



استاندارد ملی ایران

۸-۸۴۵۳

چاپ اول

اردیبهشت ۱۳۹۲



جمهوری اسلامی ایران

Islamic Republic of Iran

سازمان ملی استاندارد ایران

Iranian National Standardization Organization

INSO

8453-8

1st. Edition

May.2013

کارت‌های شناسایی - روش‌های آزمون -

قسمت ۸:

گذرگاه سری جهانی - کارت مدار مجتمع

(USB-ICC)

Identification cards — Test Methods —

Part 8:

USB-ICC

ICS:35.240.15

به نام خدا

آشنایی با سازمان ملی استاندارد ایران

مؤسسه استاندارد و تحقیقات صنعتی ایران به موجب بند یک ماده ۳ قانون اصلاح قوانین و مقررات مؤسسه استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱ تنها مرجع رسمی کشور است که وظیفه تعیین، تدوین و نشر استانداردهای ملی (رسمی) ایران را به عهده دارد.

نام موسسه استاندارد و تحقیقات صنعتی ایران به موجب یکصد و پنجاه و دومین جلسه شورای عالی اداری مورخ ۹۰/۶/۲۹ به سازمان ملی استاندارد ایران تغییر و طی نامه شماره ۲۰۶/۳۵۸۳۸ مورخ ۹۰/۷/۲۴ جهت اجرا ابلاغ شده است.

تدوین استاندارد در حوزه های مختلف در کمیسیون های فنی مرکب از کارشناسان سازمان، صاحب نظران مراکز و مؤسسات علمی، پژوهشی، تولیدی و اقتصادی آگاه و مرتبط انجام می شود و کوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانه صاحبان حق و نفع، شامل تولیدکنندگان، مصرف کنندگان، صادرکنندگان و وارد کنندگان، مراکز علمی و تخصصی، نهادها، سازمان های دولتی و غیر دولتی حاصل می شود. پیش نویس استانداردهای ملی ایران برای نظرخواهی به مراجع ذی نفع و اعضای کمیسیون های فنی مربوط ارسال می شود و پس از دریافت نظرها و پیشنهادات در کمیته ملی مرتبط با آن رشته طرح و در صورت تصویب به عنوان استاندارد ملی (رسمی) ایران چاپ و منتشر می شود.

پیش نویس استانداردهایی که مؤسسات و سازمان های علاقه مند و ذی صلاح نیز با رعایت ضوابط تعیین شده تهیه می کنند در کمیته ملی طرح و بررسی و در صورت تصویب، به عنوان استاندارد ملی ایران چاپ و منتشر می شود. بدین ترتیب، استانداردهایی ملی تلقی می شوند که بر اساس مفاد نوشته شده در استاندارد ملی ایران شماره ۵ تدوین و در کمیته ملی استاندارد مربوط که سازمان ملی استاندارد ایران تشکیل می دهد به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان بین المللی استاندارد (ISO)^۱، کمیسیون بین المللی الکتروتکنیک (IEC)^۲ و سازمان بین المللی اندازه شناسی قانونی (OIML)^۳ است و به عنوان تنها رابط^۴ کمیسیون کدکس غذایی (CAC)^۵ در کشور فعالیت می کند. در تدوین استانداردهای ملی ایران ضمن توجه به شرایط کلی و نیازمندی های خاص کشور، از آخرین پیشرفت های علمی، فنی و صنعتی جهان و استانداردهای بین المللی بهره گیری می شود. سازمان ملی استاندارد ایران می تواند با رعایت موازین پیش بینی شده در قانون، برای حمایت از مصرف کنندگان، حفظ سلامت و ایمنی فردی و عمومی، حصول اطمینان از کیفیت محصولات و ملاحظات زیست محیطی و اقتصادی، اجرای بعضی از استانداردهای ملی ایران را برای محصولات تولیدی داخل کشور و/یا اقلام وارداتی، با تصویب شورای عالی استاندارد، اجباری نماید. سازمان می تواند به منظور حفظ بازارهای بین المللی برای محصولات کشور، اجرای استانداردهای کالاهای صادراتی و درجه بندی آن را اجباری نماید. همچنین برای اطمینان بخشیدن به استفاده کنندگان از خدمات سازمان ها و مؤسسات فعال در زمینه مشاوره، آموزش، بازرسی، ممیزی و صدور گواهی سیستم های مدیریت کیفیت و مدیریت زیست محیطی، آزمایشگاه ها و مراکز کالیبراسیون (واسنجی) و وسایل سنجش، سازمان ملی استاندارد ایران این گونه سازمان ها و مؤسسات را بر اساس ضوابط نظام تأیید صلاحیت ایران ارزیابی می کند و در صورت احراز شرایط لازم، گواهینامه تأیید صلاحیت به آن ها اعطا و بر عملکرد آن ها نظارت می کند. ترویج دستگاه بین المللی یکاها، کالیبراسیون (واسنجی) و وسایل سنجش، تعیین عیار فلزات گرانبها و انجام تحقیقات کاربردی برای ارتقای سطح استانداردهای ملی ایران از دیگر وظایف این سازمان است.

1- International Organization for Standardization

2 - International Electrotechnical Commission

3- International Organization of Legal Metrology (Organisation Internationale de Metrologie Legale)

4 - Contact point

5 - Codex Alimentarius Commission

کمیسیون فنی تدوین استاندارد
«کارت‌های شناسایی – روش‌های آزمون –
قسمت ۸: گذرگاه سری جهانی – کارت مدار مجتمع (USB-ICC)»

رئیس:

سمت و/یا نمایندگی

فیاضی، مهدی
(لیسانس برق الکترونیک)

کارشناس و مسئول تدوین استاندارد امنیت شبکه- سازمان
فناوری اطلاعات ایران

دبیر:

میراسکندری، سید محمدرضا
(لیسانس کامپیوتر- نرم افزار)

مدیر کل اداره خدمات ارزش افزوده- سازمان فناوری
اطلاعات ایران

اعضا: (اسامی به ترتیب حروف الفبا)

بختیاری، شیرین
(لیسانس برق-الکترونیک)

کارشناس سازمان فناوری اطلاعات ایران

جمالی، وحید
(فوق لیسانس برق مخابرات)

کارشناس دانشگاه خواجه نصیر الدین طوسی

سعیدی، عذراء
(فوق لیسانس برق-مخابرات)

کارشناس سازمان فناوری اطلاعات ایران

صوفی زاده، جلیل
(دکترای مخابرات)

مشاور ارشد سازمان فناوری اطلاعات ایران

فرهاد شیخ احمد، لیلا
(فوق لیسانس کامپیوتر-نرم افزار)

کارشناس سازمان فناوری اطلاعات ایران

قسمتی، سیمین
(فوق لیسانس فناوری اطلاعات)

کارشناس سازمان فناوری اطلاعات ایران

کبیری، پیمان
(دکترای کامپیوتر)

عضو هیات علمی دانشگاه علم و صنعت

استادیار برق دانشگاه شهید عباسپور

لطیف شبگاهی، غلامرضا
(دکترای برق)

کارشناس دانشگاه خواجه نصیر الدین طوسی

مالکی، علیرضا
(لیسانس برق - مخابرات)

کارشناس سازمان فناوری اطلاعات ایران

موجبی، محمود
(فوق لیسانس مخابرات)

عضو هیات علمی دانشگاه خواجه نصیر الدین طوسی

میرزاحسینی، داوود
(فوق لیسانس الکترونیک)

فهرست مندرجات

صفحه	عنوان
ب	آشنایی با سازمان ملی استاندارد ایران
ج	کمیسیون فنی تدوین استاندارد
۱.	۱ هدف و دامنه کاربرد
۱	۲ مراجع الزامی
۲	۳ اصطلاحات و تعاریف
۴	۴ نمادها و کوتاه نوشتها
۴	۵ آزمون ویژگی‌های فیزیکی و الکتریکی USB
۴.	۱-۵ مقدمه
۵	۶ راه اندازی آزمون
	۱-۶ پیکربندی پایه
	۵
۶	۲-۶ ویژگی‌های تحلیل‌گر پروتکل USB
۷	۳-۶ افزاره‌های تحت آزمون
۸	۷ رده بندی آزمون‌ها و معیارهای اعتبارسنجی
۸	۱-۷ رده بندی آزمون
۸.	۲-۷ انواع آزمون
۹.	۳-۷ اهداف آزمون
۹..	۴-۷ بحرانی بودن آزمون
۱۰	۵-۷ معیارهای پذیرش و رد آزمون
۱۰	۸ روش اجرایی آزمون انطباق پروتکل USB
۱۰	۱-۸ گروه‌های آزمون‌های پروتکل USB
۱۱	۲-۸ چارچوب توصیف آزمون
۱۲	۱-۲-۸ آزمون مدیریت سیگنالهای USB
۲۴	۲-۲-۸ آزمون مدیریت زمان بندی
۲۶	۳-۲-۸ آزمون بازیابی خطا و پروتکل سطح پایین
۳۳	۴-۲-۸ آزمون‌های لغو و ره‌ایش
۳۵	۵-۲-۸ ماشین حالت USB
۳۶	۳-۸ آزمون‌های مدیریت درخواست استاندارد
۳۶	۱-۳-۸ اهداف

۳۶	۲-۳-۸ الزامات
۳۶	۳-۳-۸ شرایط ابتدایی
۳۶	۴-۳-۸ موارد نامی
۳۷	۵-۳-۸ فهرست آزمون
۳۹	۶-۳-۸ موارد خطا
۳۹	۷-۳-۸ شرایط عمومی آزمون
۴۱	۸-۳-۸ فهرست آزمون
۴۵	۹ آزمون برای تعیین انطباق با رده CCID
۴۵	۱-۹ اهداف
۴۵	۲-۹ کنترل یک آزمون درخواست انتقال
۴۵	۱-۲-۹ شرح آزمون
۴۵	۲-۲-۹ اهداف
۴۶	۳-۲-۹ ملزومات
۴۶	۴-۲-۹ شرایط ابتدایی USB-ICC
۴۶	۵-۲-۹ راه‌اندازی آزمون
۴۶	۶-۲-۹ فهرست آزمون: درخواست‌های منحصر بفرد در موارد نامی و دارای خطا
	۳-۹ کنترل یک آزمون نمودار وضعیت انتقال
	۴۲
۴۹	۱-۳-۹ اهداف
۴۹	۲-۳-۹ ملزومات
۴۹	۳-۳-۹ شرایط ابتدایی
۵۰	۴-۳-۹ نوع آزمون
۵۰	۵-۳-۹ شرح آزمون
۵۰	۶-۳-۹ نسخه A انتقال کنترل: فهرست آزمون
۵۴	کتاب‌نامه

پیش‌گفتار

استاندارد «کارت‌های شناسایی - روش‌های آزمون - قسمت ۸: گذرگاه سری جهانی - کارت مدار مجتمع (USB-ICC)» که پیش‌نویس آن در کمیسیون‌های مربوط توسط سازمان فناوری اطلاعات ایران تهیه و تدوین شده و در یکصد و چهل و هفتمین اجلاس کمیته ملی استاندارد خدمات مورخ ۹۱/۸/۲۹ مورد تصویب قرار گرفته است، اینک به استناد بند یک ماده ۳ قانون اصلاح قوانین و مقررات سازمان استاندارد و تحقیقات صنعتی ایران، مصوب بهمن ماه ۱۳۷۱، به عنوان استاندارد ملی ایران منتشر می‌شود.

برای حفظ همگامی و هماهنگی با تحولات و پیشرفت‌های ملی و جهانی در زمینه صنایع، علوم و خدمات، استانداردهای ملی ایران در مواقع لزوم تجدید نظر خواهد شد و هر پیشنهادی که برای اصلاح و تکمیل این استانداردها ارائه شود، هنگام تجدید نظر در کمیسیون فنی مربوط مورد توجه قرار خواهد گرفت. بنابراین، باید همواره از آخرین تجدیدنظر استانداردهای ملی استفاده کرد.

