INSO
9598-3
1st. Edition
Apr.2013

جمهوری اسلامی ایران
Islamic Republic of Iran
سازمان ملی استاندارد ایران
Iranian National Standardization Organization

استاندارد ملی ایران
۹۵۹۸-۳
چاپ اول
اردیبهشت ۱۳۹۲

فناوری اطلاعات- فنون امنیتی - توابع درهم‌ساز
قسمت۳: توابع درهم ساز اختصاصی

Information Technology - Security Techniques- Hash-Functions
Part 3: Dedicated hash-functions

ICS:35.040

به نام خدا

آشنایی با سازمان ملی استاندارد ایران

مؤسسهٔ استاندارد و تحقیقات صنعتی ایران به موجب بند یک مادهٔ۳ قانون اصلاح قوانین و مقررات مؤسسهٔ 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 وکوششی همگام با مصالح ملی و با توجه به شرایط تولیدی، فناوری و تجاری است که از مشارکت آگاهانه و منصفانهٔ صاحبان حق و نفع، شامل تولیدکنندگان، مصرف‌کنندگان، 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 می‌شود و پس از دریافت نظرها و پیشنهادها در کمیتهٔ 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 درکمیتهٔ ملی طرح و بررسی و درصورت 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 شمارهٔ ۵ تدوین و در کمیتهٔ ملی استاندارد مربوط که سازمان ملی استاندارد ایران تشکیل می‌دهد به تصویب رسیده باشد.

سازمان ملی استاندارد ایران از اعضای اصلی سازمان بین المللی استاندارد (ISO)[1] ،کمیسیون بین المللی الکتروتکنیک (IEC)[2] و سازمان بین المللی اندازه شناسی قانونی (OIML)[3] است و به عنوان تنها رابط[4] کمیسیون کدکس غذایی (CAC)[5] 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 بین‌المللی بهره‌گیری می‌شود .

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 استاندارد کالاهای صادراتی و درجه‌بندی آن را اجباری نماید. همچنین برای اطمینان بخشیدن به استفاده کنندگان از خدمات سازمان ها و مؤسسات فعال در زمینهٔ مشاوره، آموزش، بازرسی، ممیزی و صدورگواهی سیستم های مدیریت کیفیت و مدیریت زیست‌محیطی، آزمایشگاه ها و مراکز کالیبراسیون (واسنجی) 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، گواهینامهٔ تأیید صلاحیت به آن ها اعطا و بر عملکرد آن ها نظارت می کند. ترویج دستگاه بین المللی یکاها، کالیبراسیون (واسنجی) 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

«فناوری اطلاعات- فنون امنیتی - توابع درهم‌ساز، قسمت۳: توابع درهم‌ساز اختصاصی»

رئیس :	**سمت و/ یا نمایندگی**
فولادیان، مجید (فوق لیسانس مهندسی برق مخابرات)	مشاور سازمان فناوری اطلاعات
دبیر :	
میراسکندری، سید محمدرضا (لیسانس مهندسی کامپیوتر نرم افزار)	مدیر کل خدمات ارزش افزوده سازمان فناوری اطلاعات
اعضا: (اسامی به ترتیب حروف الفبا)	
امیریان، احسان (فوق لیسانس مهندسی کامپیوتر)	مدیرعامل شرکت هوشمندی تجاری تالی
بختیاری، شیرین (کارشناسی مهندسی برق)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
جمیل پناه، ناصر (کارشناسی ارشد مدیریت)	کارشناس سازمان فناوری اطلاعات
خوشنویسان، نازنین (لیسانس مهندسی نرم‌افزار)	نماینده دانشگاه علم و صنعت
سعیدی، عذرا (فوق لیسانس مهندسی برق مخابرات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
سلطانی حقیقت، الهه (لیسانس مهندسی برق مخابرات)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
عسگرزاده، مجید (فوق لیسانس مهندسی کامپیوتر)	مدیر پروژه موسسه تحقیقات ارتباطات و فناوری اطلاعات

فرهاد شیخ احمد، لیلا (فوق لیسانس مهندسی کامپیوتر نرم افزار)	کارشناس تدوین استاندارد سازمان فنـاوری اطلاعات
فیاضی، مهدی (لیسانس مهندسی برق مخابرات)	کارشناس مسئول تدوین استاندارد و امنیت شبکه سازمان فناوری اطلاعات
قسمتی، سیمین (فوق لیسانس فناوری اطلاعات)	کارشناس تدوین استاندارد سازمان فنـاوری اطلاعات
معروف، سینا (لیسانس مهندسی کامپیوتر)	کارشناس تدوین استاندارد سازمان فناوری اطلاعات
موجبی، محمود (فوق لیسانس مهندسی برق مخابرات)	کارشناس تدوین استاندارد سازمان فنـاوری اطلاعات
میرزایی رضایی، طیبه (فوق لیسانس فیزیک)	رئیس اداره تدوین استانداردها و نظـارت بـر امنیت سرویس‌ها سازمان فناوری اطلاعات
ناظمی، اسلام (دکتری کامپیوتر)	استادیار دانشگاه شهید بهشتی
نیسی مینایی، آصف (لیسانس مهندسی فناوری اطلاعات)	نماینده دانشگاه علم و صنعت

فهرست مندرجات

عنوان **صفحه**

پیش‌گفتار

استاندارد «فناوری اطلاعات- فنون امنیتی - توابع درهم‌ساز، قسمت۳: توابع درهم‌ساز اختصاصی» که پیش‌نویس آن درکمیسیون‌های مربوط توسط سازمان فناوری اطلاعات تهیه و تدوین شده و در دویست و پانزدهمین اجلاس کمیتهٔ ملی استاندارد رایانه و فرآوری داده مورخ ۹۱/۸/۳۰ مورد تصویب قرار گرفته است، اینک به استناد بند یک مادهٔ 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 زمینهٔ 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 تهیهٔ این استاندارد مورد استفاده قرار گرفته به شرح زیر است:

ISO/IEC 10118-3: 2004, Information technology - Security techniques- Dedicated Hash Functions . + Amendment1: 2006 + Technical Corrigendum 1:2011

مقدمه

این استاندارد یکی از مجموعه استانداردهای ملی ایران به شماره ۹۵۹۸ می‌باشد.

