů, novou hodnotu parametru a další potřebné informace.

Části softwaru, které souvisejí s čítáním či zapisováním událostí, jsou v obou případech předmětem kontroly z pohledu legální metrologie a musí být patřičně zabezpečeny.

2.5 Rozhraní

2.5.1 Hardwarové rozhraní

Pojem „rozhraní" zahrnuje jakýkoli prostředek výměny dat mezi měřidlem a jeho okolím včetně jeho realizace (elektrické, mechanické, logické) a interpretaci přenášených dat a instrukcí (ISO 7498) [5].

Hardwarovým rozhraním nazýváme fyzickou část vstupu/výstupu dat či instrukcí do/z přístroje.

2.5.2 Ochranné rozhraní

Rozhraní nazýváme „ochranným", pokud jsou splněny oba následující požadavky:

- přes rozhraní může procházet jen definovaná množina parametrů, dat či instrukcí,

- není možné přes rozhraní předat data či instrukce, které by způsobily:
  – zobrazení dat, která nejsou jasně definována a mohla by být matoucí ve vztahu k výsledkům měření,
  – změnu zobrazených, zpracovávaných či uchovávaných výsledků měření a jiných relevantních dat (např. jednotková cena, celková cena, jednotka měření),
  – neoprávněné nastavení parametrů specifických pro přístroj nebo jeho jiná rekonfigurace,
  – neoprávněnou změnu částí softwaru, které jsou předmětem kontroly z pohledu legální metrologie.

2.5.3 Softwarové rozhraní

Pokud software zahrnuje více částí, z nichž některé jsou předmětem kontroly z pohledu legální metrologie, mohou být odděleny rozhraním na softwarové úrovni. Takové rozhraní pak zajišťuje veškerý přenos dat mezi částmi softwaru, většinou pomocí proměnných či souborů, ke kterým mohou zúčastněné části softwaru přistupovat.

Softwarové rozhraní může být realizované například formou globálních proměnných, parametrů funkcí, nebo souborů s daty.

2.5.4 Ochranné softwarové rozhraní

Rozhraní mezi částí softwaru relevantní z pohledu legální metrologie a jinou částí softwaru nazýváme „ochranným", pokud jsou splněny následující požadavky:

- existuje definovaná množina parametrů, dat a instrukcí, které mohou být přes rozhraní zadávány, čteny, či měněny,
- neexistuje jiný prostředek výměny dat než toto rozhraní.

Proměnné a funkce zodpovědné za funkci ochranného softwarového rozhraní jsou předmětem kontroly z pohledu legální metrologie.

2.6 Bezpečnost dat

autentický program: program, který je ve shodě s programem vyzkoušeným v rámci schválení typu (a této shody si je vědom jak uživatel, tak zákazník). Obyčejně je dodán autorizovanou

organizací, která je odpovědná za tuto shodu NEBO za fakt, že tato shoda může být ověřena.

autentická data: přenášená data, jejichž původ může být příjemcem potvrzen NEBO uložená data, která mohou být jednoznačně přiřazena ke konkrétnímu měření.

metoda autentizace: metoda, kterou můžeme zjistit, zda jsou data nebo software autentické.

Příklad: před posláním dat jsou tato data elektronicky podepsána. Po jejich přijetí je elektronický podpis ověřen.

kontrolní součet: součet všech bytů dat (nejčastěji je použit modulo součet pro výsledek s konstantním počtem znaků).

Kontrolní součet je často používán jako jednoduchý hash algoritmus – tj. algoritmus, který ztrátově komprimuje obsah bloku dat do čísla dané délky takovým způsobem, že změna jakéhokoli bitu v bloku dat vede k jinému výsledku (hash kódu).

elektronický podpis: krátký blok dat přiřazený k přenášeným či uloženým datům za účelem zajištění jejich autenticity a neporušenosti. Je vytvořen algoritmem podpisu a s využitím soukromého klíče. Vytvoření takového podpisu obyčejně probíhá v následujících krocích: 1. hash algoritmus zkomprimuje data, která se mají podepsat, 2. algoritmus podpisu zkombinuje výsledný hash kód se soukromým klíčem, který k podpisu používáme[2].

Elektronický podpis je většinou přidán k datům, která jsou podepisována.

softwarová identifikace: metoda ověření autentičnosti softwaru NEBO řetězec identifikující určitý software (tj. např. jeho verzi)[3].

identifikace softwaru v legální metrologii: softwarová identifikace přiřazená částem software relevantním z pohledu legální metrologie[4].

Jednou z přijatelných metod je tzv. metoda ABC, která se skládá z následujících částí:

- část A je daná výrobcem a na jeho zodpovědnost zahrnuje veškeré změny částí relevantních z pohledu legální metrologie,
- část B je tvořena algoritmem (relevantním z pohledu legální metrologie), který vytváří číslo, jež je automaticky změněno při změně parametrů specifických pro dané měřidlo,

[2] v některých případech je postačující jednoduché řešení kombinující oba body, například CRC součet [11, 12] se skrytou počáteční hodnotou

[3] například číslo verze software

[4] identifikace může mít více částí

- část C je tvořena stejným algoritmem jako B, nicméně výsledek je počítán přes celý software zahrnutý identifikátorem ABC.

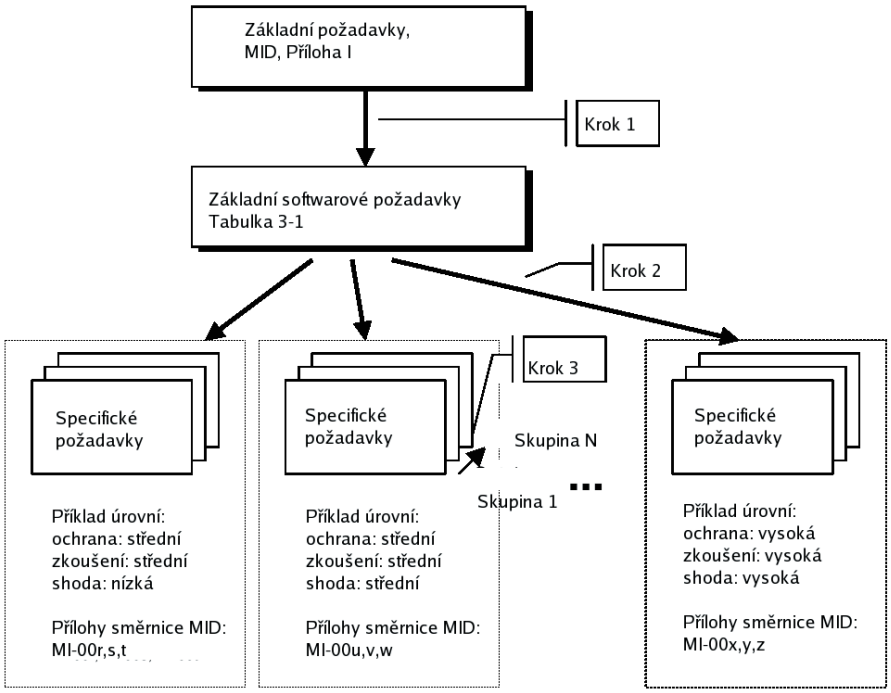
integrita software: software je identický s referenční (schválenou) verzí a nebyl modifikován, ať už úmyslně nebo neúmyslně.

3 ZÁKLADNÍ POŽADAVKY NA SOFTWARE

Důležitá poznámka: *požadavky na software se od vytvoření tohoto dokumentu změnily zásadním způsobem. Zde prezentovaná koncepce by proto neměla být používána pro zhodnocení shody s požadavky dokumentu MID, viz Úvod.*

Základem tohoto dokumentu je směrnice MID [1]. Příloha I této směrnice obsahuje základní požadavky na měřicí přístroje, které zde byly interpretovány ve smyslu požadavků na software v měřicích přístrojích. Výsledkem této interpretace je 5 hlavních oblastí obsahujících 11 *základních požadavků* na software, uvedených v tabulce 3.1. Tyto požadavky jsou velmi obecné a pro praktické použití je nezbytné je dále rozvinout. Na druhé straně existuje velké množství aplikací a možných technických řešení softwaru v měřicí technice. Aby nedocházelo k vytváření velkého množství podrobných technických požadavků souvisejících s jednotlivými aplikacemi (a nepoužitelných pro jiné aplikace), byl zvolen postup aplikování požadavků vytvořených na míru konkrétní aplikaci v několika krocích.

První krok je uveden v této kapitole: jde o aplikaci základních požadavků směrnice MID na software (viz obr. 3.1), další kroky jsou popsány v kapitolách 4 a 5.



krok 1: odvození základních požadavků podle směrnice MID
krok 2: definice skupin měřicích přístrojů, pro které je možné definovat *stejné úrovně* kritérií ochrany, zkoušení a shody
krok 3: interpretace základních požadavků pro jednotlivé skupiny a stanovení specifických požadavků podle *technických parametrů*

Obr. 3.1: rozdělení a interpretace základních požadavků

číslo	základní požadavek[5]	článek směrnice MID[6]
	Design a struktura softwaru	
ER1.1	software v měřicím přístroji musí být navržen tak, aby umožňoval snadnou kontrolu shody s požadavky tohoto dokumentu	AI-12, Článek 10
ER1.2	software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivnitelný jiným softwarem	AI-7.1, 7.2, 7.6, 10.2

číslo	základní požadavek[5]	článek směrnice MID[6]
ER1.3	software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivnitelný rozhraním měřidla	AI-7.1, 8.1
	Ochrana softwaru	
ER2.1	software musí být chráněn proti náhodným nebo neúmyslným změnám	AI-7.1, 7.2, 8.4
ER2.2	software musí být chráněn proti úmyslným změnám neoprávněnými osobami	AI-7.1, 8.2, 8.3, 8.4
ER2.3	pro účely legální metrologie může být použit pouze schválený a ověřený software. Musí být zjevné, že udávané výsledky pocházejí z tohoto softwaru	AI-7.1, 7.2, 7.6, 8.3, 10.2, 10.3, 10.4
ER2.4	chyby ve funkčnosti hardwaru, které by mohly ovlivnit měření, musí být detekovány a ošetřeny	AI-6, MI-001-7.1, MI-002-3.1, MI-004-4
	Shoda softwaru[7]	
ER3.1	software nesmí být neoprávněně měnitelný po jeho schválení	Článek 20, přílohy A až H1
ER3.2	pro ověření shody musí být k dispozici identifikace softwaru a příslušné pokyny	AI-7.6, 8.3
	Testovatelnost	
ER4.1	musí být možné testovat funkčnost přístroje	AI-12
	Dokumentace pro typové schválení	
ER5.1	software musí být i se svým hardwarovým a softwarovým prostředím dostatečně dokumentován	AI-9.3, 12, článek 10

[5] Upozornění: tyto požadavky částečně zahrnují i požadavky na hardware

[6] odkazy na směrnici 2004/22/EC (MID), AI – Příloha I směrnice MID:

článek 10	technická dokumentace
článek 20	neměnné značení
AI-6	spolehlivost
AI-7.1,7.2,7.6	vhodnost
AI-8.1,2,3,4	ochrana před zneužitím
AI-9.3	dokumentace k přístroji
AI-10.2,3,4	indikace výsledku
AI-12	posuzování shody
MI-001-7.1, MI-002-3.1	specifické požadavky pro užitková měřidla
MI-003-4.3.1, MI-004-4	
přílohy A-H1	posouzení shody

[7] Zde je míněna shoda s příslušným vzorem, který byl schválen.