منبع و ماخذی که برای تهیه این استاندارد مورد استفاده قرار گرفته به شرح زیر است:

ISO/IEC 10373-8: 2011: Identification cards - Test Methods – Part 8: USB-ICC

مقدمه

گذرگاه سری جهانی - کارت‌های مدار مجتمع (USB-ICC)^۱ افزاره‌ی پیچیده‌ای است که از پروتکل USB پشتیبانی می‌کند. پروتکل هنگام نیاز تولیدکننده کارت به آماده‌سازی طرح تایید اعتبار^۲، ساختار لایه‌ای پروتکل USB شامل به کارگیری USB-ICC در پیکربندی‌های مختلف آزمون است. به‌علاوه، هر افزاره USB متعلق به یک رده USB است. بنابراین آزمون هر افزاره USB در سطح وسیع نیاز به ارتقای دقیق یک طرح آزمون دارد که شامل یکی از سه گروه زیر است:

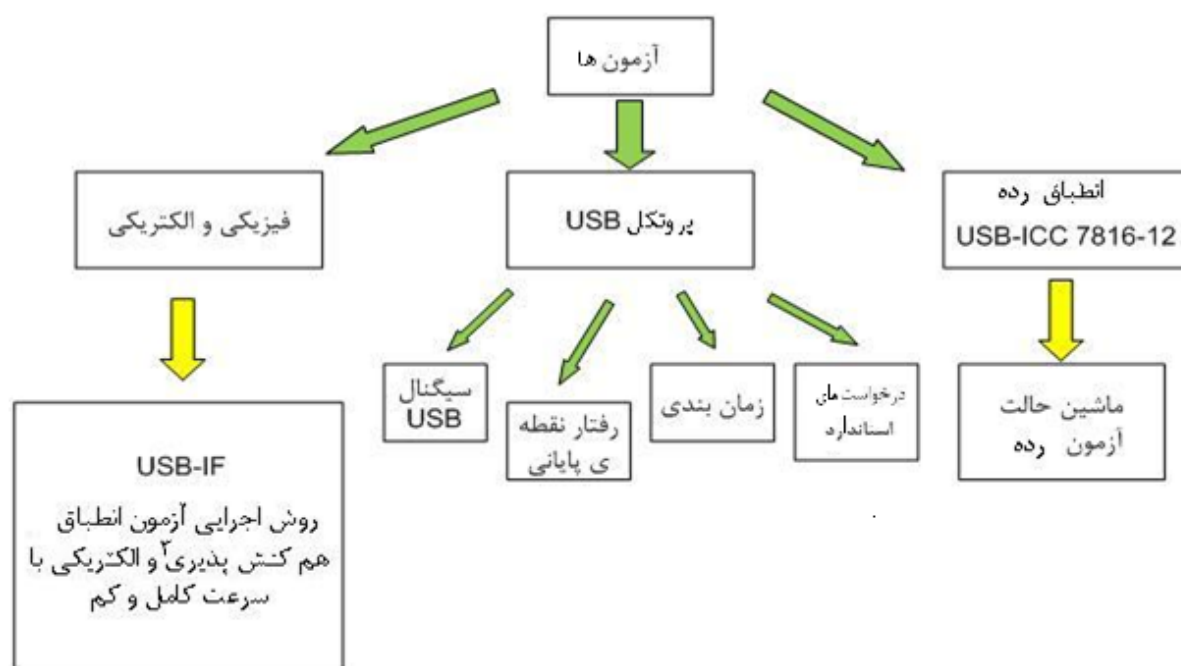
(۱) ارزیابی ویژگی‌های فیزیکی و الکتریکی؛

(۲) اجرای موثر پروتکل USB؛

(۳) اجرای آزمون‌های طراحی شده برای اثبات انطباق افزاره USB با توجه به رده مخصوص به آن؛

این گروه‌های سطح بالای آزمون از یک سری سناریوهای آزمون منحصر به فرد^۳ ساخته شده‌اند. این سناریوها افزاره را به چالش می‌کشند و طوری طراحی شده که می‌تواند هر عدم انطباق کارت را افشا کند. هدف نهایی تضمین انطباق‌پذیری USB-ICC با دیگر افزاره‌های منطبق USB است.

شکل ۱ خلاصه‌ای از چارچوب یک آزمون تایید اعتبار برای USB-ICC است که این قسمت از استاندارد ISO/IEC 10373 برای آن پیشنهاد می‌کند.



1 - Universal Serial Bus-Integrated Circuit Cards

2 - Validation Plan

3 - Interoperability

شکل ۱- مرور کلی آزمون انطباق

مطابق استاندارد ملی شماره ۷۸۱۶-۱۲، USB-ICC باید مشخصات USB 2.0 را در سطوح فیزیکی و الکتریکی دارا باشد. این مشخصات برای تمام افزاره‌های USB مشترک است؛ این خصوصیت توسط انجمن پیاده‌سازان USB و همچنین چند روش برای آزمون انطباق که می‌تواند برای محک‌زدن برخی از کارکردهای کارت استفاده شود، منتشر شده است. توجه نمایید که استاندارد ملی شماره ۷۸۱۶-۱۲ در واقع رده افزاره USB-ICC و روش‌های اجرایی آزمون مخصوص برای رده USB-ICC غیر قابل دسترس را تشریح می‌کند.

کارت‌های شناسایی – روش‌های آزمون – قسمت ۸: گذرگاه سری جهانی – کارت مدار مجتمع (USB-ICC)

۱ هدف و دامنه کاربرد

هدف از تدوین این استاندارد تعیین روش‌شناسی آزمون و فهرستی از سناریوهای آزمون برای ارزیابی انطباق یک کارت با استاندارد ملی شماره ۱۲-۷۸۱۶ می‌باشد.

این استاندارد برای موارد زیر کاربرد دارد:

- اندازه‌گیری‌های لایه فیزیکی USB 2.0 و آزمون‌های انطباق الکتریکی را مشخص می‌کند؛
- در رابطه با موضوعاتی مربوط به ابزار آزمون برای تحلیل ترافیک گذرگاه USB بحث کرده و در مورد سناریوهای آزمون تعیین شده در این استاندارد راهنمایی می‌کند؛
- رده‌بندی را برای سناریوهای آزمون تعیین شده در این استاندارد و در راستای معیارهای تایید اعتبار ارائه می‌دهد؛

- موارد آزمون را برای انطباق با افزاره‌های رده واسط کارت مدار مجتمع (USB CCID)^۱ مورد بحث قرار می‌دهد؛

یادآوری - انطباق به معنی کارت‌هایی است که محصولات USB-ICC نامیده می‌شوند و برای تطبیق با تعریف ارائه شده در استاندارد ملی شماره ۱۲-۷۸۱۶ طراحی شده‌اند.

۲ مراجع الزامی

مدارک الزامی زیر حاوی مقرراتی است که در متن این استاندارد ملی ایران به آنها ارجاع داده شده‌است. بدین ترتیب آن مقررات جزئی از این استاندارد ملی ایران محسوب می‌شود.

در صورتی که به مدرکی با ذکر تاریخ انتشار ارجاع داده شده باشد، اصلاحیه‌ها و تجدید نظرهای بعدی آن مورد نظر این استاندارد ملی نیست. در مورد مدارکی که بدون ذکر تاریخ انتشار به آن‌ها ارجاع داده شده است، همواره تاریخ تجدید نظر و اصلاحیه‌های بعدی آن‌ها مورد نظر است.

استفاده از مراجع زیر برای این استاندارد الزامی است:

2-1 ISO/IEC 7816-12:2005, *Identification cards — Integrated circuit cards — Part 12: Cards with contacts — USB electrical interface and operating procedures* [RE1 in Test Tags in this part of ISO/IEC 10373]

2-2 Universal Serial Bus specification – Revision 2.0, 27 April 2000 [RE2 in this part of ISO/IEC 10373]

۳ اصطلاحات و تعاریف

در این استاندارد اصطلاحات و تعاریف زیر به کار می‌رود:

۱-۳

درخواست USB^۱

درخواست انتقال کنترل

واحد ارسال برای حالت انتقال کنترل، که شامل یک مرحله راه اندازی^۲ (بند ۳-۵)، یک مرحله داده‌های اختیاری^۳ (بند ۳-۲) و یک مرحله وضعیت^۴ (بند ۳-۶) است.

یادآوری ۱- این درخواست بطور ویژه برای گام شمارش (بند ۳-۴) مورد استفاده قرار می‌گیرد..

یادآوری ۲- مجموع این مراحل یک لایه پروتکل عملی را ایجاد می‌کند.

یادآوری ۳- یک درخواست می‌تواند استاندارد، مخصوص رده یا مخصوص فروشنده باشد.

۲-۳

مرحله داده

قسمتی از انتقال کنترل که داده همراه با درخواست تعیین شده را در یک یا چند تراکنش گذرگاه داده حمل می‌کند.

۳-۳

نقطه‌ی پایانی^۵

کانال ارتباطی ایجاد شده بین یک میزبان و یک افزاره می‌باشد.

یادآوری- یک نقطه‌ی پایانی به جز در مود انتقال کنترل، جایی که نقطه‌ی پایانی به طور هم زمان بالارونده و پایین‌رونده است، معمولاً تک‌سویه می‌باشد.

۴-۳

شمارش^۶

روش اجرایی استاندارد برای تشخیص توسط میزبان USB-ICC برای راه اندازی یک خط ارتباطی، زمانی که میزبان یک آدرس منحصر بفرد به افزاره نشان می‌دهد و راه‌انداز افزاره به خوبی USB را پیکربندی می‌کند.

یادآوری ۱- این گام زمانی آغاز می‌شود که یک افزاره به یک درگاه USB وصل شود.

1 - USB Request

2 - Setup Stage

3 - Data Stage

4 - Status Stage

5 - Endpoint

6 - Enumeration

یادآوری ۲- این گام به یک یا چند راه‌انداز^۱ مناسب افزاره اجازه می‌دهد تا به افزاره تخصیص داده شوند.

۵-۳

مرحله راه اندازی

قسمتی از یک درخواست (مطابق بند ۱-۳) که شامل تعریف درخواست است.

یادآوری - مرحله نصب شامل شناسایی هدف درخواست، جهت آن و طول مرحله داده بعدی است.

۶-۳

مرحله وضعیت

قسمتی از یک درخواست (مطابق بند ۱-۳) برای تصدیق سراسری درخواست می‌باشد.

یادآوری - هر درخواستی (به جز درخواست SetAddress) باید قبل از تکمیل مرحله وضعیت محدود شود.

۷-۳

انتقال^۲

انتقال USB

یک یا چند تراکنش USB می‌باشد.

یادآوری - یک انتقال معمولاً تک‌سویه (جریان پایین‌رونده - از میزبان به سمت افزاره، یا جریان بالارونده - از افزاره به سمت میزبان) می‌باشد. به جز در انتقال کنترل که همیشه دوسویه است.

۸-۳

مُد USB^۳

مُد انتقال استفاده شده توسط پروتکل USB است.

یادآوری - پروتکل USB از چهار مُد انتقال استفاده می‌کند: انتقال کنترل (برای استفاده در شمارش و واسط‌های مشخص)، انتقال منقطع (شبیه‌سازی رفتار منقطع یک سخت‌افزار با استفاده از روش نمونه‌برداری)، انتقال توده‌ای (که معمولاً برای انتقال داده‌های همراه با تاخیر به کار می‌رود) و انتقال همزمان (که برای انتقال داده‌های بدون تاخیر به کار می‌رود).

۹-۳

تراکنش USB^۴

دنباله‌ی یک، دو یا سه مرحله‌ای: نشانه^۵، داده، دست تکانی^۶.

1 - Driver

2 - Transfer

3 - USB Mode

4 - USB Transaction

5 - Token

6 - Handshake

۴ نمادها و کوتاه نوشت

NA	Not applicable	کاربرد ندارد
ACK	Acknowledged	تایید شده
CCID	Integrated Circuit(s) Cards Interface Device conforming to RE5	افزاره واسط کارت‌های مدار مجتمع منطبق با RE5
NAK	Not acknowledged	تایید نشده
STALL	Indicates that a transfer is out of context or wrongly formatted. May require a host intervention.	نشان‌دهنده این است که یک انتقال از محدوده خارج یا اشتباه قالب‌بندی شده است و ممکن است به مداخله یک میزبان نیاز باشد.

۵ آزمون ویژگی‌های فیزیکی و الکتریکی USB ۱-۵ مقدمه

آزمون‌های الکتریکی برای تمامی افزاره‌های منطبق USB مشترک هستند. این مشخصات توسط نظرآزمایی سازنده USB منتشر شده‌اند. استاندارد ISO/IEC 10373-3 بهتراست به این استاندارد مبنا ارجاع دهد [RE4].

مشخصات USB 2.0 نرخ داده‌ها و زمان‌های افزایش زیر را تعریف می‌کند:

جدول ۱ - زمان‌های خیز و نرخ‌های داده

زمان افزایش	نرخ داده	
۷۵ ns - ۳۰۰ ns	۱/۵ Mbit/s	سرعت پایین (LS)
۴ ns - ۲۰ ns	۱۲ Mbit/s	سرعت کامل (FS)
۵۰۰ ps	۴۸۰ Mbit/s	سرعت بالا (HS)

تولیدکنندگان USB بهتراست یک آزمون تایید اعتبار بر پایه مستندات کاربردی USB-IF طراحی کنند.

آزمون الکتریکی USB 2.0 موارد زیر را شامل می‌شود:

- کیفیت سیگنال تفاضلی (آزمون نمودار چشمی، نرخ‌های سیگنال، پهنای EOP^۱، محدوده ولتاژ متقاطع، J-K و K-J جفت شده به همان صورت لغزش^۲ پشت‌سرهم و البته زمان‌های خیز و سقوط)؛

1 - End Of Point

2- Jitter

- بررسی جریان ورودی (جریان کشیده شده توسط CUT) در هنگام اتصال؛

- افت خط توان گذرگاه V بارگذاری شده و اندازه‌های سست؛

برای آزمون‌هایی که انطباق آن‌ها میان انجام شدنی را در سطح الکتریکی تضمین می‌کند، چه زمانی که USB-ICC مستقیماً به میزبان وصل شده‌باشد و یا زمانی که گذرگاه USB را با USB‌های دیگر تقسیم کرده باشد. مشخصات USB 2.0 نرخ داده را تا چهل برابر افزایش می‌دهد، و در نتیجه پیچیدگی سطح بالایی را برای تراشه (پرسرعت) ایجاد می‌کند. آزمون‌های ویژه برای افزاره‌های منطبق پرسرعت شامل حساسیت گیرنده، چیرپ^۱ (علامت‌دهی مخصوص در طی پروتکل تشخیص سرعت افزاره‌های پرسرعت)، یکنواختی (افزایش یا کاهش یکنواخت دامنه سیگنال پرسرعت بدون پاسخ معکوس) و اندازه‌گیری‌های مقاومت ظاهری (CUT) و کابل) است. اندازه‌گیری‌های زمان‌های افزایش بسیار کوتاه نیازمند استفاده از اسیلسکوپ‌های بدون تاخیر همراه با ملزومات مهم برای پهنای باند، نرخ نمونه داده و زمان‌های افزایش و کاهش است. افزاره اتصال USB باید بر طبق ویژگی‌های الکتریکی و پروتکل داده شده در مشخصات USB 2.0، یک اتصال الکتریکی به C1، C5، C4 و C8 برقرار کند.