فناوری اطلاعات- فنون امنیتی - توابع درهم‌ساز، قسمت۳: توابع درهم‌ساز اختصاصی

۱ هدف و دامنه کاربرد

هدف از تدوین این استاندارد، تعیین ویژگی‌های توابع درهم‌ساز اختصاصی[1] است که به عبارت دیگر توابع درهم‌سازی که به صورت خاص طراحی شده‌اند. در این استاندارد، توابع درهم‌سازی، براساس استفاده تکراری از تابع گردساز[2] هستند. هفت تابع گردساز متمایز مشخص شده‌اند تا امکان تمایز توابع درهم‌ساز اختصاصی را فراهم کنند.

اولین و سومین تابع درهم‌ساز اختصاصی در بندهای ۷ و ۹ به ترتیب کدهای درهم با طول بیشینه ۱۶۰ بیت را فراهم می‌کنند؛ دومین تابع در بند ۸ کدهای درهمی با طول بیشینه ۱۲۸ بیت فراهم می‌کند؛ چهارمین تابع در بند ۱۰ کدهای درهمی با طول بیشینه ۲۵۶ بیت ایجاد می‌کند؛ ششمین تابع در بند ۱۲ کدهای درهمی با طول ثابت ۳۸۴ بیت، و پنجمین و هفتمین تابع در بندهای ۱۱ و ۱۳ به ترتیب کدهای درهم با طول بیشینه ۵۱۲ بیت را فراهم می‌کنند.

۲ مراجع الزامی

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

در صورتی که به مدرکی با ذکر تاریخ انتشار ارجاع 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 شده است، همواره تاریخ تجدید نظر و اصلاحیه‌های بعدی آن‌ها مورد نظر است.

استفاده از مراجع زیر برای این استاندارد الزامی است:

۲-۱ استاندارد ملی شماره ۱-۹۵۹۸: سال ۱۳۸۶[3] ، فناوری اطلاعات – فنون امنیتی - توابع درهم‌ساز - قسمت اول: کلیات

۳ اصطلاحات و تعاریف

در این استاندارد، اصطلاحات و تعاریف زیر به کار می‌رود.

۳-۱

---

1 -Dedicated hash functions

2 - Round Function

[3]- معادل استاندارد بین المللی ISO10118-1:2000

بلوک[1]

رشته- بیتی به طول L_1، به عبارت دیگر طول اولین ورودی تابع گردساز است.

۲-۳

کلمه

رشته‌ای ۳۲ بیتی که در توابع درهم‌ساز اختصاصی ۱، ۲، ۳ و ۴ از بندهای ۷، ۸، ۹ و ۱۰ به ترتیب استفاده می‌شود، یا رشته‌ای ۶۴ بیتی که در توابع درهم‌ساز اختصاصی ۵ و ۶ از بندهای ۱۱ و ۱۲ به ترتیب استفاده می‌شود.

۳-۳

ماتریس

یک ماتریس ۸ در ۸ که در آن هر ورودی، رشته‌ای ۸ بیتی است که در تابع درهم‌ساز اختصاصی ۷ از بند ۱۳ استفاده می‌شود.

۴ نمادها و کوته‌نوشت‌ها

۴-۱ نمادهای مشخص شده در استاندارد ملی شماره ۱-۹۵۹۸: سال ۱۳۸۶

در این استاندارد از نمادها و علائم تعریف شده در استاندارد ملی ۱-۹۵۹۸ : سال ۱۳۸۶ استفاده می‌کند.

B_i یک بایت

D داده

H کد درهم

IV مقدار اولیه

L_1 طول (به بیت) اولین دو رشته‌ی ورودی به تابع گردساز Φ

L_2 طول (به بیت) دومین دو رشته‌ی ورودی به تابع گردساز Φ، از رشته‌ی خروجی تابع گردساز Φ و IV

L_x طول (به بیت) رشته بیتی X

Φ یک تابع گردساز، در واقع اگر X ، Y به ترتیب رشته بیت‌هایی با طول‌های L_1 و L_2 باشند، آنگاه تابع $\Phi(X,Y)$، رشته‌ی حاصل از اعمال Φ به X و Y خواهد بود.

$X \oplus Y$ XOR رشته بیت‌های X و Y (آنجا که $L_x=L_y$).

۴-۲ نمادهای مختص این قسمت از این استاندارد

نمادها و علائم ذیل برای این استاندارد مورد استفاده قرار می‌گیرند.

---

[1] - Block

a_i, a'_i دنباله‌های اندیس‌هایی که در یک تابع گردساز مشخص استفاده می‌شوند.

A^i دنباله‌ای از ماتریس‌های ثابت که در تعیین تابع گردساز بیان شده در بند ۱۳ استفاده می‌شود.

c_0 تابعی که یک رشته‌ی ۶۴ عنصری از میدان[1] $GF(2^8)$ را به عنوان ورودی گرفته و یک ماتریس ۸ در ۸ با ورودی‌هایی از میدان $GF(2^8)$ را به عنوان خروجی می‌دهد، که در تعیین تابع گردساز بیان شده در بند ۱۳، استفاده می‌شود.

c_1,c_2,c_3 توابعی که یک ماتریس ۸ در ۸ از عناصر $GF(2^8)$ را به عنوان ورودی گرفته و یک ماتریس ۸ در ۸ با ورودی‌هایی از $GF(2^8)$ را به عنوان خروجی می‌دهند، که در تعیین تابع گردساز بیان شده در بند ۱۳، استفاده می‌شوند.

c_4 تابعی که دو ماتریس ۸ در ۸ از عناصر $GF(2^8)$ را به عنوان ورودی گرفته و یک ماتریس ۸ در ۸ با ورودی‌هایی از $GF(2^8)$ را به عنوان خروجی می‌دهد، که در مشخص کردن تابع گردساز تعریف شده در بند۱۳، استفاده می‌شود.

C_i, C'_i کلمات ثابتی که در تابع گردساز استفاده می‌شوند.