4 DEFINICE ÚROVNÍ

V praxi při schvalování typu bylo zjištěno, že požadavky na jednotlivé typy přístrojů se v některých případech liší i v rámci jednoho státu, a ještě častěji se bez jakýchkoliv objektivních důvodů liší v rámci jednotlivých zemí Evropské unie. Proto byla hledána kritéria, v nichž k takovým rozdílům dochází. Pro každé z kritérií pak byly definovány tři úrovně požadavků s předpokladem, že pro každý konkrétní typ přístroje bude úroveň jednotlivých požadavků pevně dána touto pracovní skupinou, čímž se dosáhne potřebné harmonizace při zkoušení softwaru v rámci schvalování typu.

Kritéria, ve kterých dochází k rozdílnostem, jsou:

- úroveň **ochrany** softwaru proti změnám,
- intenzita **zkoušení** softwaru v rámci schvalování typu,
- úroveň **shody** mezi schváleným softwarem a softwarem v jednotlivých přístrojích.

V této kapitole jsou stanoveny úrovně pro výše uvedená kritéria.

Smyslem a výhodou definování úrovní takových kritérií je fakt, že je možné pomocí nich zajistit srozumitelnou a dobře podloženou interpretaci obecných požadavků na software. Jedná se o druhý krok v obrázku 3.1. V kapitole 6 jsou uvedeny dva příklady aplikace konkrétních úrovní tří výše uvedených kritérií při testování softwareu.

Kromě výše uvedených faktů je nezbytné brát v potaz také technické řešení a vlastnosti daného přístroje. Interpretace požadavků na software vzhledem k technickým vlastnostem měřicího systému je třetím krokem v obrázku 3.1 a bude popsána v kapitole 5. V kapitole 6 bude uveden také příklad klasifikace a zkoušení softwaru vzhledem k těmto aspektům.

Poznámky:

a) kroky 2 a 3 zde nejsou plně dokumentovány – budou upřesněny odborníky na dané měřicí přístroje

b) v současnosti jsou definovány úrovně pouze pro základní softwarové požadavky uvedené v předchozí kapitole; pro další požadavky je třeba je interpretovat obdobným způsobem.

4.1 Úroveň ochrany softwaru

Ochranou softwaru máme na mysli prostředky zamezující náhodné nebo úmyslné modifikaci programu, či dat. Úroveň ochrany je dána použitými technickými prostředky, a je tedy věcí výrobce softwaru. Definice jejích úrovní je odpovědí na následující otázky:

- jak silná musí být ochrana proti modifikaci softwaru?
- jaké nástroje může útočník použít?

Definice úrovní jsou tyto:

nízká: ochrana není zabezpečena žádným zvláštním způsobem

střední: software je zabezpečen proti záměrným změnám prováděným jednoduchými a snadno dostupnými nástroji (např. textovými editory)

vysoká: software je zabezpečen proti záměrným změnám prováděným k tomu určenými sofistikovanými nástroji (diskové editory, vývojové nástroje atd.)

Poznámky:

a) popsané úrovně se týkají záměrných změn. Pro náhodné (neúmyslné) změny je třeba při zkoušení postupovat podle aktuální situace a konfigurace měřidla,

b) výrobce může splňovat požadavky na vyšší úrovni, než předepsané,

c) obvyklá metoda zajištění plombou je analogická softwarové úrovni ochrany na úrovni střední a vysoké.

4.2 Úroveň zkoušení software

Úroveň zkoušení se týká zejména příslušné notifikované osoby a její definice je odpovědí na následující otázky:

- Jaké zdroje musí být použity pro zkoušení?
- Jaké testy budou provedeny?
- Jaká dokumentace musí být předložena?
- Jaké budou důsledky výsledků zkoušení pro žadatele?

Definice úrovní zkoušení je následující:

nízká: zkoušení měřidla je prováděno způsobem standardním pro běžné schválení typu, software není zvláštním způsobem zkoušen,

střední: kromě zkoušení měřidla způsobem standardním pro běž-

né typové schválení je software zkoušen na základě jeho dokumentace. Ta zahrnuje popis softwarových funkcí, popis parametrů atd. Pro prokázání shody s dokumentací jsou prováděny praktické zkoušky vybraných funkcí,
vysoká: kromě zkoušení odpovídajícího střední úrovni je prováděna hloubková kontrola, obyčejně na základě inspekce zdrojového kódu.
Poznámky:
a) úrovně se týkají pouze hloubky zkoušení v softwarové oblasti, ostatní metrologické vlastnosti se zkouší obvyklým způsobem,
b) výrobce a notifikovaná osoba se můžou dohodnout na zkoušení při jakékoli vyšší úrovni, než je předepsána.

4.3 Úroveň shody softwaru

Úroveň shody softwaru a schopnost softwaru prokázat při ověření tuto shodu je záležitostí všech zúčastněných subjektů: výrobce a všech zúčastněných autorit, notifikovanou osobou počínaje.

V průmyslové výrobě se může jevit požadavek na zachování neměnného softwaru v měřicím přístroji v průběhu celého jeho životního cyklu jako problematický. Na jedné straně je zde požadavek na opravy a vylepšování softwau jak z hlediska možných chyb, tak z hlediska vývoje obecných poznatků. Na druhé straně je schválení typu jako takové založeno na předpokladu neměnnosti přístroje a jeho vlastností. Následující definice úrovní shody softwaru si klade za cíl obsáhnout všechny tyto možnosti.

Odpovídáme na níže uvedené otázky:
- Jaké modifikace softwaru jsou možné po schválení, aniž vyžadují další schvalování?
- Jaké modifikace softwaru musí být ohlášeny notifikované osobě?
- Jak bude kontrolována shoda?
- Je třeba uchovat schválenou verzi softwaru?

Definice úrovní shody jsou následující:
nízká: funkce softwaru v každém přístroji se shoduje s dokumentací,
střední: funkce softwaru se shoduje s dokumentací, některé části mohou být navíc při schválení prohlášeny za neměnné, tudíž mo-

hou být nezměnitelné bez souhlasu notifikované osoby. Tyto části musí být v každém přístroji stejné,
vysoká: veškerý software musí být úplně identický se schváleným softwarem.

Poznámka:

výrobce může splňovat požadavky na libovolně vyšší úrovni než předepsané.

5 TECHNICKÉ VLASTNOSTI MĚŘICÍCH PŘÍSTROJŮ A SYSTÉMŮ

Tento dokument je určen pro všechny typy přístrojů, proto jsou požadavky na software uvedené v kapitole 3 poměrně obecné. Pro praktickou aplikaci při schvalování typu měřidla je proto nezbytné interpretovat a konkretizovat je podle softwarové a hardwarové konfigurace daného měřidla.

V kapitole 4 jsou definovány úrovně základních kritérií používaných při zkoušení softwaru, ze kterých musíme v takovém případě vybrat. Na rozdíl od toho pro různé technické vlastnosti přístrojů není nutné takovýmto způsobem definovat jednotné úrovně zkoušení, neboť mohou být objektivně definovány až při zkoušení. V této kapitole jsou proto uvedeny pouze poznámky k některým základním typům požadavků a příklady základní klasifikace hardwarové konfigurace. Příklady konkrétních požadavků při zkoušení jsou uvedeny v kapitole 2 pro dva konkrétní případy hardwarové a softwarové konfigurace přístroje.

Poznámka:

některé zde popsané technické vlastnosti přístrojů nemohou u některých typů přístrojů vůbec nastat (z hlediska legální metrologie). Konkrétní specifikace přijatelných řešení a technických vlastností bude uvedena v dodatcích tohoto dokumentu.

Poznámka:

teoretické množství všech možných hardwarových konfigurací je obrovské. Jen některé z nich jsou prakticky realizovatelné a ve většině konkrétních případů testovaných měřicích přístrojů půjde jen o velmi jednoduchou konfiguraci (viz odstavec 6.1).

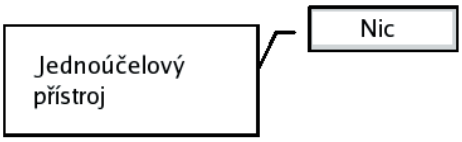
5.1 Hardwarová konfigurace

Pět základních možností hardwarových konfigurací je znázorněno na obrázku 5.1. Zařízení nebo jejich části popsané těmito konfiguracemi mohou být sestrojeny jako jednoúčelová zařízení (většinou typ a–d) nebo s využitím počítače nebo systému počítačů (většinou typ e).

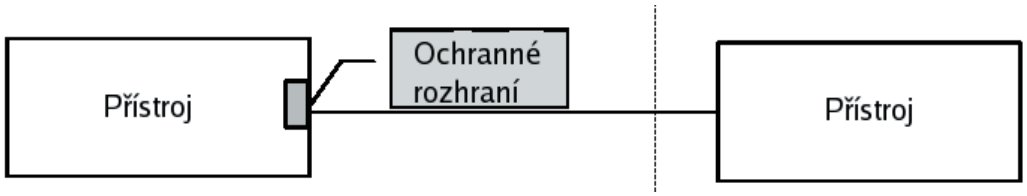
Obr. 5.1: Možné hardwarové konfigurace měřicího přístroje či systému

Části, které jsou předmětem kontroly | části, které nejsou předmětem kontroly

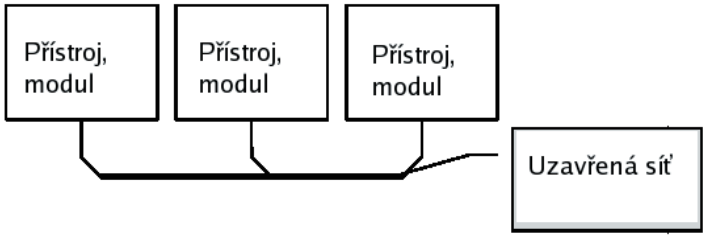
a) jednoduchý jednoúčelový přístroj bez komunikačního rozhraní



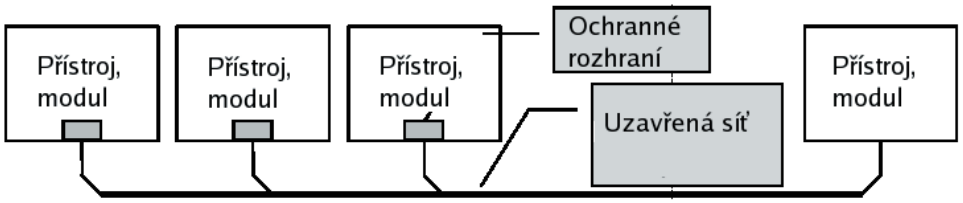
b) přístroj podléhající kontrole s možností připojení k zařízení nepodléhajícímu kontrole