یادآوری ۱- از زمان انتشار استاندارد ملی شماره ۷۸۱۶-۱۲، یک سری مشخصات USB مربوط [RE3] به عنوان مکملی برای مشخصات USB 2.0 برای فناوری کارت‌های هوشمند منتشر شده‌است. افزاره USB تراشه داخلی یا RE3 با استفاده از انتقال داده USB، ارتباط بین افزاره‌هایی که در رده‌های ولتاژ مختلف (۳۷، ۱/۸۷، ۱/۵۷، ۱/۲۷ و ۱۷) کار می‌کنند را مشخص می‌کند. استاندارد ISO/IEC 7816-12 به مشخصات USB 2.0 که نیازمند منبع تغذیه V_{BUS} در ۵۷ نامی است، اشاره می‌کند. گرچه پیکربندی و انتقال داده تعریف شده در استاندارد ملی شماره ۷۸۱۶-۱۲ ممکن است توسط کارت‌های منطبق با USB تراشه داخلی نیز پشتیبانی شوند. قواعد انطباق در رابطه با محصولات USB تراشه داخلی هنوز معین نشده‌اند.

یادآوری ۲- بر طبق استاندارد ملی شماره ۷۸۱۶-۱۲، افزاره‌های اتصال USB باید پیرو ویژگی‌های الکتریکی و قراردادهای داده شده در مشخصات USB 2.0، به C1، C5، C4 و C8 اتصال الکتریکی برقرار کنند. این سند این مشخصات الکتریکی را آدرس‌دهی نمی‌کند.

۶ راه اندازی آزمون

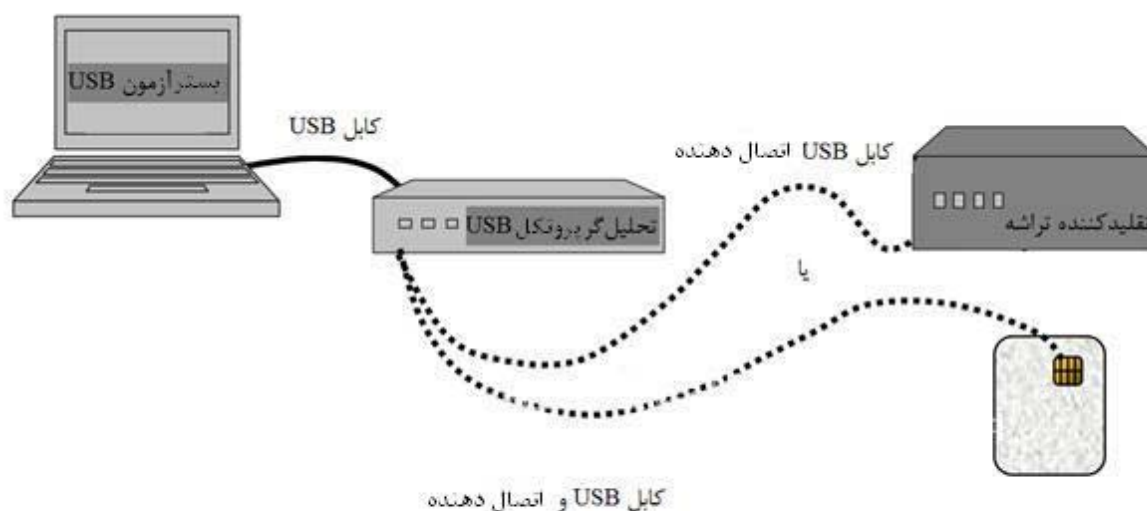
۱-۶ پیکربندی پایه

راه اندازی آزمون پایه که انجام فرایندهای آزمون تعریف شده در این قسمت را ممکن می‌سازد، به نوعی از سه افزاره مختلف ساخته شده است:

۱- **بستر آزمون USB**، یک رایانه شخصی میزبان، که تمام نرم‌افزار مورد نیاز را برای راه‌اندازی تحلیل‌گر پروتکل USB در درون خود جای داده است.

۲- **تحلیل‌گر پروتکل USB** برای گرفتن انتقال داده بین رایانه شخصی میزبان و یک افزاره تحت آزمون (برای مثال USB-ICC) پیکربندی شده است و اطلاعات را درباره بسته‌های ضبط‌شده به نمایش می‌گذارد و نرم‌افزار USB را جهت تولید ترافیک USB اجرا می‌کند.

۳- USB-ICC تحت آزمون و یا یک تقلید کننده تراشه USB-ICC. آزمون‌های مفصلی می‌تواند با استفاده از تقلید کننده تراشه، یا یک USB-ICC، یا یک سامانه توسعه دهنده دیگر (برای مثال bond-out) انجام شود. برای اطلاعات بیشتر به بند ۶-۳ مراجعه شود. شکل ۲ پیکربندی آزمون اساسی را شرح می‌دهد:



یادآوری - دیگر روش‌های پیکربندی آزمون نیز ممکن است.

شکل ۲ - پیکربندی آزمون اساسی

۲-۶ ویژگی‌های تحلیل‌گر پروتکل USB

تحلیل‌گر پروتکل USB توسط بستر آزمون USB که از طریق یک درگاه USB به آن متصل است، پیکربندی و کنترل می‌شود.

بستر آزمون USB تحلیل‌گر پروتکل USB را در مود تولید ترافیک پیکربندی می‌کند.

تحلیل‌گرهای پروتکل USB تجاری به‌طور معمول اجزا گسترش داده شده مانند یک انتقال داده USB را پشتیبانی می‌کنند. در داخل USB، در پایین‌ترین سطح تحلیل (سطح بسته) بسته‌های منحصر بفرد^۱ بیان‌گر مود ارتباط اتفاق افتاده است. دومین سطح (سطح تراکنش گذرگاه) بسته‌ها را برای تبدیل به تراکنش‌های گذرگاه (کنترل، توده، IN وقفه) ترکیب می‌کند، جایی که فعل و انفعال حقیقی و عملی بین میزبان و USB-ICC اتفاق می‌افتد. بالاترین سطح (سطح انتقال) بر اساس تعاریف رده افزاره (رده CCID برای USB-ICC) تعریف شده است.

کدگشایی سطح انتقال داده USB اجازه نمایش تعامل گذرگاه بین افزاره‌های USB مورد آزمون یا تحلیل در یک سطح را می‌دهد که بیشتر گویای عملیات در حال انجام است.

تحلیل‌گر پروتکل USB به‌طور معمول شامل شرایطی برای تشخیص و راه‌اندازی^۱ در جنبش رویدادهای زیادی می‌باشد. این رویدادها شامل شرایط گذرگاه ویژه تعریف شده توسط کاربر، بسته‌های تطبیق شده با هر گونه شناسه بسته (PID)، بسته‌های تطبیق شده با یک علامت یا تراکنش راه‌اندازی، الگوهای داده، و شرایط غیرعادی زیادی در گذرگاه (خطا) که داده بخصوص، خطاها و دیگر شرایط خواسته شده را تعیین می‌کند، می‌باشند.

این رویدادهای از پیش انتخاب و راه‌اندازی شده ممکن است توسط بستر آزمون USB ضبط و نمایش داده-شوند. تشخیص بدون تاخیر رویدادها می‌تواند توسط شخص به کار انداخته یا از کار افتاده شوند تا بتوان رویدادها را در همان زمانی که اتفاق می‌افتند راه‌اندازی کرد. این امر شامل استثناها یا شرایط خطای از پیش تعریف شده و یک سری شرایط جستجو تعریف شده توسط کاربر است.

۳-۶ افزاره‌های تحت آزمون

توصیه می‌شود که آزمون‌های تعریف شده در بندهای ۸ و ۹ با استفاده از دو بستر آزمون مختلف زیر تکمیل شوند:

الف) اولین بستر از یک تقلید کننده تراشه تحت آزمون و یک تخته واسط کارت‌خوان پیکربندی شده برای ارتباطات USB تشکیل شده است. یک رایانه شخصی نیز که از درگاه پی‌درپی استفاده می‌کند، ممکن است برای drive این تقلید کننده استفاده شود. تخته واسط کارت‌خوان (از نوع USB) می‌تواند از طریق یک متصل کننده USB که در شکل ۳ نشان داده شده، به درگاه ضبط تحلیل‌گر پروتکل USB متصل شود.



شکل ۳ - بستر آزمون ۱

ب) پیکربندی دوم آزمون از یک کارت هوشمند نمونه (ساخته شده با آخرین جزء) به همراه یک متصل کننده USB که به درگاه ضبط تحلیل‌گر پروتکل USB همان‌طور که در شکل ۴ نشان داده شده متصل شده است، تشکیل شده است.

1- Triggering On
2 - Packet ID



شکل ۴- بستر آزمون ۲

۷ رده بندی آزمون‌ها و معیارهای اعتبارسنجی

این فصل یک رده بندی از سناریوهای آزمون را که در بند ۸ با جزئیات شرح داده شده است، ارائه می‌دهد.

۱-۷ رده بندی آزمون

هر سناریوی آزمون که در بند ۸ شرح داده شد به صورت زیر و با توجه به نوع، هدف و حساسیت توصیف می‌شود:

- نوع آزمون: بر طبق معیارهای تعریف شده در بند ۲-۷ می‌تواند منفی (N) یا نامی (P) باشد.
- هدف آزمون: بر اساس معیارهای تعریف شده در بند ۴-۷ می‌تواند ویژگی‌های هسته (C)، شکل اولیه (P)، وابستگی‌های متنی (D)، امنیت (S)، و اثرات جانبی (E) باشد.
- بحرانی بودن آزمون^۱: بر اساس معیارهای تعریف شده در بند ۴-۷ می‌تواند بحرانی (C)، بزرگ (M) و یا کوچک (m) باشد.

۲-۷ انواع آزمون

نوع آزمون به شرایطی که آزمون در آن انجام می‌شود در مقایسه با شرایط واقعی وابسته است. در این بخش سه نوع آزمون معرفی شده‌اند:

- آزمون‌های منفی (N): این دسته از آزمون‌ها توانایی USB-ICC را در خارج نشدن از کنترل یا نرسیدن به وضعیتی نامطلوب هنگامی که به طور تصادفی با یک مورد غیراستاندارد مواجه می‌شوند، به چالش می‌کشد. مثال‌هایی شامل نواقص هم‌تا، افزاره‌های USB میزبان یا راه‌اندازهای غیرمنطبق. آزمون‌های منفی موجود در بند ۸ با علامت N مشخص شده‌اند؛

- آزمون‌های مثبت نامی (P): این آزمون‌ها از عملکرد مناسب کارت‌ها طبق استاندارد ملی شماره ۷۸۱۶-۱۲ اطمینان حاصل می‌کنند. آزمون‌های مثبت در بند ۸ با علامت P مشخص شده‌اند.

این آزمون‌های مثبت خود به دو دسته مختلف تقسیم می‌شوند:

1- Test Criticality
2- Peer Defects

- **آزمون مورد نامی:** یک میزبان، یک انتقال USB نامی را بر طبق USB 2.0 اجرا می‌کند و از USB-ICC انتظار می‌رود که پاسخ مناسبی را ارائه دهد.

- **آزمون مورد خطا:** میزبان عمداً یک خطای پروتکل USB را در تراکنش‌های USB ایجاد می‌کند و انتظار می‌رود که خطا توسط USB-ICC تشخیص داده شود. برای مثال، از یک خطا در بسته داده برای یک تراکنش OUT انتظار می‌رود که در یک کارت که پاسخ نمی‌دهد، تأثیر بگذارد و زمان پایان میزبان مربوطه (کارت دست‌تکانی نمی‌کند) باعث اجرای یک RETRY که توسط میزبان آغاز می‌شود، می‌گردد.