C'' یک ماتریس گردشی[2] ۸ در ۸ با ورودی‌های انتخابی از $GF(2^8)$ که در تعیین تابع گردساز بیان شده در بند۱۳، استفاده می‌شود.

D_i بلوکی مشتق شده از رشته- داده بعد از فرآیند لایه‌گذاری[3].

$d_i, e_i, f_i,$ توابعی که یک یا سه کلمه به عنوان ورودی گرفته و تنها یک کلمه به عنوان خروجی تولید می‌کنند و درمشخص کردن توابع گردساز استفاده می‌شوند.

H_i رشته‌ای L_2 بیتی که در عملیات درهم‌سازی برای ذخیره‌ی یک نتیجه‌ی میانی استفاده می‌شود.

$GF(2^8)$ میدانی که به صورت $GF(2)[x] / p_8(x)$ تعریف شده که در آن $p_8(x)=x^8+x^4+x^3+x^2+1$. عناصر این میدان رشته‌های ۸ بیتی هستند.

M یک ماتریس ۸ در ۸ که ورودی‌های آن از میدان $GF(2^8)$ انتخاب می‌شوند.

Q تعداد بلوک‌ها در رشته داده پس از فرآیندهای لایه‌گذاری و تقسیم[1].

---

1 - Field
2- Circulant matrix
3 - Padding

$R^n()$	عملیات جابه‌جایی به سمت راست[2] n بیتی، به عبارت دیگر اگر A یک کلمه و n یک عدد صحیح غیرمنفی باشد، آنگاه $R^n(A)$ نشان دهنده‌ی کلمه‌ای است که از جابه-جایی به سمت راست n مکانی محتوای A به‌دست می‌آید.
s	یک جعبه‌ی جایگزینی غیرخطی، که عنصر $x \in GF(2^8)$ را با عنصر دیگر $s[x] \in GF(2^8)$ جایگزین می‌کند.
$S^n()$	عملیات «جابه‌جایی چپ گردشی[3]» با n بیت مکانی، به عبارت دیگر اگر A یک کلمه و n یک عدد صحیح غیرمنفی باشد، آنگاه $S^n(A)$ نشان دهنده‌ی کلمه‌ای است که از جابه‌جایی- چپ *n* مکانی محتوای A به شیوه‌ی چرخشی به‌دست می‌آید.
$S'^n()$	عملیات «جابه‌جایی راست گردشی» با n بیت مکانی، به عبارت دیگر اگر A یک کلمه و *n* یک عدد صحیح غیرمنفی باشد، آنگاه $S^n(A)$ نشان دهنده‌ی کلمه‌ای است که از جابه‌جایی- راست *n* مکانی محتوای A به شیوه‌ی چرخشی به‌دست می‌آید.
t_i , t'_i	مقدارهای جابه‌جایی که در تعیین توابع گردساز استفاده می‌شوند.
W, X_i, X'_i, Y_i, Z_i	کلماتی برای ذخیره‌ی نتایج محاسبات میانی
W', X'', K_i, Y', Z'	ماتریس‌هایی با ورودی‌های انتخابی از میدان $GF(2^8)$ که برای ذخیره‌ی نتایج محاسبات میانی استفاده می‌شوند.
$\wedge$	عملیات AND منطقی بیتی در رشته- بیت‌ها، به عبارت دیگر اگر A و B کلماتی باشند، آنگاه $A \wedge B$ کلمه‌ای معادل عملیات AND منطقی بیتی A و B است.
$\vee$	عملیات OR منطقی بیتی در رشته- بیت‌ها، به عبارت دیگر اگر A و B کلماتی باشند، آنگاه $A \vee B$ کلمه‌ای معادل عملیات OR منطقی بیتی A و B است.
$\neg$	عملیات NOT منطقی بیتی در رشته- بیت‌ها، به عبارت دیگر اگر A کلمه‌ای باشد، آنگاه $\neg A$ کلمه‌ای معادل عملیات NOT منطقی بیتی A است.
$\boxplus$	عملیات جمع به پیمانه[4] 2^w ، که در آن w تعداد بیت‌های یک کلمه است؛ به عبارت دیگر اگر A و B کلماتی باشند، آنگاه $A \boxplus B$ کلمه‌ای است که با در نظر گرفتن A و B به عنوان نمایش دودویی اعداد صحیح و محاسبه‌ی جمع به پیمانه 2^w آنها به دست می‌آید، که در آن، نتیجه باید بین ۰ و $2^w - 1$ (قرار داشته باشد. مقدار W

1 - Splitting
2 - Right Shift
3 - Circular left shift
4 - Modulo

برای توابع درهم‌ساز اختصاصی یک تا چهار، تعریف شده در بند های هفت تا ۱۰، ۳۲ و برای توابع درهم‌ساز اختصاصی پنج و شش، تعریف شده در بندهای ۱۱ و ۱۲، ۶۴ است.

• عملیات ضرب ماتریس‌های ۸ در ۸ با ورودی‌های انتخابی از میدان $GF(2^8)$. به عبارت دیگر اگر A و B ماتریس‌های یکسانی باشند، آنگاه A•B ماتریسی است که با ضرب A در B به طریق زیر به‌دست می‌آید: هر یک از ورودی‌های A و یا B را به عنوان نمایش چندجمله‌ای دودویی یک عدد صحیح در نظر بگیرید (برای مثال نمایش چندجمله‌ای دودویی عدد صحیح ۸۹ (مبنای شانزده) به صورت x^7+x^3+1 است)؛ حاصلضرب دو ورودی را وقتی حاصلضرب دو چندجمله‌ای بر چندجمله‌ای $p_8(x)$ که $p_8(x)=x^8+x^4+x^3+x^2+1$، تقسیم شود، به‌عنوان باقیمانده و عملگرجمع را به عنوان عملگر ⊕ در نظر بگیرید.

:= نماد نشان‌دهنده‌ی عملیات "مساوی ساختن با" که در مشخصات رویه‌ی[1] توابع گردساز استفاده می‌شود و نشان می‌دهد که کلمه (یا در بند ۱۳، ماتریس) سمت چپ این نماد باید برابر با مقدار عبارت سمت راست نماد شود.