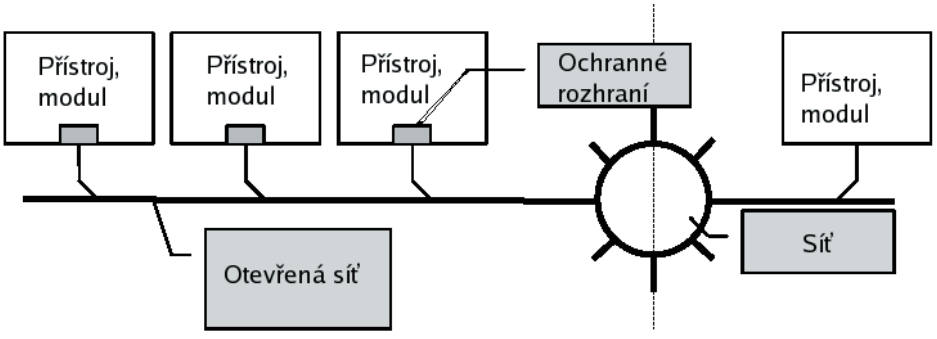
c) modulární systém, všechny přístroje podléhají kontrole, síť je uzavřená



d) modulární systém, některé přístroje nepodléhají kontrole, síť je uzavřená



e) modulární systém, některé přístroje nepodléhají kontrole, síť je otevřená



5.2 Uživatelské rozhraní

Uživatelské rozhraní se skládá ze vstupních a výstupních zařízení (klávesnice, myš, monitor, tiskárna atd.)

f) Uživatelské rozhraní je stále v režimu podléhajícím kontrole z pohledu legální metrologie.

g) Uživatelské rozhraní může být přepnuto z režimu měření do jiného režimu a zpět (uživatel může například pozastavit měření, spustit textový procesor a po jeho ukončení opět spustit měření).

h) Oba režimy mohou běžet paralelně a uživatel může mezi nimi přepínat i v průběhu měření (měřicí program je například okno v grafickém rozhraní operačního systému).

5.3 Nahrávání softwaru

i) Program nemůže být nahráván ani měněn (je zaznamenán na energeticky nezávislém paměťovém médiu, které nemůže být vyměněno).

j) Výrobce zajistí, že programy podléhající kontrole jsou neměnné, zatímco ostatní programy mohou být měněny, například pomocí výměnných médií nebo stahováním ze sítě či jiného úložného média.

k) Jakýkoliv program může být měněn, například pomocí výměnných médií nebo stahováním ze sítě či jiného úložného média.

5.4 Struktura softwaru

l) Software je předmětem kontroly z pohledu legální metrologie jako celek a nemůže být po schválení typu měněn.

m) Části softwaru jsou předmětem kontroly z pohledu legální metrologie, další části mohou být měněny i po schválení typu.
Viz obrázek 2.1 a 2.2

5.5 Prostředí

o) Softwarové prostředí je neměnné, celý přístroj a jeho software byl zhotoven pro účely měření.

p) Měřicí software je součástí běžného operačního systému.

5.6 Detekce chyb

q) Výskyt chyb je zřetelný, chyby mohou být jednoznačně detekovány nebo jsou detekovány hardwarově.
r) Výskyt chyb nemůže být jednoduše detekován ani není detekován hardwarově.

5.7 Dlouhodobé uchovávání naměřených hodnot

s) Měřidlo není vybaveno funkcí dlouhodobého uchovávání dat.
t) Data jsou v měřidle uchovávána pro další zpracování související s funkcí v oboru legální metrologie.

5.8 Měřicí princip

5.8.1 Časová závislost

u) kumulativní měření
v) jednotlivá nezávislá měření

5.8.2 Opakovatelnost

w) opakovatelná měření
x) neopakovatelná měření

5.8.3 Složitost

y) jednoduchá nebo statická měření
z) složitá nebo dynamická měření

6 INTERPRETACE ZÁKLADNÍCH POŽADAVKŮ PRO VYBRANÁ MĚŘIDLA A SYSTÉMY

Koncepce tohoto dokumentu počítá s následujícím postupem při interpretaci obecných požadavků pro konkrétní měřidlo. V první řadě určíme úrovně ochrany, zkoušení a shody podle typu přístroje a jeho použití a zkoušíme, zda přístroj vyhovuje požadavkům pro tyto zvolené úrovně. Poté aplikujeme další požadavky vyplývající z technických vlastností a konstrukce měřidla a softwaru.

Jako příklad konkretizace základních požadavků (essential requirements – **ER**) pro konkrétní hardwarovou a softwarovou konfiguraci měřidla jsou v této kapitole uvedeny dva příklady takové konfigurace a z ní plynoucích požadavků. Tato kapitola by neměla nahrazovat ucelenou sadu příloh s požadavky pro jednotlivá měřidla, ale měla by být příkladem (pokrývajícím poměrně široké spektrum problémů), jak je možné tyto požadavky vytvářet a interpretovat.

V této kapitole je software klasifikován podle kapitoly 4 a 5 a jeho klasifikace proto zahrnuje jak obecné požadavky (ochrana, zkoušení a shoda), tak technické požadavky plynoucí z konstrukce měřidla či softwaru. Příklady jsou následující:

A) jednoduchý jednoúčelový kompaktní měřicí přístroj se všemi komponentami v jednom krytu

B) složitý měřicí systém založený na využití PC a více komponentů propojených sítí.

Poznámka:

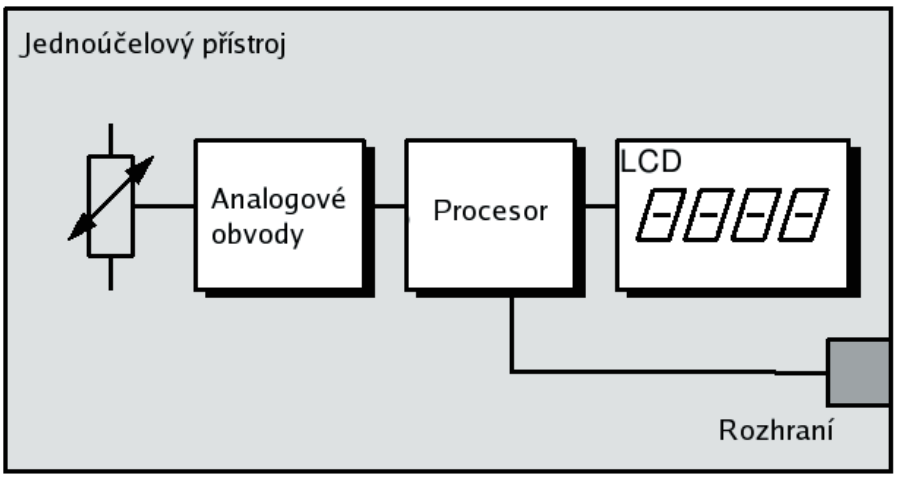
zde uvedeným přístrojům nejsou přiřazeny konkrétní úrovně obecných kritérií (ochrany, zkoušení a shody), a proto jsou vždy uvedeny požadavky plynoucí ze všech jednotlivých úrovní kritérií. Tento přístup by měl usnadnit aplikaci jednotlivých požadavků v praxi (kdy budou konkrétní úrovně daným způsobem stanoveny).

6.1 Příklad A: Jednoduchý jednoúčelový měřicí přístroj

6.1.1 Popis přístroje

Jako příklad jsme zvolili jednoduchý jednoúčelový měřicí přístroj, který může být popsán následujícími technickými parametry (viz obr. 6.1):

- uzavřený kryt: všechny komponenty jsou v jednom krytu, který může být zaplombován,
- přístroj se skládá ze senzorů (převodníků a analogové elektroniky), dalších analogových prvků (např. AD převodník), desky s mikroprocesorem a displeje,
- přístroj má hardwarové rozhraní na připojení zařízení, které není předmětem kontroly z pohledu legální metrologie,
- software je uložen v energeticky nezávislé paměti (pevně zabudovaná Flash, ROM, EEPROM, EPROM nebo PROM),
- celý software je po schválení neměnný a není v něm provedeno oddělení částí relevantních a nerelevantních z pohledu legální metrologie,
- detekce chyb: výpočet kontrolního součtu přes celý obsah paměti.



Obr. 6.1: Hardwarové schéma přístroje v příkladu A

6.1.2 Klasifikace z pohledu legální metrologie

V budoucnosti se klasifikace bude provádět např. podle tabulky A1 v příslušné příloze 1 (odpovídající danému typu přístroje). Vzhledem k tomu, že úrovně zkoušení, ochrany a shody pro jednotlivé typy přístrojů dosud nebyly stanoveny, uvádíme interpretaci základních požadavků pro všechny možné úrovně.

Poznámka:

aby bylo možné porovnat požadavky s příkladem B, byly vybrány následující úrovně (odpovídající kategorii „měřicí přístroje v obchodním styku"):

úroveň ochrany: **střední**

úroveň zkoušení: **střední**

úroveň shody: **nízká**

Tyto úrovně a z nich plynoucí výsledky jsou v textu vyznačeny šedým pozadím.

6.1.3 Technická klasifikace

Podle kapitoly 5 může být přístroj klasifikován následujícím způsobem:

vlastnost	*označení*	*vysvětlivka*
hardwarová konfigurace	b	přístroj je předmětem kontroly a je možné ho připojit k přístroji, který není předmětem kontroly
uživatelské rozhraní	f	přístroj je stále v měřicím režimu
stahování softwaru	i	programy jsou neměnné, není možné je stahovat
struktura softwaru	l	program je schválen jako celek a po schválení není možné jej měnit
softwarové prostředí	o	softwarové prostředí je neměnné, celý přístroj je navržen a konstruován výhradně pro měřicí činnost
detekce chyb	r	chyby nemohou být jednoduše detekovány ani nejsou detekovány hardwarově
dlouhodobé uchovávání	s	systém dlouhodobě neuchovává data
měřicí princip	v, w, y	jednotlivá opakovatelná jednoduchá statická měření

V následujících odstavcích jsou jednotlivé případy popisovány jejich označením, např. (b).

6.1.4 Interpretace základních požadavků

ER1.1: Software v měřicím přístroji musí být navržen tak, aby umožňoval snadnou kontrolu shody s požadavky tohoto dokumentu.

V tomto příkladě se jedná o jednoduchý jednoúčelový měřicí přístroj, jehož software nebude po schválení modifikován. Struktura softwaru v tomto případě není pro zkoušení podstatná a požadavek ER1.1 nevyžaduje další interpretaci.

Poznámka k úrovni zkoušení:

Vzhledem k tomu, že struktura softwaru není pro zkoušení podstatná, není třeba pro její kontrolu stanovit žádné úrovně zkoušení.