۳-۷ اهداف آزمون

هدف هر آزمون به یک قسمت خاص از USB-ICC تحت آزمون اشاره دارد. در این بخش پنج رده آزمون معرفی شده‌اند:

- **ویژگی‌های اصلی^۱ (C):** این‌گونه آزمون‌ها از کارت در شرایط نامی استفاده کرده و عملکرد سرویس‌های پایه USB-ICC را بررسی می‌کنند. برای مثال یک الگوریتم با چند ورودی مختلف اجرا شده و خروجی‌های مربوطه بررسی می‌شوند. در این سند این آزمون‌ها با علامت C مشخص شده‌اند؛

- **نمونه‌ی اولیه^۲ (P):** این‌گونه آزمون‌ها برای بررسی مرزهای پارامترهای ورودی یک دستور داده شده استفاده می‌شوند. این آزمون‌ها با علامت P در این سند مشخص شده است؛

- **وابستگی متنی^۳ (D):** این دسته از آزمون‌ها اطمینان حاصل می‌کنند که زمینه کلی مورد نیاز برای اجرای یک دستور فراهم باشد. این آزمون‌ها با علامت D در این سند مشخص شده‌اند؛

- **آزمون‌های امنیت^۴ (S):** این آزمون‌ها بررسی می‌کنند که دستور داده شده فقط زمانی اجرا شود که شرایط امنیتی مورد نیاز مهیا باشد. ذکر این نکته لازم است که این آزمون‌ها ممکن است با "آزمون‌های متنی" تداخل پیدا کند. در این سند این آزمون‌ها با علامت S مشخص شده‌اند؛

- **اثرات جانبی^۵ (E):** این آزمون‌ها برای اطمینان حاصل کردن از این امر است که یک دستور تا زمانی که به‌طور کامل انجام نشده است، زمینه اجرایی خود را تغییر ندهد. این آزمون‌ها در این سند با علامت E مشخص شده‌اند؛

۴-۷ بحرانی بودن آزمون

بحرانی بودن آزمون اطلاعاتی در مورد اهمیت و اثر مربوط به یک شکست در زمان انجام آزمون فراهم می‌کند. در اینجا سه مرحله معرفی شده است:

1 - Core Features

2 - Prototype

3 - Contextual Dependency

4 - Security Tests

5 - Side Effects

- بحرانی^۱ (C): یک شکست که در زمان انجام آزمون با عنوان بحرانی مشخص می‌شود، به معنی یک اشکال نرم‌افزاری جدی در USB-ICC می‌باشد که منجر به پایان یک فرایند معتبرسازی بر طبق این سند می‌شود و احتمالاً نیازمند طراحی دوباره کارت است. این آزمون‌ها با علامت C مشخص شده‌اند؛

- بزرگ^۲ (M): یک شکست که در زمان انجام آزمون با عنوان بزرگ مشخص می‌شود، به معنی یک اشکال مهم می‌باشد، اما از کارکرد مناسب دیگر اجزای کارت جلوگیری نمی‌کند. برای مثال، کشف این نوع اشکال در زمان توزیع کارت ممکن است به طراحی یک سافت ماسک^۳ برای اصلاح این اشکال منجر شود. این آزمون‌ها با علامت M مشخص شده‌اند؛

- کوچک^۴ (m): یک شکست که در زمان انجام آزمون با عنوان کوچک مشخص می‌شود، به معنی اشکالی کم‌اهمیت است، یعنی محصول در اکثر اوقات به خوبی عمل می‌کند، طوریکه نقص آن معلوم نیست. این آزمون‌ها با علامت m مشخص شده‌اند.

۵-۷ معیارهای پذیرش و رد آزمون

در بیشتر سناریوهای آزمون، برای انجام موفقیت‌آمیز آزمون، USB-ICC باید یک جواب را برای یک دستور داده شده به عنوان یک واکنش ارسال نماید.

تنها آزمونی که در آن USB-ICC داده را باز نمی‌گرداند، آزمون گسسته است که در آن توان، به‌طور عمد قطع شده و کارت زمان لازم برای پاسخگویی را دارا نخواهد بود.

در موارد دیگر، کلمه وضعیت (SW1-SW2) و داده احتمالی برگشت داده شده باید با آن‌هایی که توسط آزمون انتظار می‌رود، مطابقت داشته باشند.

یک سناریوی آزمون باید با موفقیت انجام شده در نظر گرفته شود اگر متن مربوط به آن مطابق شرایط بیان شده در این بخش اجرا شود. (بدون گزارش خطاهای کلمه وضعیت یا خطاهای داده)

یک سناریوی آزمون باید شکست خورده در نظر گرفته شود اگر متن آزمون مربوط به آن بر اساس هر کدام از شرایط بیان شده در این بخش اجرا نشود.

۸ روش اجرایی آزمون انطباق پروتکل USB

۱-۸ گروه‌های آزمون‌های پروتکل USB

این بخش چهار گروه آزمون ارزیابی عملکرد USB در سرعت پایین و/یا بالا را در مقابل الزامات USB 2.0 معرفی می‌کند:

- 1 - Critical
- 2 - Major
- 3 - Soft Mask
- 4 - Minor

- آزمون‌های مربوط به مدیریت سیگنال USB خاص توسط کارت (SE0، تعلیق، از سرگیری) که در بند ۸-۲-۱ فهرست شده‌اند. این آزمون‌ها تشخیص سرعت، بازنشانی USB، تعلیق، از سرگیری و مدیریت انباشتن بیت را پوشش می‌دهند.

- آزمون‌های مربوط به مدیریت زمان برای تاخیر بسته‌های داده که در بند ۸-۲-۲ فهرست شده‌اند.
- آزمون‌های مربوط به رفتار نقطه‌ی پایانی (مدیریت سطح پایین پروتکل و بازیابی خطا) که در بند ۸-۲-۳ فهرست شده‌اند.

- مدیریت درخواست‌های استاندارد و ماشین وضعیت همراه آن در بند ۸-۲-۴ فهرست شده‌اند.
- مشخصات گذرگاه سری جهانی - Revision 2.0, 27 April 2000 (RE2) باید برای ساختن موارد آزمون استفاده شود.

۸-۲ چارچوب توصیف آزمون

موارد آزمون در این بخش توسط یک نام و یک پرچسب شناسایی می‌شوند.
پرچسب‌های آزمون باید به شکل رو به رو ساخته شوند: T_{نام آزمون}_RE2_{بخش RE2}
عبارت RE2 در این بخش به مشخصات USB 2.0 ارجاع می‌دهد و بخش RE2 به بخش‌های مشخصات USB 2.0 ارجاع می‌دهد، جایی که ویژگی تحت آزمون تشریح شده است.
موارد آزمون بر اساس نوع، رده و سطح بحرانی بودن آن‌ها بر اساس بند ۷ مشخص می‌شوند.
این اطلاعات شناسایی در راستای توصیف آزمون و دیگر اطلاعات مربوط به آزمونی خاص در جدول ذیل به طور خلاصه آمده است:

جدول ۲ - الگو برای توصیف آزمون

نام آزمون	نوع: مطابق بند ۲-۷	هدف: مطابق بند ۳-۷	بحرانی بودن: مطابق بند ۴-۷	سند: RE2 USB Specification 2.0
برچسب: T_ {نام آزمون} _RE2_ {بخش مربوطه از RE2}				
عنوان: عنوان ارایه شده برای آزمون مشخص شده توسط نام و برچسب آن است.				
شرط آزمون: معمولاً سرعت مورد استفاده در آزمون است.				
توصیف: روش اجرایی انجام آزمون که به طور معمول به صورت گام به گام بیان می شود.				
نتایج مورد انتظار: پاسخ درست توسط USB-ICC بر اساس روش اجرایی آزمون در شرایط استاندارد است.				
زمینه ابتدایی: شرایط ابتدایی برای USB-ICC و یا راه اندازی آزمون است.				
توضیحات: اطلاعات اضافی درباره اجزا آزمون و دیگر موارد مربوطه است.				
پیاده سازی: مطابق بند ۶ راه اندازی آزمون است.				

۱-۲-۸ آزمون مدیریت سیگنال های USB

۱-۲-۸-۱ شناسایی سرعت

جدول ۳ - آزمون شناسایی سرعت

SpeedID	نوع: N	رده: C	بحرانی بودن: C	سند: RE2
برچسب: T_SpeedID_RE1_7.1.5.1				
عنوان: بررسی مدیریت سرعت USB-ICC				
شرط آزمون: آزمون در سرعت پایین و برای یک افزاره سرعت بالا انجام می‌گیرد و برعکس.				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>GetDescriptor(Device)</i> با استفاده از سرعت ارتباط اشتباه ارسال کنید.</p> <p>۳. یک <i>GetDescriptor(Device)</i> ارسال کنید و از درست بودن جواب اطمینان حاصل کنید.</p>				
نتایج مورد انتظار: پاسخ درست به دومین <i>GetDescriptor</i> .				
زمینه ابتدایی: USB-ICC باید قطع باشد.				
توضیحات:				
پیاده سازی: بستر آزمون USB				

USBReset ۲-۱-۲-۸

جدول ۴ - آزمون USBReset1

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBReset1
برچسب: T_USBReset1_RE1_7.1.7.5				
عنوان: بررسی مدیریت زمان بازنشانی				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SetAddress(0x01)</i> بعد از ۲۰ ms بی‌کار بودن افزاره ارسال کنید.</p> <p>(با SOF: Start Of Frame)</p> <p>۳. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس جدید درستی آدرس معین شده را بررسی کنید.</p> <p>۴. یک <i>SE0</i> با طول متغیر ارسال کنید (از ۲/۵ μs تا ۱۰ ms) و برای ۲۰ ms منتظر بمانید (با SOF).</p> <p>۵. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس 0x00، موثر بودن بازنشانی را بررسی کنید.</p>				
نتایج مورد انتظار: پاسخ به آخرین <i>GetDescriptor()</i> باید موجود و صحیح باشد.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر (platform) آزمون USB				

جدول ۵ - آزمون USBReset2

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBReset2
برچسب: T_USBReset2_RE1_7.1.7.5				
عنوان: بررسی مدیریت زمان بازنشانی (مورد خطا)				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SetAddress(0x01)</i> بعد از ۲۰ ms بی‌کار بودن افزاره ارسال کنید. (با SOF)</p> <p>۳. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس جدید درستی آدرس معین شده را بررسی کنید.</p> <p>۴. یک <i>SE0</i> برای ۲ μs ارسال کنید و برای ۲۰ ms منتظر بمانید (با SOF).</p> <p>۵. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس 0x01، موثر نبودن بازنشانی را بررسی کنید.</p>				
نتایج مورد انتظار: پاسخ به آخرین <i>GetDescriptor()</i> باید موجود و درست باشد.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

جدول ۶ - آزمون USBReset3

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBReset3
برچسب: T_USBReset3_RE1_7.1.7.5				
عنوان: بررسی زمان بازیابی بازنشانی				
شرط آزمون: NA				
<p style="text-align: center;">توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SetAddress(0x01)</i> بعد از ۲۰ ms بی کار بودن افزاره ارسال کنید. (با SOF)</p> <p>۳. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس جدید درستی آدرس معین شده را بررسی کنید.</p> <p>۴. یک <i>SE0</i> برای ۱۰ ms ارسال کنید.</p> <p>۵. یک <i>SetAddress(0x01)</i> بعد از ۱۰ ms بی کار بودن افزاره (idle) ارسال کنید. (با SOF)</p> <p>۶. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس 0x01، موثر بودن <i>SetAddress()</i> را بررسی کنید.</p>				
<p style="text-align: center;">نتایج مورد انتظار: <i>SetAddress()</i> باید توسط افزاره پذیرفته شود و پاسخ به آخرین <i>GetDescriptor()</i> باید موجود و درست باشد.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

جدول ۷ - آزمون USBReset4

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBReset4
برچسب: T_USBReset3_RE4_7.1.7.5				
عنوان: بررسی تاثیر بازنشانی (Reset) روی ماشین وضعیت				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SetAddress(0x01)</i> بعد از ۲۰ ms بی کار بودن افزاره ارسال کنید. (با SOF)</p> <p>۳. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس جدید درستی آدرس معین شده را بررسی کنید.</p> <p>۴. یک <i>SetConfiguration(Configuration)</i> را در آدرس جدید ارسال کنید (جایی که پیکربندی در تعریف افزاره، تعریف شده است) و صحت درخواست را بررسی کنید.</p> <p>۵. یک <i>SE0: Single Ended Zero</i> را برای مدت ۱۰ ms ارسال کنید.</p> <p>۶. با ارسال یک <i>SetConfiguration(Configuration)</i> در آدرس 0x00، موثر بودن SE0 را بررسی کنید و با یک <i>STALL</i> رد شدن <i>SetConfiguration()</i> را بررسی کنید. (درخواست خارج از زمینه).</p>				
<p>نتایج مورد انتظار: آخرین <i>SetConfiguration()</i> بخاطر نبود وضعیت «درس‌دهی‌شده»، باید با یک <i>STALL</i> رد شود.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

۳-۱-۲-۸ مدیریت تعلیق

جدول ۸ – آزمون USBSuspend

USBSuspend	نوع: P	رده: C	بحرانی بودن: C	سند: RE2
برچسب: T_USBSuspend_RE1_7.1.7.6				
عنوان: بررسی رفتار تعلیق				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SEO</i> برای مدت ۱۰ ms ارسال کنید و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۳. یک <i>SetAddress(0x01)</i> در آدرس 0x00 ارسال کنید و ۵ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۴. یک وضعیت تعلیق روی گذرگاه (انتقال داده متوقف می‌شود) برای مدت ۱۰۱۳/۵ ms به وجود آورید. (۳/۵ms بدون هیچ‌گونه انتقال داده که افزاره را مجبور به شروع یک تعلیق گذرا می‌کند + ۱۰ ms برای افزاره باقی می‌ماند تا در واقع در مود تعلیق بماند + ۱ s برای اندازه‌گیری جریان تعلیق).</p> <p>۵. یک دنباله/ز سرگیری را برای ۳۰ ms اجرا کرده و ۲۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۶. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس 0x01، معین بودن آدرس را برای افزاره بررسی کنید.</p> <p>۷. یک <i>SetConfiguration(Configuration)</i> را در آدرس جدید ارسال کنید (جایی که پیکربندی در تعریف افزاره، تعریف شده است) و صحت درخواست را بررسی کنید.</p> <p>۸. یک وضعیت تعلیق روی گذرگاه (انتقال داده متوقف می‌شود) برای مدت ۱۰۱۳/۵ ms به وجود آورید. (۳/۵ms بدون هیچ‌گونه انتقال داده که افزاره را مجبور به شروع یک تعلیق گذرا می‌کند + ۱۰ ms برای افزاره باقی می‌ماند تا در واقع در مود تعلیق بماند + ۱ s برای اندازه‌گیری جریان تعلیق).</p> <p>۹. یک دنباله/ز سرگیری را برای ۳۰ ms اجرا کرده و ۲۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۱۰. با ارسال یک <i>GetConfiguration()</i> در آدرس 0x01، معین بودن آدرس را برای افزاره بررسی کنید.</p>				
<p>نتایج مورد انتظار: آخرین <i>GetConfiguration()</i> (نقطه ۱۰) و <i>GetDescriptor()</i> (نقطه ۶) باید</p>				