۵ الزامات

کاربرانی که می‌خواهند یک تابع درهم‌ساز از این استاندارد به کار گیرند، باید انتخاب کنند:

- یکی از توابع درهم‌ساز اختصاصی مشخص شده در زیر و
- طول L_H از کد درهم H

یادآوری- اولین و دومین تابع درهم‌ساز اختصاصی به منظور تسهیل پیاده‌سازی‌های نرم‌افزاری برای رایانه‌های "little-endian"، که در آنها بایت با کمترین آدرس در یک کلمه به عنوان کم ارزش‌ترین آن تعبیر می‌شود، تعریف می‌شوند؛ برعکس، سومین، چهارمین، و پنجمین و ششمین تابع درهم‌ساز اختصاصی به منظور تسهیل پیاده‌سازی‌های نرم‌افزاری برای رایانه‌های "big-endian" ،که در آنها بایت با کمترین آدرس در یک کلمه به عنوان با ارزش‌ترین آن تعبیر می‌شوند. با این وجود با تطبیق مناسب تعریف، هر یک از این شش تابع گردساز قابل پیاده‌سازی بر روی یک رایانه‌ی "big-endian" و یا "little-endian" خواهد بود. هفتمین تابع درهم‌ساز اختصاصی به این صورت تعریف می‌شود که "endian-neutral" باشد، به صورتی که از هیچ عملیات حسابی حساس به نوع endian (مانند جمع صحیح) استفاده نمی‌کند. اگر دنباله‌های عناصر از میدان $GF(2^8)$ (یا همان بایت‌ها)، به منظور موازی سازی عملیاتی مانند OR انحصاری[2]، به کلمات رایانه-ای نگاشت شوند، تا زمانی که نگاشت معکوس سازگار باشد، آرایش بایتی در یک کلمه مهم نخواهد بود. تمامی توابع درهم‌ساز تعریف شده در این استاندارد، یک رشته- بیت به عنوان ورودی گرفته و یک رشته- بیت به عنوان خروجی می‌دهند؛ این مورد مستقل از قاعده‌ی ترتیب- بایتی مورد استفاده در هر تابع درهم‌ساز است.

1 - Procedural Specifications
2 - Exclusive-or

یادآوری- انتخاب L_H در امنیت تابع درهم‌ساز اثر می‌گذارد. تمامی توابع درهم‌ساز مشخص شده در این استاندارد، مقاوم در برابر برخورد فرض می‌شوند، در فضایی که انجام $2^{L_H/2}$ محاسبه کد درهم‌ساز، از نظر محاسباتی غیرممکن به نظر برسد.

۶ مدلی برای توابع درهم‌ساز اختصاصی

توابع درهم‌ساز مشخص شده در این استاندارد، بر اساس مدل کلی از توابع درهم‌ساز در استاندارد ملی شماره ۱-۹۵۹۸ : سال ۱۳۸۶ هستند.

در مشخصات توابع درهم‌ساز مشخص شده در این استاندارد، فرض شده است که ورودی رشته- داده‌ای لایه‌گذاری شده برای تابع درهم‌ساز، به‌صورت دنباله‌هایی از بایت‌ها است. اگر رشته- داده‌ی لایه‌گذاری شده به‌صورت دنباله‌ی n۸ بیتی، $x_0 , x_1 , \dots , x_{8n-1}$، باشد، آنگاه باید از آن به عنوان دنباله‌ای از n بایت، $B_0 , B_1 , \dots , B_{n-1}$، به صورتی که در ادامه آمده است، تعبیر کرد. هر گروه هشت بیت متوالی به عنوان یک بایت در نظر گرفته می‌شوند، اولین بیت هر گروه با ارزش‌ترین بیت آن بایت است. بنابراین

$B_i=2^7 x_{8i} + 2^6 x_{8i+1} + \dots + x_{8i+7}$

برای هر i $(0 \le i < n)$

تبدیل خروجی برای توابع درهم‌ساز مشخص شده در این استاندارد، مشتق شدن کد درهم‌سازی H از چپ-ترین بیت‌های L_H از L_2 بیت پایانی رشته‌ی خروجی H_q است.

شناساگرهایی برای هر یک از هفت تابع درهم‌ساز اختصاصی مشخص شده در این استاندارد تعریف می‌شوند. شناساگرهای تابع درهم‌ساز برای توابع درهم‌ساز اختصاصی مشخص شده در بندهای ۷، ۸، ۹، ۱۰، ۱۱، ۱۲ و ۱۳ به ترتیب برابر ۳۱، ۳۲، ۳۳، ۳۴، ۳۵، ۳۶ و ۳۷ (در مبنای شانزده) هستند. محدوده‌ی مقادیر از ۳۸ تا ۳F (در مبنای شانزده) برای استفاده‌ی آینده به عنوان شناساگرهای تابع درهم‌ساز، توسط این استاندارد کنار گذاشته می‌شود. شناساگرهای تابع درهم‌ساز همچنین در شناسه‌های شی OSI تعیین شده در پیوست پ استفاده می‌شوند.

۷ تابع درهم‌ساز اختصاصی ۱ (RIPEMD-160)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شمار ۱-۹۵۹۸: سال ۱۳۸۶ توصیف شده است، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، در صورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۱را تعریف می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر ۲^۶۴-۱ بیت، اعمال کرد.

شناساگری تابع درهم‌ساز استاندارد ISO/IEC برای تابع درهم‌ساز اختصاصی ۱ برابر ۳۱ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۱ که در این بند بیان می‌شود، به طور معمول RIPEMD-160 نامیده می‌شود [3].

۷-۱ پارامترها، توابع و ثابت‌ها

۷-۱-۱ پارامترها

برای این تابع درهم‌ساز $L_1=$ ۵۱۲، $L_2=$ ۱۶۰ و L_H بیشینه ۱۶۰ است.