ER1.2: Software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivnitelný jiným softwarem (který není předmětem kontroly z pohledu legální metrologie).

Tento požadavek je splněn díky tomu, že v přístroji není žádný další software (l,o)[o] a software nemůže být po schválení typu měněn (i)[i].

Poznámka k úrovni zkoušení:

Vzhledem k tomu, že v přístroji není další software (který by nebyl předmětem kontroly z pohledu legální metrologie), není při zkoušení nutné posuzovat nějakým zvláštním způsobem možnost ovlivnění jiným softwarem, a to ani pro vyšší úrovně zkoušení.

Poznámka k úrovni shody:

Struktura softwaru má vliv na zachování shody v průběhu životního cyklu softwaru. Pro jednoduchou konfiguraci jako v tomto příkladě však není nutné požadavek ER1.2 dále interpretovat.

ER1.3: Software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivněn rozhraním měřidla.

Měřidlo uvedené v tomto příkladě má rozhraní (b)[b], ke kterému může být připojeno zařízení nepodléhající kontrole z pohledu legální metrologie. Pokud je možné prokázat, že se jedná o ochranné rozhraní, není nutné ho zabezpečovat úřední značkou.

Poznámka k úrovni ochrany:

***Nízká**: rozhraní nemusí bát zabezpečeno úřední značkou, ani když se nejedná o ochranné rozhraní.*

Poznámka k úrovni zkoušení:

***Nízká:** Výrobce deklaruje, že se jedná o ochranné rozhraní a že měřené hodnoty ani měřicí funkce nemohou být ovlivněny příkazy nebo daty předávanými přes rozhraní. Není nutné toto stanovisko prověřovat.*

***Střední:** Výrobce v dokumentaci uvede kompletní popis všech příkazů a parametrů, které mohou být předávány přes ochranné rozhraní, a deklaruje, že seznam je kompletní.*

Zkoušení je prováděno na základě dokumentace a zkouší se, zda nemohou být měřená data či měřicí funkce nedovoleně ovlivněny daty procházejícími přes ochranné rozhraní.

***Vysoká:** Zkoušení je prováděno na základě zdrojového kódu softwaru a zkouší se, zda nemohou být měřená data či měřicí funkce nedovoleně ovlivněny daty procházejícími přes ochranné rozhraní.*

---

[o] softwarové prosředí je neměnné a celý přístroj je postaven pro účely měření

[i] není možné nahrávání softwaru, software je umístěn na energeticky nezávislém paměťovém médiu

[b] zařízení podléhající kontrole, které může být připojeno k zařízení nepodléhajícímu kontrole

ER2.1: Software musí být chráněn proti náhodným nebo neúmyslným změnám

Existují dva zdroje neúmyslných změn: fyzické vlivy a nesprávné zacházení ze strany uživatele. Pokud je celý přístroj zkoušen v souladu s příslušnými předpisy (EMC, teplotní vlivy, vlhkost atd.), není nutné zvláštním způsobem kontrolovat vliv fyzických vlivů na chod programu. Pravděpodobnost neúmyslných změn vlivem špatného zacházení je minimalizována tím, že je přístroj neustále v měřicím režimu a že se v něm nenachází software jiný než schválený (f, l, o)[f]. Neúmyslné změny by tak mohly být pouze důsledkem nesprávného chování softwaru, který je zkoušen (nesmí být proto například možné změnit neúmyslně parametry přístroje).

Poznámky k úrovni ochrany:

Úroveň ochrany je stejná, jako u úmyslných změn (viz ER2.2).

Poznámky k úrovni zkoušení:

Nízká: *zacházení s přístrojem je prakticky vyzkoušeno s využitím manuálu k přístroji*

Střední: *kromě praktických zkoušek je zacházení s přístrojem a jeho správné chování testováno i na základě dokumentace (manuálu a dodatečné dokumentace k softwaru).*

Vysoká: *zkouška je prováděna navíc na základě kontroly zdrojového kódu; kontrolujeme, zda není možné nesprávným zacházením docílit chybného chování programu.*

ER2.2: Software musí být chráněn proti úmyslným změnám neoprávněnými osobami

Pravděpodobnost úmyslných změn je minimalizována tím, že je přístroj neustále v měřicím režimu a že se v něm nenachází software jiný než schválený (f, l, o). Úmyslné změny by tak mohly být pouze důsledkem nesprávného chování softwaru, který je zkoušen (nesmí být proto například možné změnit úmyslně parametry přístroje neoprávněnou osobou).

Poznámky k úrovni ochrany:

Nízká: *není zapotřebí zvláštní ochrana.*

Střední/vysoká: software musí být hardwarově či softwarově zabezpečen proti změnám.

[f] uživatelské rozhraní je vždy v režimu podléhajícímu kontrole

Poznámky k úrovni zkoušení:

Nízká: *všechny operace musí být vyzkoušeny na základě manuálu k programu s ohledem na zabezpečení parametrů a programu.*
Střední: všechny ochranné prostředky zmíněné v dokumentaci musí být navíc testovány také prakticky.
Vysoká: *navíc je testováno uživatelské rozhraní, aby se potvrdilo, že existuje pouze omezená množina povolených příkazů.*

ER2.3: Pro účely legální metrologie může být použit pouze schválený a ověřený software. Musí být zjevné, že udávané výsledky pocházejí z tohoto softwaru.

V tomto příkladu se jedná o jednoúčelový přístroj, který je jako celek určen pro měření a jako celek se také posuzuje. Proto výsledky měření a další funkce mohou být označeny jako legálně relevantní prostřednictvím plomb a úředních značek.

Poznámky k úrovni ochrany:

Střední/vysoká: program a paměť musí být chráněny proti vyjmutí z přístroje.

Poznámky k úrovni shody:

Nízká: *výrobce může upravit software, aniž přitom změní jeho identifikaci, musí však informovat notifikovanou osobu, pokud dochází ke změnám částí relevantních z pohledu legální metrologie. Při ověření se kontroluje shoda identifikace softwaru se schváleným softwarem.*
Střední: *vzhledem k tomu, že jde o jednoduchá měření, požadavky jsou stejné jako pro nižší úroveň.*
Vysoká: *v každém přístroji je software identický se schváleným softwarem. Při ověření se kontroluje, zda je software identický. Úřední značka poskytuje uživateli záruku, že prezentované výsledky měření jsou generovány schváleným softwarem.*

ER2.4: Chyby ve funkčnosti hardwaru, které by mohly ovlivnit měření, musí být detekovány a ošetřeny.

V příkladu A jsou některé chyby detekovány a software zajišťuje patřičnou reakci (r)[r].

Poznámky k úrovni zkoušení:

Nízká: *přístroj je zkoušen na základě jeho manuálu. Vzhledem k to-*

[r] přítomnost chyby není jednoznačně pozorovatelná, nemůže být jednoduše zjištěna přístroji nezávislými na měřidle a není hardwarově detekována

mu, že chyby funkčnosti nastávají sporadicky, nejsou zvláštním způsobem testovány.
Střední: *mechanismus ošetření chyb popsaný v dokumentaci je testován vyvoláním příslušných chyb.*
Vysoká: *testování chyb je prováděno jako u střední úrovně. Navíc jsou vyvolány další chyby (nepopsané v dokumentaci) a kontroluje se, jak se přístroj zachová.*

ER3.1: Software nesmí být neoprávněně měněn po jeho schválení.

Povolené změny po schválení software jsou dány příslušnou úrovní shody:

Poznámky k úrovni shody:
Nízká: *software v měřicím přístroji je ve shodě se schválenou dokumentací. I přes drobné změny zdrojového kódu zůstává funkčnost softwaru ve shodě s dokumentací:*
- *modifikace je možné provádět, pokud nezmění funkčnost (uvedenou ve schválené dokumentaci) a pokud je zároveň informována notifikovaná osoba. Změny ve funkčnosti vyžadují nové schválení,*
- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu,*
- *notifikovaná osoba uchovává schválenou dokumentaci. Ve výjimečných případech může uchovávat i spustitelný soubor.*

Střední: *pro zde uvedený příklad jednoduchého jednoúčelového přístroje jsou požadavky stejné jako pro nižší úroveň.*
Vysoká: *celý software je identický se schválenou verzí:*
- *jakákoliv modifikace vede k novému schválení (resp. k dodatku schválení),*
- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu,*
- *notifikovaná osoba uchovává spustitelný soubor (soubory) odpovídající celému softwaru.*

ER3.2: Pro ověření shody musí být k dispozici identifikace softwaru a příslušné pokyny.

Požadavky jsou dány příslušnou úrovní shody:

Poznámky k úrovni shody:

Nízká: *software v měřicím přístroji je ve shodě se schválenou dokumentací. I přes drobné změny zdrojového kódu zůstává funkčnost softwaru ve shodě s dokumentací:*

- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu. Identifikace může být zobrazena stále nebo na vyžádání uživatelem.*

Střední: *pro zde uvedený příklad jednoduchého jednoúčelového přístroje jsou požadavky stejné jako pro nižší úroveň.*

Vysoká: celý software je identický se schválenou verzí:

- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu.*

ER4.1: Musí být možné testovat funkčnost přístroje.

Vzhledem k tomu, že se při schválení typu přístroje provádí standardní metrologické testy, je tento požadavek pro funkce související s měřením splněn automaticky.

Poznámky k úrovni zkoušení:

Nízká: *při metrologických zkouškách jsou zkoušeny pouze části softwaru související s měřicí funkcí. Ostatní funkce nejsou pokryty a ani nemusí být testovatelné, výrobce ovšem musí deklarovat, že jsou tyto funkce ve shodě s příslušnými požadavky (ochranné rozhraní, detekce chyb atd.)*

Střední: *čistě softwarové funkce jsou zkoušeny na základě dokumentace, zároveň se kontroluje, zda jsou dokumentované funkce kompletní a konzistentní.*

Vysoká: *využívá se inspekce zdrojového kódu. Kromě metrologických zkoušek, které jsou i v tomto případě velmi efektivní, jsou zkoušeny také ostatní částí softwaru, a to kontrolou zdrojového kódu nebo jeho analýzou speciálními nástroji. Kontrolují se aspekty, jako jsou ochranné rozhraní, oddělení částí softwaru atd.*

ER5.1: Software musí být i se svým hardwarovým a softwarovým prostředím dostatečně dokumentován.