مورد قبول واقع شده و پاسخ درست را بازگردانند.
زمینه ابتدایی: افزاره باید قطع باشد.
توضیحات: این آزمون تنها زمانی معنا پیدا می کند که جریان مصرف شده توسط افزاره در مود تعلیق اندازه گیری شود. (مطابق پیوست ۱). در ضمن، این آزمون می تواند در جلسه خودکار استفاده شود تا نشان دهد که این مود تعلیق عملکرد افزاره را مختل می کند یا نه. این آزمون باید برای هر پیکربندی در طول عملکرد عادی افزاره انجام شود.
پیاده سازی: بستر آزمون USB

۴-۱-۲-۸ مدیریت از سرگیری^۱

جدول ۹ – آزمون USBResume

USBResume	نوع: P	رده: C	بحرانی بودن: C	سند: RE2
برچسب: T_USBResume_RE1_7.1.7.7				
عنوان: بررسی رفتار ازسرگیری				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SEO</i> برای مدت ۱۰ ms ارسال کنید و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۳. یک <i>SetAddress(0x01)</i> در آدرس 0x00 ارسال کنید و ۵ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۴. یک وضعیت تعلیق روی گذرگاه (انتقال داده متوقف می‌شود) برای مدت ۱۰۱۳/۵ ms به وجود آورید. (۳/۵ms بدون هیچگونه انتقال داده که افزاره را مجبور به شروع یک تعلیق گذرا می‌کند + ۱۰ ms برای افزاره باقی می‌ماند تا در واقع در مود تعلیق بماند + ۱ s برای اندازه‌گیری جریان تعلیق).</p> <p>۵. یک دنباله/ز سرگیری را برای ۲۰ ms اجرا کرده و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۶. با ارسال یک <i>GetDescriptor(Device)</i> در آدرس 0x01، معین بودن آدرس را برای افزاره بررسی کنید.</p> <p>۷. یک <i>SetConfiguration(Configuration)</i> را در آدرس جدید ارسال کنید (جایی که پیکربندی در تعریف افزاره، تعریف شده است) و صحت درخواست را بررسی کنید.</p> <p>۸. یک وضعیت تعلیق روی گذرگاه (انتقال داده متوقف می‌شود) برای مدت ۱۰۱۳/۵ ms به وجود آورید. (۳/۵ms بدون هیچگونه انتقال داده که افزاره را مجبور به شروع یک تعلیق گذرا می‌کند + ۱۰ ms برای افزاره باقی می‌ماند تا در واقع در مود تعلیق بماند + ۱ s برای اندازه‌گیری جریان تعلیق).</p> <p>۹. یک دنباله/ز سرگیری را برای ۲۰ ms اجرا کرده و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۱۰. با ارسال یک <i>GetConfiguration()</i> در آدرس 0x01، معین بودن آدرس را برای افزاره بررسی کنید.</p>				
<p>نتایج مورد انتظار: آخرین <i>GetConfiguration()</i> (نقطه ۱۰) و <i>GetDescriptor()</i> (نقطه ۶) باید موردقبول واقع شده و پاسخ درست را بازگردانند.</p>				

زمینه ابتدایی: افزاره باید قطع باشد.
توضیحات: این آزمون باید برای هر پیکربندی در طول عملکرد عادی افزاره انجام شود.
اجرا: بستر آزمون USB

۵-۱-۲-۸ مدیریت انباشتن بیت^۱

جدول ۱۰ - آزمون USBBitStuff1

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBBitStuff1
برچسب: T_USBBitStuff1_RE1_7.1.9				
عنوان: بررسی آشکارسازی یک خطای انباشتن بیت توسط افزاره				
شرایط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SEO</i> برای مدت ۱۰ ms ارسال کنید و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۳. یک <i>GetDescriptor(0xFF)</i> در آدرس 0x00 با یک خطای انباشتن بیت ارسال کنید و بررسی کنید که افزاره هیچگونه دست‌تکانی را باز نمی‌گرداند.</p> <p>۴. یک تراکنش IN ارسال کرده و مطمئن شوید که افزاره یک STALL را باز می‌گرداند.</p> <p>۵. یک <i>GetDescriptor(Device)</i> در آدرس 0x00 با یک خطای انباشتن بیت ارسال کنید و بررسی کنید که مرحله راه‌اندازی تصدیق شده‌است.</p> <p>۶. یک تراکنش IN ارسال کرده و مطمئن شوید که افزاره یک پاسخ درست را باز می‌گرداند.</p>				
<p>نتایج مورد انتظار: افزاره باید مرحله درخواست راه‌اندازی دارای خطا را رد کند (و همچنین تراکنش بالایی پیش‌رو را نیز رد کند)، و مرحله راه‌اندازی را، که شامل خطا نیست، قبول کند (و همچنین تراکنش بالایی پیرو را قبول کرده و مطابق آن رفتار کند).</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون باید برای تمام نقاط پایانی اجرا شده برای افزاره انجام شود.				
پیاده‌سازی: بستر آزمون USB				

جدول ۱۱ – آزمون USBBitStuff2

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBBitStuff2
برچسب: T_USBBitStuff2_RE1_7.1.9				
عنوان: بررسی مدیریت درست انباشتن بیت توسط افزاره				
شرایط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک <i>SEO</i> برای مدت ۱۰ ms ارسال کنید و ۱۰ ms در وضعیت بدون کار منتظر بمانید. (با <i>SOF</i>)</p> <p>۳. یک <i>GetDescriptor(0xFF)</i> در آدرس 0x00 با یک خطای انباشتن بیت ارسال کنید و بررسی کنید که افزاره هیچگونه دست‌تکانی را باز نمی‌گرداند.</p> <p>۴. یک تراکنش IN ارسال کرده و مطمئن شوید که افزاره یک <i>STALL</i> را باز می‌گرداند.</p> <p>۵. یک <i>GetDescriptor(Device)</i> در آدرس 0x00 با یک خطای انباشتن بیت ارسال کنید و بررسی کنید که مرحله راه‌اندازی تصدیق شده‌است.</p> <p>۶. یک تراکنش IN ارسال کرده و مطمئن شوید که افزاره یک پاسخ درست را باز می‌گرداند.</p>				
<p>نتایج مورد انتظار: افزاره باید مرحله درخواست راه‌اندازی دارای خطا را رد کند (و همچنین تراکنش بالایی پیش‌رو را نیز رد کند)، و مرحله راه‌اندازی را، که شامل خطا نیست، قبول کند (و همچنین تراکنش بالایی پیش‌رو را قبول کرده و مطابق آن رفتار کند).</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون باید برای تمام نقاط پایانی اجرا شده برای افزاره انجام شود.				
پیاده‌سازی: بستر آزمون USB				

جدول ۱۲ - آزمون USBIPDelay1

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBIPDelay1
برچسب: T_USBIPDelay1_RE1_7.1.18				
عنوان: بررسی زمان پاسخ افزاره				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف: ۱. روش اجرایی اتصال افزاره را انجام دهید. ۲. یک حلقه ۱۰۰ تایی درخواست <i>GetDescriptor(Device)</i> ارسال کنید و اطمینان حاصل کنید که پاسخ افزاره (<i>ACK</i> یا <i>DATA0</i> یا <i>DATA1</i>) بعد از پایان بسته فرستاده شده توسط میزبان، کمتر از ۷/۵ برابر زمان یک بیت باشد.				
نتایج مورد انتظار: نتیجه مورد انتظار با اندازه گیری زمان داخل بسته بدست خواهد آمد.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات: این آزمون باید برای تمام نقاط پایانی اجرا شده برای افزاره انجام شود.				
پیاده سازی: بستر آزمون USB				

جدول ۱۳ – آزمون USBIPDelay2

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBIPDelay2
برچسب: T_USBIPDelay2_RE1_7.1.18				
عنوان: بررسی زمان پاسخ افزاره				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
<p style="text-align: center;">توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک مرحله راه‌اندازی <i>GetDescriptor(Device)</i> با تاخیر میان بسته‌ای متغیر ارسال کنید (از ۸ تا ۲۰ برابر زمان یک بیت با افزایش ۱ زمان بیت در هر مرحله).</p> <p>۳. افزاره را بازنشانی کنید (دنباله SE0 معتبر).</p> <p>۴. یک مرحله راه‌اندازی با قاعده از یک درخواست <i>GetDescriptor(Device)</i>، که توسط یک تراکنش <i>IN</i> پیروی می‌شود، ارسال کنید. <i>ACK</i> تراکنش <i>IN</i> را بعد از یک تاخیر متغیر ارسال کنید (از ۸ تا ۲۰ برابر زمان بیت با افزایش ۱ زمان بیت در هر مرحله).</p> <p>۵. با توجه به طول بسته، یک تراکنش <i>IN</i> یا دو تراکنش پی‌درپی خروجی ارسال کنید.</p>				
<p style="text-align: center;">نتایج مورد انتظار: اگر تراکنش نقطه ۵ یک تراکنش <i>IN</i> باشد، افزاره باید داده اولین تراکنش <i>IN</i> را تکرار کند یا تراکنش <i>OUT</i> دوم باید برای موارد دیگر تصدیق شود.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
<p style="text-align: center;">توضیحات:</p> <p>ذکر این نکته لازم است که رفتار افزاره برای تاخیرهای داخل بسته بین ۱۶ تا ۱۸ برابر زمان بیت مشخص نشده‌است، یعنی رفتار افزاره باید به اجرای این تاخیرها وابسته باشد.</p> <p>باید یادآوری کنیم که آزمون‌ها تحت دو شرایط انجام می‌شوند؛ این شرایط توسط یک بازنشانی افزاره از هم جدا می‌شوند.</p> <p>این آزمون باید برای هر نقطه انتهایی اجرا شده برای افزاره انجام شود</p>				
پیاپی: بستر آزمون USB				

۳-۲-۸ آزمون بازیابی خطا و پروتکل سطح پایین

۱-۳-۲-۸ آشکارسازی خطای کد افزونگی دوره‌ای (CRC)^۱

جدول ۱۴ - آزمون USBCRC1

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBCRC1
برچسب: T_USBCRC1_RE1_8.7.1				
عنوان: بررسی رد شدن یک علامت به همراه یک CRC اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف: ۱. روش اجرایی اتصال افزاره را انجام دهید. ۲. یک درخواست <i>GetDescriptor(Device)</i> با یک خطای CRC در نشانه <i>SETUP</i> ارسال کنید. ۳. <i>GetDescriptor(Device)</i> درست را ارسال کنید و درستی نتایج مورد انتظار را بررسی کنید.				
نتایج مورد انتظار: افزاره نباید هیچگونه دست تکانی را در مرحله راه‌اندازی آسیب دیده، بازگرداند، و تراکنش <i>IN</i> باید با یک <i>STALL</i> رد شود.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات: این آزمون باید حداقل یکبار برای تمام نقاط انتهایی اضافه شده با یک تراکنش مناسب انجام شود.				
پیاده‌سازی: بستر آزمون USB				

جدول ۱۵ – آزمون USB CRC2

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBCRC2
برچسب: T_USBCRC2_RE1_8.7.1				
عنوان: بررسی رد شدن یک علامت به همراه یک CRC اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> با یک خطای <i>CRC</i> در فیلد <i>DATA</i> در بسته داده ارسال کنید.</p> <p>۳. <i>GetDescriptor(Device)</i> درست را ارسال کنید و درستی نتایج مورد انتظار را بررسی کنید.</p>				
نتایج مورد انتظار: افزاره نباید هیچگونه دست‌تکانی را در مرحله راه‌اندازی آسیب دیده بازگرداند، و تراکنش <i>IN</i> باید با یک <i>STALL</i> رد شود.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون باید حداقل یکبار برای تمام نقاط انتهایی اضافه شده با یک تراکنش مناسب انجام شود.				
اجرا: بستر آزمون USB				

۸-۲-۳-۲ خطای PID

جدول ۱۶ – آزمون USBPID1

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBPID1
برچسب: T_USBPID1_RE1_8.7.1				
عنوان: بررسی رد شدن یک بسته نشانه حاوی یک PID اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> با یک خطای <i>PID</i> در نشانه <i>SETUP</i> ارسال کنید (مقدار <i>PID</i> به خوبی قالببندی شده اما وجود ندارد).</p> <p>۳. <i>GetDescriptor(Device)</i> صحیح را ارسال کرده و درستی نتایج مورد انتظار را بررسی کنید.</p> <p>۴. از نقطه ۲ با یک <i>SETUP PID</i>، به طور بد قالببندی شده دوباره شروع کنید.</p> <p>۵. از نقطه ۲ با یک مقدار اشتباه بد قالببندی شده به جای <i>SETUP PID</i> دوباره شروع کنید.</p>				
نتایج مورد انتظار: هر زمان که <i>PID</i> اشتباه باشد، علامت و تراکنش متناظر باید رد شوند.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون باید حداقل یکبار برای تمام نقاط انتهایی اضافه شده با یک تراکنش مناسب انجام شود.				
پیاده‌سازی: بستر آزمون USB				