۷-۱-۲ قاعده‌ی مرتب‌سازی بایت

در مشخصات تابع گردساز این بند فرض شده است که بلوک ورودی به تابع گردساز به‌صورت یک دنباله از کلمات ۳۲ بیتی است، هر بلوک ۵۱۲ بیتی از ۱۶ تا از این کلمات ساخته می‌شود. یک دنباله از ۶۴ بایت، $B_0, B_1, \ldots, B_{63}$، باید به عنوان یک دنباله‌ی ۱۶ کلمه‌ای، $Z_0, Z_1, \ldots, Z_{15}$، به صورت زیر تعبیر شود. هر گروه چهار بایتی متوالی به عنوان یک کلمه در نظر گرفته می‌شود. اولین بایت یک کلمه، کم ارزش‌ترین بایت آن است. بنابراین :

$$Z_i = 2^{24}B_{4i+3} + 2^{16}B_{4i+2} + 2^8B_{4i+1} + B_{4i}, \qquad (0 \le i \le 15).$$

برای تبدیل یک کد درهم از یک دنباله از کلمات به یک دنباله بایت، باید معکوس این فرایند دنبال شود.

یادآوری- مرتب‌سازی بایت مشخص شده در اینجا متفاوت از زیربند ۹-۱-۲ است.

۷-۱-۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی کلمات ۳۲ بیتی توصیف می‌شود. دنباله‌ای از توابع $g_0, g_1, \ldots, g_{79}$، در این تابع گردساز استفاده می‌شود، که در آن هر تابع g_i، $0 \le i \le 79$، سه کلمه X_0, X_1 و X_2 را به عنوان ورودی گرفته و تنها یک کلمه را به عنوان خروجی برمی‌گرداند.

توابع g_i به صورت زیر تعریف می‌شوند:

$g_i(X_0,X_1,X_2) = X_0 \oplus X_1 \oplus X_2, \qquad (0 \le i \le 15),$
$g_i(X_0,X_1,X_2) = (X_0 \wedge X_1) \vee (X_0 \wedge X_2), \qquad (16 \le i \le 31),$
$g_i(X_0,X_1,X_2) = (X_0 \vee X_1) \oplus X_2, \qquad (32 \le i \le 47),$
$g_i(X_0,X_1,X_2) = (X_0 \wedge X_2) \vee (X_1 \wedge X_2), \qquad (48 \le i \le 63),$
$g_i(X_0,X_1,X_2) = X_0 \oplus (X_1 \vee X_2), \qquad (64 \le i \le 79).$

۷-۱-۴ ثابت‌ها

دو دنباله از کلمات ثابت $C_0, C_1, \ldots, C_{79}$ و $C'_0, C'_1, \ldots, C'_{79}$ در این تابع گردساز استفاده می‌شود. در یک نمایش در مبنای شانزده (که در آن با ارزش‌ترین بیت متناظر با چپ‌ترین بیت است) این ثابت‌ها به صورت زیر تعریف می‌شوند:

$C_i =$ 00000000, $\qquad (0 \le i \le 15),$
$C_i =$ 5A827999, $\qquad (16 \le i \le 31),$
$C_i =$ 6ED9EBA1, $\qquad (32 \le i \le 47),$
$C_i =$ 8F1BBCDC, $\qquad (48 \le i \le 63),$
$C_i =$ A953FD4E, $\qquad (64 \le i \le 79),$

$C'_i =$ 50A28BE6, $\qquad (0 \le i \le 15),$
$C'_i =$ 5C4DD124, $\qquad (16 \le i \le 31),$
$C'_i =$ 6D703EF3, $\qquad (32 \le i \le 47),$

C'_i = 7A6D76E9, $(48 \le i \le 63)$,
C'_i = 00000000, $(64 \le i \le 79)$.

دو دنباله از ۸۰ مقدار جابه‌جایی در این تابع گردساز استفاده می‌شود، که در آن هر مقدار جابه‌جایی بین ۵ و ۱۵ قرار دارد. این دنباله‌ها را با ($t_0, t_1, \dots, t_{79}$) و ($t'_0, t'_1, \dots, t'_{79}$) نشان می‌دهیم. دو دنباله‌ی دیگر ۸۰ تایی در این تابع گردساز استفاده می‌شوند، که در آن هر مقدار در دنباله بین صفر و ۱۵ قرار دارد. این دنباله‌ها را با ($a_0, a_1, \dots, a_{79}$) و ($a'_0, a'_1, \dots, a'_{79}$) نشان می‌دهیم. هر چهار دنباله در جدول ۱ در زیر بیان شده‌اند.

جدول۱ : استفاده گردسازی دنباله‌ها

7	6	5	4	3	2	1	0	i
9	7	8	5	12	15	14	11	t_i
5	15	15	13	11	9	9	8	t'_i
7	6	5	4	3	2	1	0	a_i
4	11	2	9	0	7	14	5	a'_i

15	14	13	12	11	10	9	8	i
8	9	7	6	15	14	13	11	t_i
6	12	14	14	11	8	7	7	t'_i
15	14	13	12	11	10	9	8	a_i
12	3	10	1	8	15	6	13	a'_i

23	22	21	20	19	18	17	16	i
15	7	9	11	13	8	6	7	t_i
11	9	8	12	7	15	13	9	t'_i
3	15	6	10	1	13	4	7	a_i
10	5	13	0	7	3	11	6	a'_i

ادامه جدول ۱

31	30	29	28	27	26	25	24	i
12	13	7	11	9	15	12	7	t_i
11	13	15	6	7	12	7	7	t'_i
8	11	14	2	5	9	0	12	a_i
2	1	9	4	12	8	15	14	a'_i

47	46	45	44	43	42	41	40	i
5	7	12	5	6	13	8	14	t_i
5	7	13	13	14	5	13	12	t'_i
12	5	11	13	6	0	7	2	a_i
13	4	0	10	2	12	8	11	a'_i

55	54	53	52	51	50	49	48	i
8	9	15	14	15	14	12	11	t_i
14	6	14	14	11	8	5	15	t'_i
4	12	8	0	10	11	9	1	a_i
0	15	11	3	1	4	6	8	a'_i