Pro jednoúčelový přístroj uvedený v tomto příkladu musí být poskytnuta alespoň následující dokumentace:

Poznámky k úrovni zkoušení:

Nízká: *postačuje manuál k přístroji a technická dokumentace. Není nutné poskytovat speciální dokumentaci k softwaru, nicméně poskytnutá dokumentace musí zahrnovat identifikaci softwaru a příslušné deklarace výrobce (např. deklaraci ochranného rozhraní).*

Střední: *navíc je nutné poskytnout následující dokumentaci:*

- *podrobný popis všech parametrů a funkcí relevantních z pohledu legální metrologie,*
- *popis měřicích algoritmů (např. výpočet ceny, zaokrouhlování),*
- *identifikaci softwaru,*
- *kompletní popis všech parametrů a příkazů, které je možné zadat přes ochranné rozhraní, spolu s deklarací jeho kompletnosti,*
- *odkaz na příslušné požadavky tohoto dokumentu,*
- *manuál k přístroji.*

Vysoká: *navíc je nutné poskytnout zdrojový kód programu spolu s následující dokumentací:*

- *logický diagram softwaru (např. diagram toku dat, nebo Nassi Shneirdenmannův diagram),*
- *podrobný popis funkcí každého modulu, který je relevantní z pohledu legální metrologie,*
- *popis přenášených datových struktur.*

6.2 Příklad B: Složitý měřicí systém využívající PC

Složitý systém popsaný v tomto příkladu může být použit v aplikacích například v automatických železničních vážicích mostech, měřicí přístroje rozměrů v kombinaci s vážicími systémy, systémy předávacích bodů atd.

6.2.1 Popis přístroje

Jako příklad jsme zvolili systém skládající se z více komponent (modulů) propojených otevřenou sítí. Má následující vlastnosti:

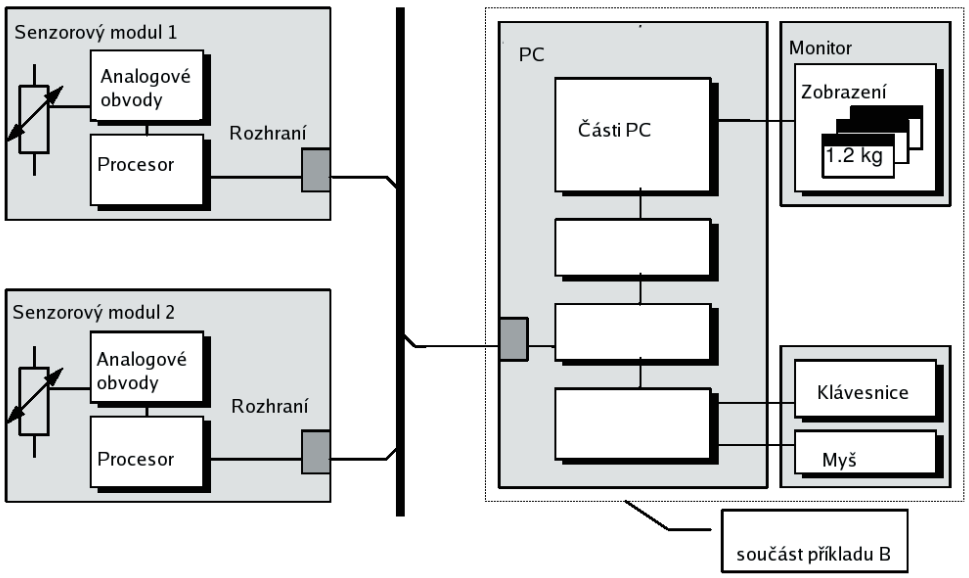
- „senzorové moduly" jsou vybaveny senzory, analogovou a digitální elektronikou (převodník, mikroprocesor) a digitálním rozhraním jsou připojeny do sítě. Nemají však vlastní indikaci.
- Jádrem systému, „centrální jednotkou", je osobní počítač, který je zároveň využit k zobrazování a měřených a uchovávaných hodnot.

- Každý senzorový modul posílá po síti data do centrální jednotky a po síti z ní přijímá povely.
- Centrální jednotka uchovává data pro účely legální metrologie.
- Centrální jednotka je vybavena grafickým rozhraním operačního systému.
- Funkce relevantní z pohledu legální metrologie jsou uchovány ve spustitelném souboru nebo knihovně, která je uložena na pevném disku počítače[8].
- Relevantní software v centrální jednotce získává data ze senzorů, zobrazuje je v okně, ukládá je pro další použití a exportuje je do dalších programů, které nejsou předmětem legální kontroly.
- Proces měření může být dynamický a složitý (např. automatické kolejové vážící mosty, při měření rozměrů a statickém vážení je proces jednoduchý a statický).

V následujícím textu se zabýváme pouze centrální jednotkou (jednotlivé senzory mohou být posuzovány podle příkladu A).

Poznámka:

pokud by se jednalo o „vysokou„ úroveň shody, nebo ochrany, zde popsaný přístroj by technicky nevyhověl požadavkům.



Obr. 6.2: Hardwarové schéma přístroje v příkladu B

[8] dynamická knihovna je soubor funkcí nebo tříd, která může být použita jiným programem. Může být vytvořena nezávisle a jeji vnitřní implementace je skryta.

6.2.2 Klasifikace z pohledu legální metrologie

V budoucnosti se klasifikace bude provádět např. podle tabulky A1 v příslušné příloze 1 (odpovídající danému druhu měřicího přístroje). Vzhledem k tomu, že úrovně zkoušení, ochrany a shody pro jednotlivé typy přístrojů dosud nebyly stanoveny, uvádíme interpretaci základních požadavků pro všechny možné úrovně.

Poznámka:

přístroj, který popisujeme v příkladu B, je poměrně složitý a je pro něj nezbytné interpretovat velké množství základních požadavků. Pro názornost byly vybrány následující úrovně (odpovídající kategorii „měřicí přístroje v obchodním styku"):
úroveň ochrany: ***střední***
úroveň zkoušení: ***střední***
úroveň shody: ***nízká***
Tyto úrovně a z nich plynoucí výsledky jsou v textu vyznačeny šedým pozadím. Jsou v souladu s dokumentem WELMEC 2.3, kde jsou již aplikovány pro vážicí zařízení.

6.2.3 Technická klasifikace

Podle kapitoly 5 může být přístroj (zde pouze centrální jednotka) klasifikován následujícím způsobem:

vlastnost	označení	vysvětlivka
hardwarová konfigurace	e	modulární systém, některé moduly podléhají kontrole, otevřená síť
uživatelské rozhraní	h	režimy podléhající a nepodléhající kontrole mohou být spuštěny paralelně, uživatel má přístup k příkazovému řádku
stahování softwaru	k	může být stahován jakýkoli program, prostřednictvím záznamových médií nebo sítě
struktura softwaru	m	některé části softwaru jsou předmětem kontroly, některé ne – ty mohou být měněny i po schválení typu
softwarové prostředí	p	standardní operační systém, není určen výhradně pro měření
detekce chyb	r	neexistují hardwarové prostředky detekce chyb a chyby nemohou být detekovány jednoduše
dlouhodobé uchovávání	t	měřená data jsou uchovávána pro další využití

vlastnost	označení	vysvětlivka
měřicí princip	v, x, z	jednotlivá neopakovatelná složitá měření

V následujících odstavcích jsou jednotlivé případy popisovány jejich označením, např. (k).

6.2.4 Interpretace základních požadavků

Interpretace se týká pouze centrální jednotky (osobního počítače).

ER1.1: Software v měřicím přístroji musí být navržen tak, aby umožňoval snadnou kontrolu shody s požadavky tohoto dokumentu.

V tomto příkladu výrobce předpokládá, že bude možné měnit (např. upgradovat) části softwaru, které nejsou pod kontrolou z pohledu legální metrologie (m). Je proto nutné provést oddělení částí software tak, aby jedna jeho část byla tvořena všemi funkcemi a daty souvisejícími s legální metrologií a druhá část byla tvořena ostatními funkcemi a daty. Druhá část pak může být modifikována i po ověření softwaru.

Poznámka k úrovni zkoušení:

Nízká: *design a struktura softwaru nemohou být prověřeny běžnými metrologickými zkouškami. Výrobce proto deklaruje, že všechny požadované funkce jsou naprogramovány správně (existence oddělení softwaru, ochranné rozhraní, neměnnost části relevantní z pohledu legální metrologie). Výrobce nemusí poskytovat speciální dokumentaci k těmto funkcím a pravdivost této deklarace není nikterak kontrolována při zkoušení softwaru.*

Střední: *design a struktura softwaru (existence oddělení softwaru, ochranné rozhraní atd.) jsou zkoušeny na základě popisu dodaného výrobcem. Kontroluje se, zda dokumentace obsahuje popis všech relevantních funkcí a zda jsou definovány správně a konzistentně.*

Vysoká: *design a struktura softwaru (existence oddělení softwaru, ochranné rozhraní atd.) jsou navíc zkoušeny na základě zdrojového kódu.*

Poznámka k úrovni shody:

Nízká: *Výrobce* ***deklaruje****, že software implementovaný v každém přístroji je a bude ve shodě s poskytnutou dokumentací. Oddělení*

softwaru musí být ve všech dalších verzích vyvíjených v budoucnosti zachováno.
Střední: *výrobce musí zajistit, že část relevantní z pohledu legální metrologie bude v každém přístroji identická s částí schválenou. Pokud není porušeno oddělení softwaru, může být druhá část modifikována, aniž je o tom zpravena notifikovaná osoba. Výrobce deklaruje, že oddělení zůstane zachováno ve všech budoucích verzích ve stejné podobě, a pokud by tomu tak nebylo, musí notifikovanou osobu informovat.*
Dokumentaci schváleného softwaru a jeho kód (spustitelný soubor) uchovává notifikovaná osoba.
Vysoká: *žádná část softwaru nesmí být modifikována. Je tedy nepřípustné, aby měl software zařazený do této úrovně strukturu uvedenou v tomto příkladu.*

ER1.2: Software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivnitelný jiným softwarem (který není předmětem kontroly z pohledu legální metrologie).

Aby bylo možné předávat data mezi jednotlivými částmi softwaru (podléhající a nepodléhající kontrole), je nezbytné implementovat softwarové ochranné rozhraní. Takové rozhraní má dvě funkce:
• interakci mezi částmi softwaru,
• tok dat mezi částmi softwaru.

V tomto příkladě je část, která je předmětem kontroly z pohledu legální metrologie, zkompilována do knihovny a ostatní programy mohou volat funkce exportované z této knihovny, a tím způsobem získávat data nebo zadávat příkazy.

Část softwaru, která je předmětem kontroly z pohledu legální metrologie, nesmí být ovlivněna okolním prostředím. V tomto případě, kdy je možné do počítače volně nahrávat programy a data a v počítači paralelně spouštět programy ve víceúlohovém operačním systému (p), musíme do tohoto prostředí zahrnout vše, co by potenciálně mohlo být na počítači spuštěno. V tomto příkladě se za ochranu funkcí relevantních z pohledu legální metrologie považují prostředky operačního systému.

Obdobně jako u příkladu A musí být aplikovány prostředky pro ochranu softwaru před neoprávněným vlivem uživatele (viz též ER2.2), zde navíc se zahrnutím ochrany před tvůrci ostatního softwaru v počítači; navíc musí být aplikovány další mechanismy na ochranu prezentovaných hodnot (viz ER2.3).