جدول ۱۷ – آزمون USBPID2

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBPID2
برچسب: T_USBPID2_RE1_8.7.1				
عنوان: بررسی رد شدن یک داده علامت حاوی یک PID اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> با یک خطای PID در نشانه <i>DATA</i> ارسال کنید (مقدار PID به خوبی قالببندی شده اما وجود ندارد).</p> <p>۳. <i>GetDescriptor(Device)</i> صحیح را ارسال کرده و درستی نتایج مورد انتظار را بررسی کنید.</p> <p>۴. از نقطه ۲ با یک <i>DATA</i> PID بد قالببندی شده دوباره شروع کنید.</p> <p>۵. از نقطه ۲ با یک مقدار اشتباه که بد قالببندی شده به جای <i>DATA</i> PID دوباره شروع کنید.</p>				
نتایج مورد انتظار: هر زمان که PID اشتباه باشد، علامت و تراکنش متناظر باید رد شوند.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون باید حداقل یکبار برای تمام نقاط انتهایی اضافه شده با یک تراکنش مناسب انجام شود.				
پیاده‌سازی: بستر آزمون USB				

جدول ۱۸ – آزمون USBPID3

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBPID3
برچسب: T_USBPID3_RE1_8.7.1				
عنوان: بررسی رد شدن یک بسته دست‌تکانی حاوی یک PID اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> با یک خطای PID در نشانه <i>HANDSHAKE</i> اولین تراکنش ورودی مرحله داده ارسال کنید (مقدار PID به خوبی قالب‌بندی شده اما وجود ندارد).</p> <p>۳. <i>GetDescriptor(Device)</i> صحیح را ارسال کرده و درستی نتایج مورد انتظار را بررسی کنید.</p> <p>۴. از نقطه ۲ با یک <i>HANDSHAKE</i> PID بد قالب‌بندی شده دوباره شروع کنید.</p> <p>۵. از نقطه ۲ با یک مقدار اشتباه بد قالب‌بندی شده به جای <i>HANDSHAKE</i> PID دوباره شروع کنید.</p>				
<p>نتایج مورد انتظار: هر زمان که PID اشتباه باشد، علامت و تراکنش متناظر باید رد شوند، به این معنی که دومین تراکنش IN مرحله وضعیت (که اگر اندازه داده از ۱۸ بیشتر بود، باید اضافه شود) باید حاوی اطلاعات اولی باشد.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
<p>توضیحات:</p> <p>این آزمون باید حداقل یکبار برای تمام نقاط انتهایی اضافه شده با یک تراکنش مناسب انجام شود.</p>				
پیاده‌سازی: بستر آزمون USB				

۳-۳-۲-۸ آشکارسازی خطای تغییر داده

جدول ۱۹ – آزمون USBDataToggle

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBDataToggle
برچسب: T_USBDataToggle_RE1_8.6				
عنوان: بررسی درستی مدیریت تعویض داده در افزاره				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> با یک نشانه DATA1 برای بسته داده به جای یک DATA0 ارسال کنید.</p>				
نتایج مورد انتظار: درخواست رد می‌شود و نه داده و نه دست‌تکانی نباید توسط افزاره بازگردانده شوند.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون همچنین باید حداقل یکبار برای تمامی نقاط پایانی اضافه شده با یک تراکنش مناسب در صورت امکان انجام شود (فقط برای نقاط پایانی OUT).				
پیاده‌سازی: بستر آزمون USB				

۸-۲-۳-۴ تشخیص خطای آدرس

جدول ۲۰ - آزمون USBAddress

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBAddress
برچسب: T_USBAddress_RE1_8.7.1				
عنوان: بررسی رد شدن یک تراکنش با آدرس اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> به آدرس‌های متفاوت ارسال کنید (از 0x00 تا 0xEF با گام افزایش ۵).</p> <p>۳. یک <i>SetAddress(0x5A)</i> ارسال کنید.</p> <p>۴. از نقطه ۲ با یک آدرس دیگر در محدوده بیان شده دوباره شروع کنید.</p>				
نتایج مورد انتظار: <i>GetDescriptor()</i> باید هر بار که آدرس اشتباه است، بطور کامل رد شود.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون همچنین باید حداقل یکبار برای تمامی نقاط پایانی اضافه شده با یک تراکنش مناسب در صورت امکان انجام شود (فقط برای نقاط پایانی خروجی).				
پیاده‌سازی: بستر آزمون USB				

۸-۲-۳-۵ خطای نقطه‌ی پایانی

جدول ۲۱ – آزمون USBEndP

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	SPEndP
برچسب: T_USBEndP_RE1_8.7.1				
عنوان: بررسی رد شدن یک تراکنش با نقطه‌ی پایانی اشتباه				
شرط آزمون: آزمون باید برای تمام نقاط پایانی موجود انجام شود.				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. یک درخواست <i>GetDescriptor(Device)</i> به نقاط پایانی متفاوت ارسال کنید (از 0x00 تا 0x0F و از 0x80 تا 0x8F با گام افزایش ۱).</p>				
نتایج مورد انتظار: <i>GetDescriptor()</i> باید هر بار که نقطه‌ی پایانی اشتباه است، بطور کامل رد شود.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
این آزمون همچنین باید حداقل یکبار برای تمامی نقاط پایانی اضافه شده با یک تراکنش مناسب در صورت امکان انجام شود (فقط برای نقاط پایانی خروجی).				
پیاده‌سازی: بستر آزمون USB				

۸-۲-۴ آزمون‌های لغو و رهایش
 ۸-۲-۴-۱ رسیدگی درخواست لغو

جدول ۲۲ – آزمون USBCancel

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBCancel
برچسب: T_USBCancel_RE1_5.5				
عنوان: بررسی مدیریت درست درخواست لغو توسط افزاره				
شرط آزمون: NA				
توصیف: ۱. روند اتصال افزاره را انجام دهید. ۲. دو درخواست پی در پی (<i>GetDescriptor(Device)</i>) ارسال کنید که باعث می شود مرحله راه اندازی دومی قبل از مرحله وضعیت اولی آغاز شود.				
نتایج مورد انتظار: دومین (<i>GetDescriptor()</i>) باید به درستی به کار گرفته شود.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده سازی: بستر آزمون USB				

۸-۲-۴-۲ رسیدگی درخواست رهایش

جدول ۲۳ - آزمون USBAbandon

سند: RE2	بحرائی بودن: C	رده: C	نوع: P	USBAbandon
برچسب: T_USBAbandon_RE1_5.5				
عنوان: بررسی مدیریت درست درخواست واگذاری توسط افزاره				
شرط آزمون: NA				
توصیف: ۱. روش اجرایی اتصال افزاره را انجام دهید. ۲. دو درخواست پی در پی (<i>GetDescriptor(Device)</i>) ارسال کنید، اولی کوتاه می شود به جهت این که مرحله وضعیت بلافاصله بعد از مرحله راه اندازی ارسال می شود				
نتایج مورد انتظار: افزاره باید مرحله وضعیت اولی را قبول کند و مرحله دومی رفتار درستی از خود نشان دهد.				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده سازی: بستر آزمون USB				

۵-۲-۸ ماشین حالت USB^۱

جدول ۲۴ – آزمون USBStdReq4

USBSStdReq4	نوع: P	رده: C	بحرانی بودن: C	سند: RE2
برچسب: T_USBSStdReq4_RE1_9.1				
عنوان: بررسی مدیریت درست وضعیت‌های قراردادی، آدرس‌دهی شده و پیکربندی شده USB				
شرط آزمون: NA				
توصیف: ۱. روش اجرایی اتصال افزاره را انجام دهید. ۲. درخواست‌های استفاده شده در این آزمون <i>SetAddress()</i> و <i>SetConfiguration()</i> هستند. مراحل گذر (وضعیت‌های قراردادی، آدرس‌دهی شده و پیکربندی شده USB) شرح داده شده در فصل ۹-۱ باید آزمون شوند.				
نتایج مورد انتظار: هر درخواست باید قبول شود و مرحله گذر باید با استفاده از سازوکار مناسب تصدیق شود (برای مثال رد کردن یک <i>SetConfiguration(0x01)</i> برای یک افزاره در وضعیت قراردادی).				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

۳-۸ آزمون‌های مدیریت درخواست استاندارد

۱-۳-۸ اهداف

هدف این آزمون تایید پشتیبانی درست افزاره از تمام درخواست‌های استاندارد است.

۲-۳-۸ الزامات

تصدیق پردازش درست هر درخواست استاندارد در موارد نامی، امری ضروری است (مواردی که پشتیبانی نمی‌شوند باید توسط افزاره و دستور STALL اعمال شوند).

۳-۳-۸ شرایط ابتدایی

قبل از هر آزمون هر درخواست، کارت (یا تقلید کننده) در وضعیت USB مناسب است (وضعیت قراردادی برای یک *GetDescriptor*، وضعیت آدرس‌دهی شده برای یک *SetConfiguration* ...).

۴-۳-۸ موارد نامی

شرایط آزمونی رایج:

جدول ۲۵ – آزمون USBStdReq1

سند: RE2	بحرانی بودن: C	رده: C	نوع: P	USBStdReq1
برچسب: T_USBStdReq1_RE1_9.4x				
عنوان: بررسی مدیریت درست تمامی درخواست‌های استاندارد				
شرط آزمون: NA				
توصیف:				
<p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. درخواست‌های استاندارد و پارامترهای متناظر در جدول‌های ۳-۹ تا ۶-۹ فهرست شده‌اند. ماشین حالت باید مطابق این آزمون‌ها عمل کند.</p>				
<p>نتایج مورد انتظار: هر درخواست باید به طور درست مورد استفاده قرار گیرد (وقتی توسط افزاره پشتیبانی می‌شود) یا با یک STALL رد شود (وقتی که توسط افزاره پشتیبانی نمی‌شود). باید به توصیف‌گرهای برگشته از افزاره توجه ویژه‌ای داشت و آن‌ها را در مقابل توصیف‌گرهایی که از تعریف افزاره در پایگاه آزمون USB نتیجه می‌شود، بررسی کرد.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
<p>توضیحات: باید یادآوری کرد که بیشتر درخواست‌های استاندارد بعد از مرحله شمارش قابل استفاده باقی خواهند ماند. این درخواست‌ها باید زمانی که افزاره در عملکرد عادی قرار دارد، آزمون شوند.</p>				
پیاده‌سازی: بستر آزمون USB				

۵-۳-۸ فهرست آزمون

جدول ۲۶ - فهرست آزمون

T_USBStdReq1_RE2_9.4.5a	آزمون درخواست GetStatus (افزاره)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.5b	آزمون درخواست GetStatus (نقطه پایانی)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.5c	آزمون درخواست GetStatus (واسط)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.1	آزمون درخواست ClearFeature (برای افزاره، نقطه پایانی، واسط)	درخواست‌های ClearFeatures متوقف شده است
T_USBStdReq1_RE2_9.4.9a	آزمون درخواست SetFeature (افزاره)	متوقف شده است
T_USBStdReq1_RE2_9.4.9b	آزمون درخواست SetFeature (نقطه پایانی)	متوقف شده است
T_USBStdReq1_RE2_9.4.9c	آزمون درخواست SetFeature (واسط)	متوقف شده است
T_USBStdReq1_RE2_9.4.6	آزمون درخواست SetAddress ()	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.3a	آزمون درخواست GetDescriptor (افزاره)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.3b	آزمون درخواست GetDescriptor (پیکربندی)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.3c	آزمون درخواست GetDescriptor (USB-ICC) با یک کارت منطبق با پروتکل ISO7816-12	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.3d	آزمون درخواست GetDescriptor (رشته)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.8a	آزمون درخواست SetDescriptor (افزاره)	متوقف شده است
T_USBStdReq1_RE2_9.4.8b	آزمون درخواست SetDescriptor (پیکربندی)	متوقف شده است
T_USBStdReq1_RE2_9.4.8c	آزمون درخواست SetDescriptor (رشته)	متوقف شده است
T_USBStdReq1_RE2_9.4.8d	آزمون درخواست SetDescriptor	متوقف شده است

	(واسط)	
T_USBStdReq1_RE2_9.4.2	آزمون درخواست GetConfiguration ()	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.7	آزمون درخواست SetConfiguration ()	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.4	آزمون درخواست GetInterface ()	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.10a	آزمون درخواست SetInterface (واسط درست)	به درستی رفتار کرده است
T_USBStdReq1_RE2_9.4.10b	آزمون درخواست SetInterface (واسط نامعلوم)	متوقف شده است
T_USBStdReq1_RE2_9.4.11	آزمون درخواست SynchFrame ()	متوقف شده است

۶-۳-۸ موارد خطا

برای این آزمون‌ها، هر درخواست استاندارد به تنهایی، به USB-ICC همراه با موارد زیر ارسال می‌شود:

- پارامتر wIndex متناقض؛
- پارامتر wValue متناقض؛
- هر دو پارامتر wValue و wIndex متناقض؛
- مقدار wLength کمتر از مقدار واقعی داده منتقل شده؛
- مقدار wLength بیشتر از مقدار منتقل شده؛
- خطا در جهت انتقال داده (در نوع bmRequest)؛
- خطا در گیرنده (در نوع bmRequest).