63	62	61	60	59	58	57	56	i
12	5	6	8	6	5	14	9	t_i
8	15	5	12	9	12	9	6	t'_i
2	6	5	14	15	7	3	13	a_i
14	10	7	9	13	2	12	5	a'_i

71	70	69	68	67	66	65	64	i
12	13	8	6	11	5	15	9	t_i
6	14	5	12	9	12	5	8	t'_i
10	2	12	7	9	5	0	4	a_i
7	8	5	1	4	10	15	12	a'_i

79	78	77	76	75	74	73	72	i
6	5	8	11	14	13	12	5	t_i
11	11	13	15	5	6	13	8	t'_i
13	15	6	11	8	3	1	14	a_i
11	9	3	0	14	13	2	6	a'_i

۷-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۱۶۰ بیتی ذیل باشد، که در اینجا به عنوان یک دنباله از پنج کلمه Y0, Y1, Y2, Y3, Y4 در یک نمایش مبنای شانزدهی نشان داده می‌شود، که در آن Y0 چپ‌ترین ۳۲ بیت از ۱۶۰ بیت است.

$Y_0 = 67452301,$
$Y_1 = EFCDAB89,$
$Y_2 = 98BADCFE,$

Y_3 = 10325476,
Y_4 = C3D2E1F0.

۲-۷ روش لایه‌گذاری

رشته داده‌ی D نیاز به لایه‌گذاری دارد تا شامل تعدادی بیت گردد. (مضربی صحیحی از ۵۱۲ است) رویه لایه‌گذاری به صورت زیر عمل می‌کند:

۱- تک بیت ′1 به D الحاق می‌شود.

۲- نتیجه‌ی مرحله‌ی قبل به تعدادی بین صفر و۵۱۱، ′0 بیت الحاق می‌شود به گونه‌ای که طول رشته‌ی نتیجه (به بیت) متناسب با ۴۴۸ به پیمانه ۵۱۲ (نسبت ۷ به ۸) است. به طور واضح‌تر، اگر طول اصلی D، L_D و r باقیمانده‌ی تقسیم L_D بر ۵۱۲ باشد، آنگاه تعداد صفرهای الحاقی برابرr-447 (اگر $r \leq 447$) و یا r-959 (اگر$r > 447$) خواهد بود. نتیجه یک رشته بیت به طول ۶۴ بیت به غیر از مضربی صحیح از ۵۱۲ خواهد بود.

۳- نمایش دودویی۶۴ بیتی L_D را به دو رشته‌ی ۳۲ بیتی تقسیم کنید، یکی نشان‌دهنده‌ی "با ارزش‌ترین نیمه" از L_D و بقیه "کم ارزش‌ترین نیمه". حال رشته‌ی نتیجه از مرحله‌ی قبل را به این دو رشته‌ی ۳۲ بیتی الحاق کنید، طوری که "کم ارزش‌ترین نیمه" قبل از "با ارزش‌ترین نیمه" قرار گیرد.

در توصیف تابع گردساز آمده در زیر، هر بلوک داده‌ی ۵۱۲ بیتی D_i، $1 \leq i \leq q$، به عنوان دنباله‌ای از ۱۶ کلمه، $Z_0, Z_1, \ldots, Z_{15}$، در نظرگرفته می‌شود که در آن Z_0 متناظر با چپ‌ترین ۳۲ بیت D_i است.

یادآوری- الحاق دو رشته‌ی ۳۲ بیتی L_D در مرحله‌ی ۳ به گونه‌ای است که این دو رشته‌ی ۳۲ بیتی به طور مستقیم به عنوان کلمات Z_{14} و Z_{15} از بلوک داده‌ی آخر، استفاده می‌شوند؛ براساس قاعده‌ی مرتب‌سازی بایت در بند ۷-۱-۲، کم‌ارزش‌ترین هشت‌تایی L_D چپ‌ترین هشت‌تایی و با ارزش‌ترین هشتایی L_D راست‌ترین هشتایی است.

۳-۷ توصیف تابع گردساز

تابع گردساز Φ به طریق زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W, X_0, X_1, X_2, X_3, X_4, X'_0, X'_1, X'_2, X'_3, X'_4$ برای نمایش یازده کلمه‌ی متمایز استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۵۱۲ بیت (اولین) ورودی به Φ، در $Z_0, Z_1, \ldots, Z_{15}$ قرار گیرند که در آن Z_0 شامل چپ‌ترین ۳۲ بیت این ۵۱۲ بیت است و همچنین ۱۶۰ بیت (دومین) ورودی به Φ، در ۵ کلمه Y_0، Y_1، Y_2، Y_3 و Y_4 قرار گیرد:

۲- قرار دهید: $X_0 := Y_0$ ، $X_1 := Y_1$ ، $X_2 := Y_2$، $X_3 := Y_3$ و $X_4 := Y_4$

۳- قرار دهید: $X'_0 := Y'_0$، $X'_1 := Y'_1$ ، $X'_2 := Y'_2$ ، $X'_3 := Y'_3$ و $X'_4 := Y'_4$.

۴- برای i از ۰ تا ۷۹ ، چهار مرحله‌ی زیر را به ترتیب مشخص شده دنبال کنید:

الف- $W := S^{t_i}(X_0 \uplus g_i(X_1, X_2, X_3) \uplus Z_{ai} \uplus C_i) \uplus X_4$؛

ب- $X_0 := X_4;\ X_4 := X_3;\ X_3 := S^{10}(X_2);\ X_2 := X_1;\ X_1 := W$؛

پ- $W := S^{t'_i}(X'0 \uplus g_{79-i}(X'_1, X'_2, X'_3) \uplus Z_{a'i} \uplus C'_i) \uplus X'_4$؛