Poznámka k úrovni ochrany (ER 1.2):

Nízká: *není vyžadována žádná zvláštní ochrana.*

Střední: *viz ER 2.2.*

Vysoká: *technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

Poznámka k úrovni zkoušení (ER 1.2):

Nízká: *design a struktura softwaru nemůže být prověřena běžnými metrologickými zkouškami. Výrobce proto deklaruje, že všechny požadované funkce jsou naprogramovány správně (existence oddělení softwaru, ochranné rozhraní, neměnnost části relevantní z pohledu legální metrologie). Výrobce nemusí poskytovat speciální dokumentaci k těmto funkcím a pravdivost této deklarace není nikterak kontrolována při zkoušení softwaru.*

Střední: *softwarové rozhraní je zkoušeno na základě dokumentace dodané výrobcem. Kontroluje se:*

- *zda se jedná o ochranné rozhraní, tj. neexistuje jiný způsob předávání příkazů a dat a existuje uzavřená množina dokumentovaných příkazů, které mohou být pomocí rozhraní předávány,*
- *zda není možné rozhraní obejít (kompilace funkcí do samostatné knihovny spolu s dokumentací k exportovaným funkcím je dostačující),*
- *zda je operační systém schopen ochránit software před vlivy okolního prostředí (např. víceúlohový operační systém, viz též ER2.2).*

Vysoká: *navíc se provádí kontrola na základě zdrojového kódu.*

Poznámka k úrovni shody (ER 1.2):

Nízká: *výrobce deklaruje, že rozhraní je ochranné a že je instalováno stejným způsobem u každého přístroje.*

Výrobce dodává dokumentaci, ve které jsou popsány funkce ochranného rozhraní a pravidla pro jeho správné použití.

Pokud dojde ke změně softwaru souvisejícího s ochranným rozhraním, musí výrobce informovat notifikovanou osobu.

Notifikovaná osoba uchovává po schválení typu dokumentaci k soft-

warovému rozhraní, výjimečně i kód programu ve formě spustitelného souboru.

Střední: *software, který vytváří ochranné rozhraní, musí být v každém přístroji identický se schváleným softwarem.*
Výrobce dodává dokumentaci, ve které jsou popsány funkce ochranného rozhraní a pravidla pro jeho správné použití.
Výrobce nebo uživatel může modifikovat software, který není předmětem kontroly z pohledu legální metrologie, pokud při tom nedojde k narušení či změně ochranného rozhraní. Pokud dojde ke změně softwaru souvisejícího s ochranným rozhraním, musí výrobce informovat notifikovanou osobu.
Notifikovaná osoba uchovává po schválení typu dokumentaci k softwarovému rozhraní a kód programu ve formě spustitelného souboru.
Vysoká: *Žádná část softwaru se nemůže měnit. Technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

ER1.3: Software v měřicím přístroji musí být navržen tak, aby nebyl nedovoleně ovlivnitelný rozhraním měřidla.

Měřicí systém uvedený v tomto příkladě se skládá z několika senzorových modulů, které jsou k centrální jednotce připojeny prostřednictvím otevřené sběrnice. Pro získání měřených hodnot je tedy nezbytné využít rozhraní měřidla a komunikace mezi jednotlivými částmi systému. Existuje několik způsobů, jakými by mohly být měřené hodnoty ovlivněny:

1. *přes rozhraní by spolu s přenosem dat mohlo dojít k nepřípustnému zadávání příkazů a ovládání centrální jednotky,*
2. *přenášené hodnoty by mohly být ovlivněny v průběhu přenosu po síti,*
3. *přenášené hodnoty by mohly mít jiný původ než v daném senzorovém modulu.*

Bod 2 a 3 je ošetřen v požadavcích na přenos dat (ER2.2) a jejich zabezpečení (ER2.2). Bod 1 souvisí s vlastnostmi rozhraní měřidla a je dále diskutován v tomto odstavci.

Poznámka k úrovni ochrany (ER 1.3):

Nízká: *neexistuje ochrana proti ovlivňování přístroje nebo dat prostřednictvím rozhraní.*

Střední: *rozhraní musí být navrženo a realizováno tak, aby propouštělo pouze příkazy, které nemohou neoprávněně ovlivnit funkce podléhající kontrole z pohledu legální metrologie (viz též ER2.2).*

Vysoká: *stejná jako pro střední úroveň.*

Poznámka k úrovni zkoušení (ER 1.3):

Nízká: *vlastnosti rozhraní nemohou být posouzeny na základě běžných metrologických zkoušek. Výrobce proto deklaruje, že není možné přes rozhraní předávat příkazy, které by neoprávněně ovlivnily funkce podléhající kontrole z pohledu legální metrologie. Tato vlastnost však není žádným zvláštním způsobem zkoušena.*

Střední: *rozhraní je zkoušeno na základě dokumentace, ve které jsou uvedeny:*

- *definice a popis funkcí, které mohou být ovládány prostřednictvím rozhraní,*
- *definice a popis parametrů, které mohou být nastaveny či měněny prostřednictvím rozhraní,*
- *specifikace nastavovaných parametrů a ovládaných funkcí, pokud jsou předmětem kontroly z pohledu legální metrologie.*

Na základě dokumentace je zkoušeno, zda se jedná o ochranné rozhraní, tj. zda není možné prostřednictvím tohoto rozhraní neoprávněně měnit data či spouštět příkazy a zda výrobce deklaruje, že jím popsaná množina povolených příkazů je úplná.

Vysoká: *rozhraní je navíc zkoušeno na základě kontroly zdrojového kódu.*

Poznámka k úrovni shody (ER 1.3):

Nízká: *výrobce deklaruje, že není možné prostřednictvím rozhraní předávat příkazy, které by neoprávněně měnily data či spouštěly funkce, které jsou předmětem kontroly z pohledu legální metrologie.*

Střední/vysoká: *software související s funkcí rozhraní musí být identický se schváleným softwarem a jakékoliv jeho změny musí být dány na vědomí notifikované osobě.*

ER2.1: Software musí být chráněn proti náhodným nebo neúmyslným změnám.

V tomto příkladu můžeme uvažovat o následujících zdrojích neúmyslných změn:

• fyzikální vlivy (teplota, vlhkost, elektromagnetické záření),
• elektromagnetické vlivy na přenosový kanál,
• softwarové problémy, viry,
• neúmyslné editování programu či dat textovým editorem,
• chyby ve zkoušeném softwaru (např. možnost změnit neoprávněně data, která nesmí být měněna).

Poznámky k úrovni ochrany (ER 2.1):

úroveň ochrany je stejná jako u úmyslných změn (viz ER2.2).

Poznámky k úrovni zkoušení (ER 2.1):

Nízká: *výrobce* ***deklaruje****, že existují prostředky detekce neúmyslných změn a přiměřená reakce na tyto změny. Není prováděno zvláštní zkoušení kromě běžných metrologických zkoušek zahrnujících ovládání přístroje přes uživatelské rozhraní a zkoušky fyzikálních vlivů.*

Střední: *prostředky pro detekci neúmyslných změn jsou zkoušeny na základě dokumentace poskytnuté výrobcem. Je zkoušeno, zda*

• je popsán algoritmus detekce (program může například počítat svůj vlastní kontrolní součet a porovnávat jej se známou hodnotou; v případě rozdílu zastaví svůj chod),
• protokol přenosu dat umožňuje detekci neúmyslných změn (pro tyto účely je dostačující, je-li zajištěna ochrana proti úmyslným změnám ve smyslu odstavce ER2.1),
• je uživatelské rozhraní kompletně popsáno.

Vysoká: *navíc je prováděna kontrola softwaru zodpovědného za přesnost dat na základě jeho zdrojového kódu.*

ER2.2: Software musí být chráněn proti úmyslným změnám neoprávněnými osobami.

Ochrana kódu programu

V centrální jednotce je k dispozici příkazový řádek a mohou být nahrány různé softwarové nástroje, například editory.

Poznámky k úrovni ochrany (ER 2.2):

Nízká: *není vyžadována zvláštní ochrana.*

Střední: *kód programu musí být chráněn proti změnám jednoduchými nástroji, například textovými editory. V tomto příkladu program počítá svůj vlastní kontrolní součet a ten porovnává s nominální hodnotou; v případě nesouhlasu zastaví svoji činnost. Předpokládáme, že*

s pomocí textového editoru by bylo velmi obtížné najít v binárním souboru starý kontrolní součet, spočítat nový a zapsat ho zpět.
Běh programu samotného není možné pomocí textového editoru ovlivnit.
Vysoká: *kód programu musí být chráněn proti změnám sofistikovanými nástroji, jako jsou debuggery nebo diskové editory, a to na úrovni srovnatelné s nejvyšším zabezpečením programů používaných například v elektronickém bankovnictví.*

Technické řešení uvedené v tomto příkladě je **neslučitelné** s vysokou úrovní ochrany. Pro její dosažení by bylo nezbytné využít dodatečného hardwaru pro kontrolu integrity programu.

Poznámky k úrovni zkoušení (ER 2.2):
Nízká: *výrobce **deklaruje**, že je program schopen detekovat záměrné změny a reagovat na ně. Není třeba poskytovat pro tyto účely zvláštní dokumentaci a není v tomto ohledu prováděna žádná kontrola kromě praktických testů uživatelského rozhraní v průběhu metrologických zkoušek.*
Střední: *prostředky detekce změn jsou zkoušeny na základě dokumentace poskytnuté výrobcem. Prověřuje se, zda je v dokumentaci popsán algoritmus detekce (program může například počítat svůj vlastní kontrolní součet a porovnávat jej se známou hodnotou; v případě rozdílu zastaví svůj chod).*
Detekce změn je prakticky zkoušena s využitím textového editoru.
Vysoká: *navíc je prováděna kontrola algoritmu detekce a detekce samotné na základě zdrojového kódu.*

Ochrana parametrů specifických pro typ měřidla

Parametry specifické pro typ měřidla jsou většinou součástí kódu programu (pak se na ně vztahují požadavky uvedené v předchozím odstavci), případně jsou od kódu odděleny (pak se na ně vztahují požadavky uvedené v následujícím odstavci).

Ochrana parametrů specifických pro měřidlo

Existuje pouze jediný rozdíl mezi parametry specifickými pro typ měřidla a parametry konkrétního měřidla a ten spočívá pouze v možnosti nastavení parametrů konkrétního měřidla před ověřením měřidla (a nesmí být možné je ze strany uživatele či neoprá-

něného subjektu měnit po jeho ověření). Interpretace požadavků je proto poněkud odlišná od požadavků na kód programu.