ذکر این نکته لازم است که برای برخی از درخواست‌های استاندارد، رفتار توسط [RE2] تعریف نشده است و نتایج مورد انتظار به اجرای کارت وابسته است.

۷-۳-۸ شرایط عمومی آزمون

جدول ۲۷ – آزمون USBStdReq2

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBStdReq2
برچسب: T_USBStdReq2_RE1_9.4x				
<p>عنوان: بررسی رد شدن درست درخواست‌های استاندارد زمانی که با پارامترهای اشتباه ارسال می‌شوند.</p>				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. درخواست‌های استاندارد و پارامترهای متناظر در جدول‌های ۹-۳ تا ۹-۶ فهرست شده‌اند. ماشین حالت باید مطابق این آزمون‌ها عمل کند. پارامترها باید تا حد امکان نزدیک به مقادیر درست انتخاب شوند.</p>				
<p>نتایج مورد انتظار: هر درخواست باید بدون توجه به این که توسط افزاره پشتیبانی می‌شود یا نه، رد شود.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

جدول ۲۸ - آزمون USBStdReq3

سند: RE2	بحرانی بودن: C	رده: C	نوع: N	USBStdReq3
برچسب: T_USBStdReq3_RE1_9.4				
عنوان: بررسی رد کردن درخواست‌های استاندارد خارج از زمینه توسط افزاره				
شرط آزمون: NA				
<p>توصیف:</p> <p>۱. روش اجرایی اتصال افزاره را انجام دهید.</p> <p>۲. درخواست‌های استاندارد و پارامترهای متناظر در جداول ۳-۹ تا ۶-۹ فهرست شده‌اند. ماشین حالت باید مطابق این آزمون‌ها عمل کند. پارامترها باید تا حد امکان نزدیک به مقادیر درست انتخاب شوند.</p>				
<p>نتایج مورد انتظار: هر درخواست باید بدون توجه به این که توسط افزاره پشتیبانی می‌شود یا نه، رد شود.</p>				
زمینه ابتدایی: افزاره باید قطع باشد.				
توضیحات:				
پیاده‌سازی: بستر آزمون USB				

۸-۳-۸ فهرست آزمون

جدول ۲۹ - فهرست آزمون

مرجع آزمون	سناریوی آزمون	خطا در درخواست	نتیجه مورد انتظار	رفتار مشخص شده در [RE2]
ET_USBStdReq1_RE2_9.4.5a	GetStatus()	خطا در wValue	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.5b		خطا در wIndex	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.5c		خطا در wIndex و wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.5a		wLength کمتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.5b		wLength بیشتر	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.5c		خطا در جهت انتقال داده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.5a		خطا در گیرنده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.6a	SetAddress()	خطا در wValue	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.6b		خطا در wIndex	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.6c		خطا در wIndex و wValue	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.6d		wLength بیشتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.6e		خطا در جهت انتقال داده	متوقف شده است	بلی

ET_USBStdReq1_RE2_9.4.3a	GetDescriptor()	خطا در wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.3b		خطا در wIndex	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.3c		خطا در wIndex و wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.3d		wLength کمتر	به درستی رفتار کرده است	بلی
ET_USBStdReq1_RE2_9.4.3e		wLength بیشتر	به درستی رفتار کرده است	بلی
ET_USBStdReq1_RE2_9.4.3f		خطا در جهت انتقال داده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.2a	GetConfiguration()	خطا در wValue	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.2b		خطا در wIndex	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.2c		خطا در wIndex و wValue	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.2d		wLength کمتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.2e		wLength بیشتر	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.2f		خطا در جهت انتقال داده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.7a	SetConfiguration()	خطا در wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.7b		خطا در wIndex	به درستی رفتار کرده است	خیر

ET_USBStdReq1_RE2_9.4.7c		خطا در wIndex و wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.7d		wLength بیشتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.7e		خطا در جهت انتقال داده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.4a	GetInterface()	خطا در wValue	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.4b		خطا در wIndex	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.4c		خطا در wIndex و wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.4d		wLength کمتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.4e		wLength بیشتر	به درستی رفتار کرده است	خیر
ET_USBStdReq1_RE2_9.4.4f		خطا در جهت انتقال داده	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.10a	SetInterface()	خطا در wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.10b		خطا در wIndex	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.10c		خطا در wIndex و wValue	متوقف شده است	بلی
ET_USBStdReq1_RE2_9.4.10d		wLength بیشتر	متوقف شده است	خیر
ET_USBStdReq1_RE2_9.4.10e		خطا در جهت انتقال داده	متوقف شده است	بلی

۹ آزمون برای تعیین انطباق با رده CCID

نکته مهم - در این استاندارد ملی فقط USB-ICC در مود نسخه A انتقال کنترل را آدرس دهی می‌کند. ویرایش ویژگی‌ها می‌تواند دیگر مودهای انتقال مشخص شده توسط استاندارد ملی شماره ۷۸۱۶-۱۲ را اضافه کند.

۱-۹ اهداف

اثبات انطباق USB-ICC در مود طبیعی عملکرد آن برای استفاده از پیش تعیین شده:

(۱) شمارش در انطباق با USB-ICC باید از توصیف‌گرهای کارت استاندارد ملی شماره ۷۸۱۶-۱۲ در بند ۷، استفاده کند.

(۲) عملکرد مناسب مجموعه دستورات استاندارد ملی شماره ۷۸۱۶-۱۲، برای مدهای انتقال داده در استاندارد ملی شماره ۷۸۱۶-۱۲ در بند ۸ با کاربرد، شرح داده شده است.

- یک سری از آزمون‌های مطابقت با درخواست‌های خاص USB باید بر طبق پیکربندی قرارداد USB استاندارد ملی ۷۸۱۶-۱۲ باشد.

- یک سری از آزمون‌های خاص پیرو اشکالات تاریخی، در تراشه‌های USB پیدا می‌شوند.

(۳) اجرای درست روند تبادلات APDU باید بر طبق ماشین وضعیت استاندارد بیان شده در استاندارد ملی شماره ۷۸۱۶-۱۲ باشد.

تبادل داده بین میزبان و USB-ICC با استفاده از انتقال توده‌ای یا انتقال‌های کنترل انجام می‌شود. برای انتقال کنترل، دو راه انجام ممکن است. این راه‌ها توسط استاندارد ملی شماره ۷۸۱۶-۱۲ نسخه A و B نام-گذاری شده‌اند. مود انتقال توده‌ای منطبق با مشخصات CCID است، از یک زیرمجموعه از پیام‌ها/درخواست‌ها همانطور که در این مشخصات تعریف شده‌اند، استفاده می‌کند. سناریوهای ارایه شده در این سند مطابق قوانین زیر می‌باشند:

(۱) پردازش درخواست A انتقال کنترل، مطابق استاندارد ملی شماره ۷۸۱۶-۱۲ بند ۸. کارایی USB-ICC در دو مورد آزمون می‌شود: مورد نامی و خطاهای معرفی شده در پارامترهای درخواست و تصدیق این امر که کارت آنها را تشخیص داده و مطابق RE2 عمل کند.

(۲) تصدیق انطباق ماشین حالت: USB-ICC یک مجموعه کامل از درخواست‌های از پیش تعیین شده را ارسال می‌کند تا تصدیق کند که آیا کارت وضعیت داخلی خود را مطابق انتقال‌های تعریف شده توسط نمودارهای وضعیت بخش ۸-۱ تغییر می‌دهد یا خیر.

۲-۹ کنترل یک آزمون درخواست انتقال

۱-۲-۹ شرح آزمون

آزمون هر درخواست استاندارد ملی شماره ۷۸۱۶-۱۲ در وضعیت نامی و وضعیت با خطا در پارامترهای درخواست.

۲-۲-۹ اهداف

(۱) برای تصدیق این امر که افزاره تحت آزمون درخواست‌های USB را مطابق استاندارد ملی شماره ۷۸۱۶-۱۲ بخش ۸-۲-۱، پشتیبانی می‌کند.

۲) برای تصدیق توانایی تشخیص خطا توسط افزاره تحت آزمون زمانی که پارامترهای آزمون اشتباه هستند.

۳-۲-۹ الزامات

وضعیت نامی درخواست‌های USB استاندارد ملی شماره ۷۸۱۶-۱۲ باید درست پردازش شوند. درخواست‌های USB استاندارد ملی شماره ۷۸۱۶-۱۲ که دارای خطا در پارامترها هستند، طبق قواعد زیر پردازش می‌شوند:

۱) مرحله SETUP همواره تایید شده‌است، مگر این‌که یک خطای بسته وجود داشته باشد (بسته‌های SETUP یا DATA0). در این مورد USB-ICC پاسخ نمی‌دهد و میزبان یک دستور RETRY را انجام می‌دهد.

۲) یک خطای پروتکل USB در طول مرحله داده، باید پاسخ ندادن USB-ICC را نتیجه شود. گرچه، کارت ممکن است یکی از دستورات STALL یا NACK یا تراکنش DATA STAGE IN را اجرا کند.

۳) در جدول قبل، «متوقف شده است» به موقعیتی اطلاق می‌شود که نقطه پایانی USB-ICC یک STALL را که مرحله STATUS انتقال داده را دست‌تکانی می‌کند، باز می‌گرداند:

- مانند بسته دست‌تکانی برای یک مرحله OUT STATUS

- بلافاصله بعد از نشانه IN برای یک مرحله IN STATUS

۴-۲-۹ شرایط ابتدایی USB-ICC

افزاره در وضعیت قراردادی USB است.

۵-۲-۹ راه‌اندازی آزمون

این آزمون باید با تقلید کننده و کارت نمونه انجام شود. این آزمون باید با تقلید کننده یا کارت نمونه پیکربندی شده در پروتکل USB، استاندارد ملی شماره ۷۸۱۶-۱۲ اجرا شود. (مطابق بند ۲)

۶-۲-۹ فهرست آزمون: درخواست‌های منحصر بفرد در موارد نامی و دارای خطا

جدول ۳۰ - فهرست آزمون

به درستی رفتار کرده است	مورد نامی	test_icc_power_off_1	Icc_power_off request
متوقف شده است	خطا در wValue	test_icc_power_off_2	
متوقف شده است	خطا در wIndex	test_icc_power_off_3	
متوقف شده است	خطا در wIndex و wValue	test_icc_power_off_4	
متوقف شده است	خطا در جهت انتقال داده	test_icc_power_off_5	
متوقف شده است	wLength بیشتر	test_icc_power_off_6	
متوقف شده است	نوع bmRequest به عنوان	test_icc_power_off_9	

درخواست استاندارد			
Icc_power_on request	test_icc_power_on_1	مورد نامی	به درستی رفتار کرده است
	test_icc_power_on_2	خطا در wValue	متوقف شده است
	test_icc_power_on_3	خطا در wIndex	متوقف شده است
	test_icc_power_on_4	خطا در wIndex و wValue	متوقف شده است
	test_icc_power_on_5	خطا در جهت انتقال داده	متوقف شده است
	test_icc_power_on_7	wLength کمتر	به درستی رفتار کرده است
	test_icc_power_on_8	wLength=0	متوقف شده است
	test_icc_power_on_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است
Data_Block Request	test_data_block_A_1	مورد نامی	به درستی رفتار کرده است
(مورد کلمه وضعیت)	test_data_block_A_2	خطا در wValue	متوقف شده است
	test_data_block_A_3	خطا در wIndex	متوقف شده است
	test_data_block_A_4	خطا در wIndex و wValue	متوقف شده است
	test_data_block_A_5	خطا در جهت انتقال داده	متوقف شده است
	test_data_block_A_6	wLength بیشتر	متوقف شده است یا ارسال SW
	test_data_block_A_7	wLength کمتر	متوقف شده است
	test_data_block_A_8	wLength=0	متوقف شده است
	test_data_block_A_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است
Data_Block	test_data_block_B_1	مورد نامی	به درستی رفتار کرده

request			است
(مورد داده)	test_data_block_B_2	خطا در wValue	متوقف شده است
	test_data_block_B_3	خطا در wIndex	متوقف شده است
	test_data_block_B_4	خطا در wIndex و wValue	متوقف شده است
	test_data_block_B_5	خطا در جهت انتقال داده	متوقف شده است
	test_data_block_B_6	wLength بیشتر	متوقف شده است
	test_data_block_B_7	wLength کمتر	ارسال بسته با طول کوتاه یا صفر
	test_data_block_B_8	wLength = 0	متوقف شده است
	test_data_block_B_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است
Get_Icc_Status request	test_get_icc_status_1	مورد نامی	به درستی رفتار کرده است
	test_get_icc_status_2	خطا در wValue	متوقف شده است
	test_get_icc_status_3	خطا در wIndex	متوقف شده است
	test_get_icc_status_4	خطا در wIndex و wValue	متوقف شده است
	test_get_icc_status_5	خطا در جهت انتقال داده	متوقف شده است
	test_get_icc_status_6	wLength بیشتر	متوقف شده است
	test_get_icc_status_7	wLength کمتر	متوقف شده است
	test_get_icc_status_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است
Xfr_block request	test_xfr_block_A_1	مورد نامی	به درستی رفتار کرده است
(مورد APDU)	test_xfr_block_A_2	خطا در wValue	متوقف شده است

	test_xfr_block_A_3	خطا در wIndex	متوقف شده است
	test_xfr_block_A_4	خطا در wIndex و wValue	متوقف شده است
	test_xfr_block_A_5	خطا در جهت انتقال داده	متوقف شده است
	test_xfr_block_A_6	wLength بیشتر	متوقف شده است
	test_xfr_block_A_7	wLength کمتر	متوقف شده است
	test_xfr_block_A_8	wLength=0	متوقف شده است
	test_xfr_block_A_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است
Xfr_block request	test_xfr_block_B_1	مورد نامی	به درستی رفتار کرده است
(مورد داده)	test_xfr_block_B_2	خطا در wValue	متوقف شده است
	test_xfr_block_B_3	خطا در wIndex	متوقف شده است
	test_xfr_block_B_4	خطا در wIndex و wValue	متوقف شده است
	test_xfr_block_B_5	خطا در جهت انتقال داده	متوقف شده است
	test_xfr_block_B_8	wLength=0	متوقف شده است
	test_xfr_block_B_9	نوع bmRequest به عنوان درخواست استاندارد	متوقف شده است

۳-۹ کنترل یک آزمون نمودار وضعیت انتقال

۱-۳-۹ اهداف

برای تصدیق این امر که USB-ICC، که پروتکل USB، استاندارد ملی شماره ۷۸۱۶-۱۲ را پشتیبانی می‌کند، همان طور که در بخش‌ها تعریف شده‌است، با نمودار وضعیت مطابقت داشته باشد.