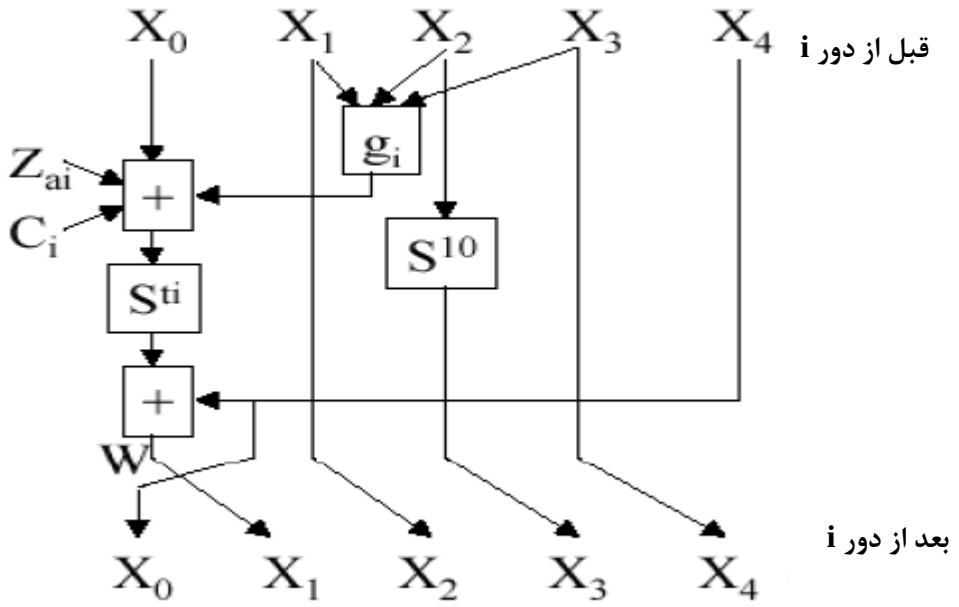
ت- $X'_0 := X'_4;\ X'_4 := X'_3;\ X'_3 := S^{10}(X'_2);\ X'_2 := X'_1;\ X'_1 := W$.

۵- قرار دهید:

$W := Y_0,$
$Y_0 := Y_1 \uplus X_2 \uplus X'_3,$
$Y_1 := Y_2 \uplus X_3 \uplus X'_4,$
$Y_2 := Y_3 \uplus X_4 \uplus X'_0,$
$Y_3 := Y_4 \uplus X_0 \uplus X'_1,$
$Y_4 := W \uplus X_1 \uplus X'_2.$

۶- پنج کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4 ، خروجی تابع گردساز Φ را نمایش می‌دهند. بعد از تکرار آخرین تابع گردساز، ۵کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4 باید با استفاده از معکوس رویه مشخص شده در ۷-۱-۲، به دنباله‌ای از ۲۰ بایت تبدیل شود، که در آن Y_0 باید جای خود را به چهار بایت اول، Y_1 به چهار بایت بعدی و به همین ترتیب تا آخر بدهند. بنابراین اولین (چپ‌ترین) بایت متناظر با کم‌ارزش‌ترین بایت Y_0 و ۲۰ امین (راست‌ترین) بایت متناظر با ارزش‌ترین بایت Y_4 خواهد بود. باید ۲۰ بایت، با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۱۶۰ بیت تبدیل شود؛ به عبارت دیگر اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۱۶۰ امین (راست‌ترین) بیت متناظر با کم ارزش‌ترین بیت ۲۰امین (راست‌ترین) بایت خواهد بود.

شکل ۱ در زیر مراحل الف و ب از مورد ۴ از تابع گردساز Φ در تابع درهم‌ساز اختصاصی ۱ (RIPEMD-160) را نشان می‌دهد. (نیمه‌ی دیگر با همان مراحل پ و ت مشابه هستند.) در تابع گردساز Φ مراحل الف تا ت از مورد ۴، ۸۰ بار (i=0, … , 79) استفاده می‌شوند.



شکل ۱- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۱

۸ تابع درهم‌ساز اختصاصی ۲ (RIPEMD-128)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه، و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸ : سال ۱۳۸۶ توصیف شده است، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، در صورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۲ را بیان می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل بیشینه $2^{64}-1$ بیت، اعمال کرد.

شناسه‌ی تابع درهم‌ساز استاندارد ISO/IEC برای تابع درهم‌ساز اختصاصی ۲، برابر ۳۲ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۲ که در این بند بیان می‌شود، به طور معمول RIPEMD-128 نامیده می‌شود، [3]. این تابع درهم‌ساز بهتر است تنها در برنامه‌های کاربردی استفاده شود که در آن کد درهم شامل ۱۲۸ بیت یا کمتر، به قدر کافی امن در نظر گرفته شود.

۸-۱ پارامترها، توابع و ثابت‌ها

۸-۱-۱ پارامترها

برای این تابع درهم‌ساز $L_1=$ ۵۱۲، $L_2=$ ۱۲۸ و L_H بیشینه ۱۲۸ است.

۸-۱-۲ قرارداد مرتب‌سازی بایت

قرارداد مرتب‌سازی بایت برای این تابع درهم‌ساز نیز مانند تابع بند ۷ است.

۸-۱-۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی کلمات ۳۲ بیتی توصیف می‌شود. دنباله‌ای از توابع $g_0, g_1, \ldots, g_{63}$ ، در این تابع گردساز استفاده می‌شود که در آن هر تابع g_i، $0 \le i \le 63$ ، سه کلمه X_0 , X_1 , X_2 را به عنوان ورودی می‌گیرد و تنها یک کلمه را به عنوان خروجی برمی‌گرداند.

توابع g_i نیز مانند ۶۴ تابع اول بیان شده در زیربند ۷-۱-۳ تعریف می‌شوند.

۸-۱-۴ ثابت‌ها

دو دنباله از کلمات ثابت $C_0, C_1, \ldots , C_{63}$ و $C'_0, C'_1, \ldots , C'_{63}$ در این تابع گردساز استفاده می‌شود. در یک نمایش در مبنای شانزده (که در آن با ارزش‌ترین بیت متناظر با سمت چپ‌ترین بیت است) این ثابت‌ها به صورت زیر تعریف می‌شوند:

C_i = 00000000, $(0 \le i \le 15)$,
C_i = 5A827999, $(16 \le i \le 31)$,
C_i = 6ED9EBA1, $(32 \le i \le 47)$,
C_i = 8F1BBCDC, $(48 \le i \le 63)$,

C'_i = 50A28BE6, $(0 \le i \le 15)$,
C'_i = 5C4DD124, $(16 \le i \le 31)$,

C'_i = 6D703EF3, $(32 \le i \le 47)$,
C'_i = 7A6D76E9, $(48 \le i \le 63)$,

دو دنباله از ۶۴ مقدار جابه‌جایی در این تابع گردساز استفاده می‌شود، که در آن هر مقدار جابه‌جایی بین ۵ و ۱۵ قرار دارد. این دنباله‌ها را با ($t_0, t_1, \dots, t_{63}$) و ($t'_0, t'_1, \dots, t'_{63}$) نشان می‌دهیم و برابر با ۶۴ مقدار اول در دنباله‌های متناظر زیربند ۷-۱-۴ تعریف می‌شوند.