Poznámky k úrovni ochrany (ER 2.2):

Nízká: *není zapotřebí zvláštní ochrana.*

Střední/Vysoká: *zabezpečení proti editaci kódu proti jeho změnám jednoduchými, nebo sofistikovanými nástroji musí být doplněno ochranou parametrů, buď hardwarovou nebo softwarovou. V konfiguraci uvedené v tomto příkladu je tyto podmínky možné splnit, jen pokud jsou všechny parametry uloženy v senzorových modulech, které mohou být zaplombovány.*

Poznámky k úrovni zkoušení (ER 2.2):

Nízká: *výrobce* ***deklaruje****, že žádné ze specifických parametrů nejsou uloženy v centrální jednotce. Nejsou prováděny speciální testy kromě kontroly neměnnosti parametrů při standardních metrologických zkouškách.*

Střední: *výrobce v dokumentaci popisuje všechny specifické parametry a jejich umístění a zabezpečení v softwaru. Na základě dokumentace je posouzeno, zda nemohou být neoprávněně měněny. Kromě toho se provádějí testy uživatelského rozhraní.*

Vysoká: *software je navíc zkoušen za základě jeho zdrojového kódu, zejména části související s uchováváním specifických parametrů. Pro jejich ochranu musí být použity hardwarové prostředky.*

Ochrana proti obcházení softwarového rozhraní (ER2.2)

V tomto příkladu je software vybaven ochranným softwarovým rozhraním, jehož obejitím či zneužitím by bylo v principu možné ovlivnit kód, data nebo parametry programu. Požadavky na kód a parametry programu jsou dány v předchozích odstavcích.

Ochrana přenášených dat (ER2.2)

V tomto příkladu jsou data přenášena po síti ze senzorových modulů do centrální jednotky. Data musí být chráněna ze dvou důvodů:

- data by mohla být v průběhu přenosu ovlivněna (porušení integrity),
- data by mohla být jiného původu než z příslušného senzorového modulu (porušení autentičnosti).

Poznámky k úrovni ochrany (ER 2.2):

Nízká: *není zapotřebí zvláštní ochrana.*

Střední: integrita: *přenášená data musí být chráněna proti změnám jednoduchými nástroji, například textovými editory. Toto může být zajištěno například digitálním podepisováním nebo šifrováním dat (viz 2.6).*

Úroveň zabezpečení se odvíjí od použitého algoritmu a délky jeho klíče – pro tuto úroveň je postačující například algoritmus CRC [11, 12] s délkou klíče 2 byty.

autentičnost: *příjemce dat musí být schopen poznat, že data pocházejí ze správného zdroje a jsou aktuální. Přijatelným řešením může být například:*

- *zaznamenání adres všech senzorových modulů, připojení adresy k měřené hodnotě a zpětnou kontrolou adresy po přijetí dat,*
- *připojení časového údaje k měřené hodnotě a jeho kontroly po přijetí dat.*

Všechny údaje, které jsou nezbytné pro ověření autentičnosti, musí být přitom obsahem jednoho datového bloku, který je podepisován nebo šifrován jako celek.

Data, která jsou poškozená, nesmí být dále vyhodnocována.

Klíč použitý pro podepisování nebo ověřování podpisu musí být považován za data podléhající kontrole z pohledu legální metrologie.

Vysoká: *přenášená data musí být chráněna proti změnám sofistikovanými nástroji, jako jsou debuggery nebo diskové editory, a to na úrovni srovnatelné s nejvyšším zabezpečením programů používaných například v elektronickém bankovnictví.*

Požadavky jsou obdobné jako pro střední úroveň s tím, že přijatelným algoritmem pro podepisování je například DEA s minimální délkou klíče 128 bitů.

Technické řešení uvedené v tomto příkladě je ***neslučitelné*** *s vysokou úrovní ochrany. Pro její dosažení by bylo nezbytné využít dodatečného hardwaru pro kontrolu integrity dat.*

Poznámky k úrovni zkoušení (ER 2.2):

Nízká: *výrobce* ***deklaruje****, že je program schopen detekovat změny dat při přenosu a reagovat na ně. Není třeba poskytovat pro tyto účely zvláštní dokumentaci a není v tomto ohledu prováděna žádná kontrola.*

Střední: *prostředky detekce změn jsou zkoušeny na základě dokumentace poskytnuté výrobcem. Prověřuje, se zda*

- *je použit vhodný algoritmus a vhodná délka klíče,*
- *jsou všechna data potřebná pro ověření autentičnosti přenášena jako jeden podpisovaný nebo šifrovaný blok spolu s měřenou hodnotou,*
- *nemůže být klíč nalezen pomocí textového editoru.*

Prakticky se zkouší poslat poškozený datový blok a reakce na tuto situaci.

Vysoká: *navíc se na základě zdrojového kódu zkouší software související s kontrolou přenesených dat.*

Ochrana dlouhodobě uchovávaných dat (ER2.2)

V tomto příkladu jsou v centrální jednotce dlouhodobě uchovávána data pro další použití. Součástí softwaru na centrální jednotce je program, který dokáže starší data zpracovávat a prezentovat uživateli. Pomocí tohoto programu je možné jednoznačně přiřadit dřívější měření konkrétním datům.

Poznámky k úrovni ochrany (ER 2.2):

Nízká: *není zapotřebí zvláštní ochrana*

Střední: integrita: *uložená data musí být chráněna proti změnám jednoduchými nástroji, například textovými editory. Toto může být zajištěno například digitálním podepisováním nebo šifrováním dat (viz 2.6). Úroveň zabezpečení se odvíjí od použitého algoritmu a délky jeho klíče – pro tuto úroveň je postačující například algoritmus CRC [11, 12] s délkou klíče 2 byty.*

autentičnost: při použití dat musí být možné přiřadit data konkrétnímu měření. Přijatelným řešením může být například:

- *přiřazení jedinečného ID k naměřeným hodnotám,*
- *připojení časového údaje k měřené hodnotě.*

Všechny údaje, které jsou nezbytné pro přiřazení, musí být přitom obsahem jednoho datového bloku, který je podepisován nebo šifrován jako celek.

Data, která jsou poškozená, nesmí být dále vyhodnocována.

Klíč použitý pro podepisování nebo ověřování podpisu musí být považován za data podléhající kontrole z pohledu legální metrologie.

Vysoká: *uložená data musí být chráněna proti změnám sofistiko-*

vanými nástroji, jako jsou debuggery nebo diskové editory, a to na úrovni srovnatelné s nejvyšším zabezpečením programů používaných například v elektronickém bankovnictví.
Požadavky jsou obdobné jako pro střední úroveň s tím, že přijatelným algoritmem pro podepisování je například DEA[9] s minimální délkou klíče 128 bitů.
Technické řešení uvedené v tomto příkladě je neslučitelné s vysokou úrovní ochrany. Pro její dosažení by bylo nezbytné využít dodatečného hardwaru pro kontrolu integrity dat.
Poznámky k úrovni zkoušení (ER 2.2):
Nízká: *výrobce* **deklaruje**, *že je program schopen detekovat změny uložených dat a reagovat na ně. Není třeba poskytovat pro tyto účely zvláštní dokumentaci a není v tomto ohledu prováděna žádná kontrola.*
Střední: *prostředky detekce změn jsou zkoušeny na základě dokumentace poskytnuté výrobcem. Prověřuje se, zda*
- *je použit vhodný algoritmus a vhodná délka klíče,*
- *jsou všechna data potřebná pro přiřazení měření uložena jako jeden podpisovaný nebo šifrovaný blok spolu s měřenou hodnotou,*
- *nemůže být klíč nalezen pomocí textového editoru.*

Prakticky se zkouší poškodit datový blok v centrální jednotce (textovým editorem) a reakce na tuto situaci.
Vysoká: *navíc se na základě zdrojového kódu zkouší software související s kontrolou přenesených dat.*

ER2.3: Pro účely legální metrologie může být použit pouze schválený a ověřený software.

Při použití standardního osobního počítače musí být v této souvislosti ošetřeny následující aspekty:
- uživatel nebo výrobce by mohl spustit program, který nebyl schválen[k],
- na počítači by mohl být spuštěn program, který by modifikoval (např. překreslil) okna prezentovaná uživateli, a tím ovlivnil hodnoty čtené uživatelem[hp].

---

9 specifikováno v [10]

k může být nahrán jakýkoli program, například z výměnného média nebo sítě

h uživatelské rozhraní operačního systému v režimu podléhajícím kontrole a v režimu nepodléhajícím kontrole mohou existovat souběžně

p software je vnořen do operačního systému nebo podobného prostředí nevytvořeného přímo pro účely měření

Instalace programu

Výrobce může instalovat jen program, který byl schválen.

Poznámky k úrovni shody (ER 2.3):

Nízká: *výrobce může modifikovat software, aniž přitom změní jeho identifikaci. Pokud se změna týká částí (zde knihoven) relevantních z pohledu legální metrologie, musí být o změně informována notifikovaná osoba. Při ověření je kontrolována shoda identifikace softwaru se schváleným.*

Střední: *část softwaru, která je relevantní z pohledu legální metrologie, musí být identická se schváleným programem. Fakt, že je tato část identická, se zkouší při ověření na základě identifikace softwaru a je vyjádřen označením měřidla úřední značkou.*

Vysoká: *žádná část softwaru se nemůže měnit a celý software musí být identický se schváleným softwarem. Technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

Záměna programu v měřidle po schválení

Uživatel může na osobním počítači, sloužícím v tomto příkladě jako centrální jednotka, spouštět libovolné programy (k).

Poznámky k úrovni ochrany (ER 2.3):

Nízká: *není vyžadována zvláštní ochrana.*

Střední: *nepředpokládá se, že by bylo možné vytvořit program, který by mohl simulovat funkci schváleného programu pomocí jednoduchých nástrojů, jako jsou textové editory. Vzhledem k tomuto faktu a vzhledem k tomu, že taková záměna by mohla být kvalifikována jako trestný čin, nejsou vyžadovány zvláštní prostředky pro zabezpečení spuštění jiného programu.*

Poznámka: *pokud výrobce sám produkuje programy, které by mohly být pro takové zneužití využity, musí zajistit, že nebude možné je na počítač nainstalovat.*

Vysoká: *technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

Poznámky k úrovni shody (ER 2.3):

Nízká/Střední: *uživatel nebo posuzovatel může zkontrolovat identitu programu na základě jeho identifikace a hodnoty uvedené na štítku přístroje.*

Vysoká: *technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

Identifikace a prezentace výsledků

Pokud je na počítači možné spustit více programů současně, uživatel musí stále vidět prezentaci výsledků z programu podléhajícího kontrole z pohledu legální metrologie. Priorita zobrazení oken na obrazovce musí být proto patřičně upravena.