۲-۳-۹ ملزومات

تمام درخواست‌های بدون انتظار باید متوقف شوند. تمام درخواست‌های دارای انتظار باید به درستی بکار روند.

۳-۳-۹ شرایط ابتدایی

افزاره در وضعیت قرار دادی USB، بدون قدرت قبل از هر آزمون، قرار دارد.

۹-۳-۴ نوع آزمون

این آزمون باید با تقلیدکننده و کارت نمونه انجام شود. این آزمون باید با یک کارت یا یک تقلیدکننده پیکربندی شده در پروتکل USB استاندارد ملی شماره ۱۲-۷۸۱۶ گذرانده شود.

۹-۳-۵ شرح آزمون

هر آزمون نمودار وضعیت شامل دو مرحله مختلف است:

(۱) مرحله اول تحلیل بازنشانی USB بعد از جاگذاری کارت به متصل کننده است. برای این مرحله، بررسی موارد زیر (بعد از اولین بازنشانی USB) لازم است:

- یک درخواست SetConfiguration() با پیکربندی 1 و آدرس 0، STALL می‌شود؛
 - یک درخواست GetDescriptor() با آدرس 0 پذیرفته شده و به درستی به کار می‌رود؛
 - یک SetAddress() با هر آدرس متفاوت از 0 پذیرفته می‌شود؛
 - یک درخواست SetConfiguration() با آدرس تعریف شده و پیکربندی 1 پذیرفته می‌شود.
- در تمامی موارد، رده افزاره کارت هوشمند از توصیف‌گر مخصوص رده، همانطور که در RE5 و استاندارد ISO/IEC 7816-12 بند ۷-۲ شرح داده شد، استفاده می‌کند.
- (۲) در مرحله دوم (کارت در وضعیت تعریف شده توسط مرحله اول است)، هدف تصدیق آغازسازی دوباره واسط مخابره USB با یک بازنشانی USB و بدون آغازسازی دوباره وضعیت APDU می‌باشد. آزمون‌ها به شرح زیر می‌باشند:

- یک درخواست XFR_BLOCK را برای ارسال یک APDU اجرا کنید (برای مثال SELECT)؛
- یک بازنشانی USB را اجرا کنید؛
- یک درخواست GET_ICC_STATUS با آدرس از قبل تعریف شده قابل شناسایی نخواهد بود (افزاره پاسخ نمی‌دهد)؛
- یک GetDescriptor() با آدرس تعریف شده در درخواست SetAddress() قبلی پذیرفته نخواهد شد (افزاره پاسخ نمی‌دهد)؛
- یک SetConfiguration() با آدرس 0، STALL می‌شود؛
- یک SetAddress با هر آدرس به جز 0 پذیرفته می‌شود؛
- یک درخواست SetConfiguration() با آدرس تعریف شده و پیکربندی 1 پذیرفته می‌شود؛
- یک درخواست GET_ICC_STATUS با آدرس تعریف شده پذیرفته می‌شود؛
- مابقی عملیات دستورات APDU (GET_ICC_STATUS، XFR_BLOCK، DATA_BLOCK) با آدرس درست پذیرفته می‌شوند.

۹-۳-۶ نسخه A انتقال کنترل: فهرست آزمون

۹-۳-۶-۱ آزمون ۱

- 1) USB Reset
- 2) Get_Descriptor: Device Type
- 3) USB Reset
- 4) Set_Address: new address 3
- 5) Get_Descriptor: device type

- 6) Get_Descriptor: configuration type with length = 9
- 7) Get_Descriptor: configuration type with length = 255
- 8) Get_Descriptor: string type
- 9) Get_Descriptor: string type
- 10) Get_Descriptor: string type
- 11) Get_Descriptor: string type
- 12) Get_Descriptor: string type
- 13) Get_Descriptor: device type
- 14) Get_Descriptor: configuration type with length = 9
- 15) Get_Descriptor: configuration type with length = 255
- 16) Icc_power_Off: stalled (device not in USB configured state)
- 17) Icc_power_On: stalled (device not in USB configured state)
- 18) Get_Icc_Status: stalled (device not in USB configured state)
- 19) Xfr_Block: stalled (device not in USB configured state)
- 20) Data_Block: stalled (device not in USB configured state)
- 21) Set_Configuration
- 22) Get_Icc_Status: stalled (the device wait a Icc_power_Off request)
- 23) Xfr_Block: stalled (the device wait a Icc_power_Off request)
- 24) Data_Block: stalled (the device wait a Icc_power_Off request)
- 25) Icc_power_Off: treated correct
- 26) Get_Icc_Status: treated correctly: Busy 1
- 27) Xfr_Block: stalled (incorrect request in this case)
- 28) Data_Block: stalled (incorrect request in this case)
- 29) Get_Icc_Status: treated correctly: Busy 2
- 30) Get_Icc_Status: treated correctly: Busy 3
- 31) Get_Icc_Status: treated correctly: Busy 4
- 32) Get_Icc_Status: treated correctly: Busy 5
- 33) Get_Icc_Status: treated correctly: Busy 6
- 34) Get_Icc_Status: treated correctly: Busy 7
- 35) Get_Icc_Status: treated correctly: Busy 8
- 36) Get_Icc_Status: treated correctly: Busy 9
- 37) Get_Icc_Status: treated correctly: Busy A
- 38) Get_Icc_Status: treated correctly: Busy B
- 39) Get_Icc_Status: treated correctly: Busy C
- 40) Get_Icc_Status: treated correctly: Busy D
- 41) Get_Icc_Status: treated correctly: Busy E
- 42) Get_Icc_Status: treated correctly: Busy F
- 43) Get_Icc_Status: treated correctly: Busy 0
- 44) Get_Icc_Status: treated correctly: Busy 1
- 45) Get_Icc_Status: treated correctly: Busy 2
- 46) Get_Icc_Status: treated correctly: Data Transfer Phase
- 47) Xfr_Block: stalled (incorrect request in this case)
- 48) Data_Block: stalled (incorrect request in this case)
- 49) Icc_power_On: treated correctly
- 50) Icc_power_On: stalled (incorrect request in this case)
- 51) Get_Icc_Status: treated correctly: Waiting for an APDU command
- 52) Xfr_Block: stalled (incorrect request in this case)
- 53) Data_Block: stalled (incorrect request in this case)
- 54) Xfr_Block: treated correctly

- 55) Xfr_Block: stalled (incorrect request in this case)
- 56) Get_Icc_Status: treated correctly: Busy 1
- 57) Icc_power_On: stalled (incorrect request in this case)
- 58) Data_Block: stalled (incorrect request in this case)
- 59) Get_Icc_Status: treated correctly: Data Transfer Phase
- 60) Icc_power_On: stalled (incorrect request in this case)
- 61) Data_Block: stalled (incorrect request in this case)
- 62) Xfr_Block: treated correctly
- 63) Data_Block: stalled (incorrect request in this case)
- 64) Get_Icc_Status: treated correctly: Busy 1
- 65) Icc_power_On: stalled (incorrect request in this case)
- 66) Xfr_Block: stalled (incorrect request in this case)
- 67) Get_Icc_Status: treated correctly: Status Word Phase
- 68) Icc_power_On: stalled (incorrect request in this case)
- 69) Xfr_Block: stalled (incorrect request in this case)
- 70) Data_Block: treated correctly
- 71) Get_Icc_Status: treated correctly: Waiting for an APDU command
- 72) Xfr_Block: treated correctly
- 73) Icc_power_On: stalled (incorrect request in this case)
- 74) Xfr_Block: stalled (incorrect request in this case)
- 75) Get_Icc_Status: treated correctly: Data Transfer Phase
- 76) Data_Block: treated correctly
- 77) Get_Icc_Status: treated correctly: Status Word Phase
- 78) Data_Block: treated correctly

۲-۶-۳-۹ آزمون ۲

- 1) USB Reset
- 2) Set_Address: new address 3
- 3) Get_Descriptor: device type
- 4) Set_Configuration
- 5) Icc_power_Off: treated correctly
- 6) Get_Icc_Status: treated correctly: Busy 1
- 7) Get_Icc_Status: treated correctly: Busy 2
- 8) Get_Icc_Status: treated correctly: Busy 3
- 9) Get_Icc_Status: treated correctly: Data Transfer Phase
- 10) Icc_power_On: treated correctly
- 11) Get_Icc_Status: treated correctly: Waiting for an APDU command
- 12) Icc_power_Off: treated correctly
- 13) Get_Icc_Status: treated correctly: Busy 1
- 14) Get_Icc_Status: treated correctly: Data Transfer Phase
- 15) Icc_power_On: treated correctly

۳-۶-۳-۹ آزمون ۳

- 1) USB Reset
- 2) Set_Address: new address 3
- 3) Get_Descriptor: device type
- 4) Set_Configuration
- 5) Icc_power_Off: treated correctly

- 6) Get_Icc_Status: treated correctly: Busy 1
- 7) Get_Icc_Status: treated correctly: Busy 2
- 8) Get_Icc_Status: treated correctly: Busy 3
- 9) Get_Icc_Status: treated correctly: Data Transfer Phase
- 10) Icc_power_On: treated correctly
- 11) Get_Icc_Status: treated correctly: Waiting for an APDU command
- 12) Xfr_Block: treated correctly
- 13) Icc_power_Off: treated correctly
- 14) Get_Icc_Status: treated correctly: Busy 1
- 15) Get_Icc_Status: treated correctly: Data Transfer Phase
- 16) Icc_power_On: treated correctly

۴-۳-۹-۶-۴ آزمون ۴

- 1) USB Reset
- 2) Set_Address: new address 3
- 3) Get_Descriptor: device type
- 4) Set_Configuration
- 5) Icc_power_Off: treated correctly
- 6) Get_Icc_Status: treated correctly: Busy 1
- 7) Get_Icc_Status: treated correctly: Busy 2
- 8) Get_Icc_Status: treated correctly: Busy 3
- 9) Get_Icc_Status: treated correctly: Data Transfer Phase
- 10) Icc_power_On: treated correctly
- 11) Get_Icc_Status: treated correctly: Waiting for an APDU command
- 12) Xfr_Block: treated correctly
- 13) Get_Icc_Status: treated correctly: Busy 1
- 14) Get_Icc_Status: treated correctly: Busy 2
- 15) Get_Icc_Status: treated correctly: Data Transfer Phase
- 16) Xfr_Block: treated correctly
- 17) Get_Icc_Status: treated correctly: Busy 1
- 18) Icc_power_Off: treated correctly
- 19) Get_Icc_Status: treated correctly: Busy 2 (For ISO7816-12V3 implementation)
- 20) Get_Icc_Status: treated correctly: Data Transfer Phase
- 21) Icc_power_On: treated correctly

۵-۳-۹-۶-۵ آزمون ۵

- 1) USB Reset
- 2) Set_Address: new address 3
- 3) Get_Descriptor: device type
- 4) Set_Configuration
- 5) Icc_power_Off: treated correctly

حلقه در اینجا: برای این آزمون، گزینه مود تکرار USB باید آماده شود.

- 6) Get_Icc_Status: treated correctly: Busy X

کتابنامه

[1] *Inter-Chip USB Supplement to the USB 2.0 Specification – Revision 1.0*, 13 March 2006 [RE3 in this part of ISO/IEC 10373]

[2] *Full and Low Speed Electrical and Interoperability Compliance Test Procedure – Revision 1.3*, February 2004 [RE4 in this part of ISO/IEC 10373]

[3] *Universal Serial Bus – Device Class: Smart Card – CCID – Specification for USB Integrated Circuit(s) Cards Interface Devices – Revision 1.1*, 22 April 2005 [RE5 in this part of ISO/IEC 10373]