Poznámky k úrovni ochrany (ER 2.3):

Nízká: *není vyžadována zvláštní ochrana.*

Střední: *předpokládáme, že je možné využít textového editoru (a ostatních jednoduchých nástrojů) pro vytvoření zavádějící prezentace výsledků, která by měla být použita namísto okna softwaru, který je předmětem kontroly. Je proto nezbytné takové operaci zabránit, zejména následujícími prostředky:*

- *měřené a zpracovávané hodnoty nejsou exportovány z programu dříve, než jsou zobrazeny (nebo uloženy do dlouhodobé paměti),*
- *program vytváří okno, které je vždy viditelné a nemůže být překryto dalšími okny. Pokud není okno vidět, program nezpracovává data,*
- *grafické uživatelské rozhraní programu vypadá tak, že není možné je nasimulovat jednoduchými prostředky. Navíc je jeho obraz uložen v uživatelském manuálu.*

Poznámka: *předpokládáme, že snaha zde uvedeným způsobem ovlivnit prezentované výsledky měření by byla trestným činem.*

Vysoká: *technické řešení uvedené v tomto příkladu není pro tuto úroveň přípustné.*

Poznámky k úrovni zkoušení (ER 2.3):

Nízká: *výrobce* ***deklaruje****, že jsou použity příslušné prostředky pro zajištění viditelnosti okna s prezentovanými hodnotami a že data nejsou exportována z programu dřív, než jsou zobrazena. Nejsou prováděny zvláštní zkoušky.*

Střední: *Prostředky ochrany prezentace jsou zkoušeny na základě dokumentace, konkrétně se zkouší, zda*

- *nejsou měřené hodnoty exportovány dříve, než jsou zobrazeny výsledky,*
- *je okno s prezentací výsledků vždy viditelné,*
- *je okno navrženo tak, aby nebylo možné je snadno zaměnit s výsledkem použití textového editoru.*

Prakticky je zkoušeno, zda je okno s prezentací výsledků měření vždy viditelné.

Vysoká: *na základě zdrojového kódu je kontrolována část softwaru zodpovědná za obnovování okna a další funkce související s ochranou prezentace výsledků.*

ER2.4: Chyby ve funkčnosti hardwaru, které by mohly ovlivnit měření, musí být detekovány a ošetřeny.

V tomto příkladu předpokládáme, že jsou některé chyby detekovány a ošetřeny.

Poznámky k úrovni zkoušení (ER 2.4):

Nízká: *přístroj je prakticky zkoušen (v rámci metrologických zkoušek), vzhledem k tomu, že chyby ve funkčnosti nastávají ojediněle, nejsou zvláštním způsobem testovány.*

Střední: *simulujeme příslušné chyby a testujeme mechanismy detekce a ošetření chyb popsané v dokumentaci.*

Vysoká: *navíc testujeme i reakci na další možné chyby nepopsané v dokumentaci.*

ER3.1: Software nesmí být neoprávněně měnitelný po jeho schválení.

Možné změny softwaru po schválení závisí na zvolené úrovni shody:

Poznámky k úrovni shody:

Nízká: *software instalovaný v každém přístroji musí být ve shodě se schválenou dokumentací. Funkčnost musí být identická, až na drobné změny a opravy prováděné následujícím způsobem:*

- *části podléhající kontrole je možné měnit, jen jestliže nedojde ke změně funkcí a charakteristik přístroje. Notifikovaná osoba musí být o změnách informována. Pokud dojde ke změně funkcí nebo charakteristik přístroje, musí být provedeno nové schválení,*
- *pokud existuje oddělení softwarových částí a je využíváno ochranné softwarové rozhraní, mohou být části nepodléhající kontrole měněny bez vědomí notifikované osoby,*
- *schválená dokumentace a výjimečně i kód programu jsou uloženy u notifikované osoby.*

Střední: *části podléhající kontrole jsou identické se schváleným softwarem:*

- *jakékoliv modifikace těchto částí vyžadují novou identifikaci a schválení,*
- *pokud existuje oddělení softwarových částí a je využíváno ochranné softwarové rozhraní, mohou být části nepodléhající kontrole měněny bez vědomí notifikované osoby,*
- *schválená dokumentace a kód programu jsou uloženy u notifikované osoby.*

Vysoká: *veškerý software je identický se schváleným softwarem:*

- *jakékoli modifikace softwaru vyžadují novou identifikaci a schválení,*
- *schválená dokumentace a kód programu jsou uloženy u notifikované osoby.*

ER3.2: Pro ověření shody musí být k dispozici identifikace softwaru a příslušné pokyny.

Musí existovat popis jak získat identifikaci software, přičemž požadavky závisí na zvolené úrovni shody:

Poznámky k úrovni shody:

Nízká: *software v měřicím přístroji je ve shodě se schválenou dokumentací. I přes drobné změny zdrojového kódu zůstává funkčnost softwaru ve shodě s dokumentací:*

- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu. Identifikace může být zobrazena stále nebo na vyžádání uživatelem.*

Střední: *části podléhající kontrole jsou identické se schváleným softwarem:*

- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu.*

Vysoká: *celý software je identický se schválenou verzí:*

- *při ověření je shoda se schváleným softwarem kontrolována na základě identifikace uvedené v certifikátu schválení typu.*

ER4.1: Musí být možné testovat funkčnost přístroje.

V tomto příkladu se jedná o složité měření, které nemůže být snadno a často kontrolováno na základě jednoduchých metrologických zkoušek.

Poznámky k úrovni zkoušení:

Nízká: *výrobce poskytne při schválení pro kontrolu sadu vstupních a výstupních hodnot spolu s podmínkami měření a deklarací,*

že byly všechny hodnoty získány se schvalovanou verzí softwaru. Další funkce měřidla, které by nebyly takovou zkouškou pokryty, nemusí být výrobcem zkoušeny, musí však být deklarováno, že splňují příslušné požadavky.

Požadavky na výrobce: výrobce poskytne při schválení pro kontrolu sadu vstupních a výstupních hodnot, spolu s podmínkami měření a deklarací, že byly všechny hodnoty získány se schvalovanou verzí softwaru. Výsledky jsou dokumentované a je deklarována jejich shoda či neshoda se správnými hodnotami.

Střední: *pro kontrolu je použito zařízení simulující vstupy z jednotlivých přístrojů. S výsledky vypočítanými na základě takových simulací se zachází jako s výsledky při běžném měření. Kromě toho jsou provedeny i praktické zkoušky se skutečnými vstupy. Pokud je to nutné, je možné podle průběhu testů požadovat další, speciální testy.*

Požadavky na výrobce: přístroj by měl být vybaven rozhraním, pomocí kterého by bylo možné zadávat/získávat vstupy či výstupy. Pokud tomu tak není, musí takové zařízení na vyžádání dodat.

Vysoká: *nástroji. Typickým příkladem takových částí jsou ochranné rozhraní, oddělení softwaru atd.*

Pro kontrolu je použito zařízení simulující vstupy z jednotlivých přístrojů. S výsledky vypočítanými na základě takových simulací se zachází jako s výsledky při běžném měření. Kromě toho jsou provedeny i praktické zkoušky se skutečnými vstupy, nástroji pro analýzu software. Tímto způsobem se kontroluje například ochranná funkce rozhraní, oddělení software, apod.

Navíc je provedena zkouška na základě zdrojového kódu, zejména kontrola funkcí, které jsou podstatné pro funkčnost přístroje, a to buď prostou prohlídkou kódu, nebo speciálními

Požadavky na výrobce: přístroj by měl být vybaven rozhraním, pomocí kterého by bylo možné zadávat/získávat vstupy či výstupy. Pokud tomu tak není, musí takové zařízení na vyžádání dodat.

ER5.1: Software musí být i se svým hardwarovým a softwarovým prostředím dostatečně dokumentován.

Pro modulární měřicí systém uvedený v tomto příkladu musí být poskytnuta alespoň následující dokumentace:

Poznámky k úrovni zkoušení:

Nízká: *postačuje manuál k přístroji a technická dokumentace. Není nutné poskytovat speciální dokumentaci k softwaru, nicméně poskytnutá dokumentace musí zahrnovat identifikaci softwaru a příslušné deklarace výrobce (např. deklaraci ochranného rozhraní).*

Střední: *navíc je nutné poskytnout následující dokumentaci:*

- *podrobný popis všech parametrů a funkcí relevantních z pohledu legální metrologie,*
- *popis měřicích algoritmů (např. výpočet ceny, zaokrouhlování),*
- *popis grafického uživatelského rozhraní, nabídek atd.,*
- *identifikaci softwaru,*
- *kompletní popis všech parametrů a příkazů, které je možné zadat přes ochranné softwarové rozhraní, spolu s deklarací jeho kompletnosti,*
- *kompletní popis všech parametrů a příkazů, které je možné zadat přes ochranné rozhraní, spolu s deklarací jeho kompletnosti,*
- *popis přenášených dat,*
- *požadavky na operační systém a hardware,*
- *odkaz na příslušné požadavky tohoto dokumentu,*
- *manuál k přístroji.*

Vysoká: *navíc je nutné poskytnout zdrojový kód programu spolu s následující dokumentací:*

- *logický diagram softwaru,*
- *podrobný popis funkcí každého modulu, který je relevantní z pohledu legální metrologie,*
- *popis přenášených datových struktur.*

7 LITERATURA

[1] Directive 2004/22/EC of the European Parliament and of the Council of 31 March 2004 on measuring instruments. Official Journal of the European Union L 135/1, 30. 4. 2004
[2] Council Directive 90/384/EEC on the harmonization of the laws of the Member States relating to non-automatic weighing instruments. Official Journal of the European Communities, L 189, Vol. 33, 20. 7. 1990, 1–16
[3] Guide for Examining Software (Non-automatic Weighing Instruments), WEMEC 2. 3. 1995
[4] IEC 65(Sec)183 Software Documentation, 1994
[5] ISO 7498-1 to -4, Information technology – Open Systems Interconnection, 1989–1997
[6] ISO/IEC 9126 Information technology; Software product evaluation, October 1994
[7] ISO/IEC 12119 Information technology; Software packages; Quality requirements and testing, August 1995
[8] Information Technology Security Evaluation Criteria (ITSEC), June 1991, Version 1.2, Document COM(90) 314, Luxembourg
[9] ISO 8731-1:1987 Banking – Approved algorithms for message authentication - Part 1 : DEA
[10] ISO 10126-2:1991 Banking – Procedures for message encipherment - Part 2: DEA algorithm
[11] ITU-T (formerly CCITT) V.42
[12] ISO/IEC 13239 Information technology – Telecommunication an Information exchange between systems -High-level data link control (HDLC) procedures, 1996
[13] Security of computerized instruments, Jean-François Magana, OIML Bulletin Volume XL, Number 3, July 1999
[14] ISO 2382-1, Information technology, Part 1: Fundamental Terms, 1993 ISO 2382-7, Information technology, Part 7: Computer Programming, 1989
[15] Software Requirements and Validation Guide, Version 1.00, 29 October 2004, European Growth Network „MID-Software", contract number G7RT-CT-2001-05064, 2004
[16] Software Guide (Measuring Instruments Directive 2004/22/EC), WEMEC 7.2, Issue 1, 2005

8 REVIZE TOHOTO DOKUMENTU

vydání	datum	Změny od přechozího vydání
2	Květen 2005	modifikace a uvedení do souladu s dokumentem Welmec 7.2, odstavce 1.1, 3, 4, 4.1, 7,
		Doplněna předmluva, mapa na úvodní straně, 8
		Příloha I smazána

Nakladatelský servis: Bořivoj Kleník, PhDr. – Q-art.
Redakční uzávěrka: 30. 11. 2006. NEPRODEJNÉ