در انتها، دو دنباله‌ی دیگر ۶۴ تایی در این تابع گردساز استفاده می‌شود، که در آن هر مقدار در دنباله بین صفر و ۱۵ قرار دارد. این دنباله‌ها را با ($a_0, a_1, \dots, a_{63}$) و ($a'_0, a'_1, \dots, a'_{63}$) نشان می‌دهیم و برابر با ۶۴ مقدار اول در دنباله‌های متناظر زیربند ۷-۱-۴ تعریف می‌شوند.

۸-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۱۲۸بیتی ذیل باشد، که در اینجا به عنوان یک دنباله از پنج کلمه Y_0, Y_1, Y_2, Y_3 در یک نمایش مبنای شانزدهی نشان داده می‌شود، که در آن Y_0 چپ‌ترین ۳۲ بیت از ۱۲۸بیت است.

Y_0 = 67452301,
Y_1 = EFCDAB89,
Y_2 = 98BADCFE,
Y_3 = 10325476.

۸-۲ روش لایه‌گذاری

روش لایه‌گذاری مورد استفاده‌ی این تابع درهم‌ساز مشابه روش لایه‌گذاری تعریف شده در زیر بند ۷-۲ است.

۸-۳ توصیف تابع گردساز

تابع گردساز Φ مانند زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W, X_0, X_1, X_2, X_3, X'_0, X'_1, X'_2, X'_3,$ برای نمایش یازده کلمه‌ی متمایز استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۵۱۲ بیت (اولین) ورودی به Φ، در $Z_0, Z_1, \dots, Z_{15}$ قرار گیرند که در آن Z_0 شامل سمت چپ‌ترین ۳۲ بیت این ۵۱۲ بیت است. همچنین فرض کنید که ۱۶۰ بیت (دومین) ورودی به Φ، در ۴ کلمه Y_3, Y_2, Y_1, Y_0 قرار می‌گیرد.

۲- قرار دهید: $X_0 := Y_0$ ، $X_1 := Y_1$ ، $X_2 := Y_2$ و $X_3 := Y_3$.

۳- قرار دهید: $X'_0 := Y'_0$ ، $X'_1 := Y'_1$ ، $X'_2 := Y'_2$ و $X'_3 := Y'_3$.

۴- برای $i := 0$ تا ۶۳، چهار مرحله‌ی زیر را به ترتیب مشخص شده دنبال کنید:

الف- $W := S^{ti}(X_0 \boxplus g_i(X_1, X_2, X_3) \boxplus Z_{ai} \boxplus C_i) \boxplus X_4$ ؛

ب- $X_0 := X_3; X_3 := X_2; X_2 := X_1; X_1 := W$ ؛

پ- $W := S^{t'i}(X'0 \boxplus g_{63-i}(X'_1, X'_2, X'_3) \boxplus Z_{a'i} \boxplus C'_i)$ ،

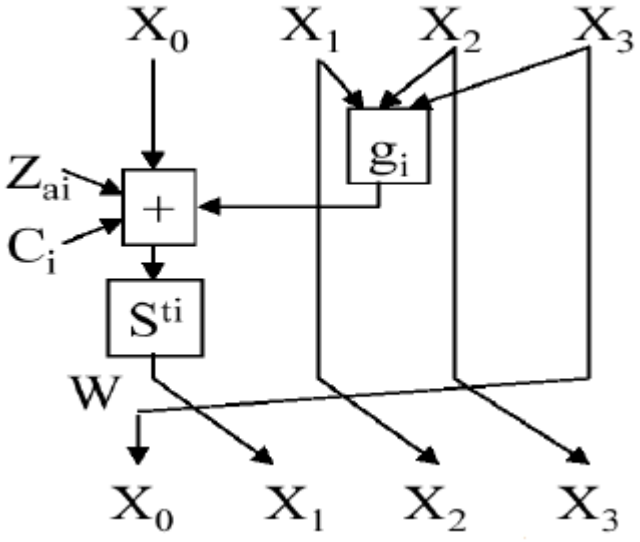
ت- $X'_0 := X'_3; X'_3 := X'_2; X'_2 := X'_1; X'_1 := W$.

۵- قرار دهید

$W := Y_0$,
$Y_0 := Y_1 \boxplus X_2 \boxplus X'_3$,
$Y_1 := Y_2 \boxplus X_3 \boxplus X'_0$,
$Y_2 := Y_3 \boxplus X_0 \boxplus X'_1$,
$Y_3 := W \boxplus X_1 \boxplus X'_2$.

۶- چهار کلمه‌ی Y_0, Y_1, Y_2, Y_3 ، خروجی تابع گردساز Φ را نمایش می‌دهند. بعد از تکرار آخرین تابع گردساز، باید چهارکلمه‌ی Y_0, Y_1, Y_2, Y_3 با استفاده از معکوس رویه مشخص شده در ۷-۱-۲، به دنباله‌ای از ۲۰ بایت تبدیل شود که در آن Y_0 باید جای خود را به چهار بایت اول، Y_1 به چهار بایت بعدی و به همین ترتیب تا آخر بدهند. بنابراین اولین (چپ‌ترین) بایت متناظر با کم‌ارزش‌ترین بایت Y_0 و ۱۶امین (راست‌ترین) بایت متناظر با ارزش‌ترین بایت Y_3 خواهد بود. ۱۶ بایت باید با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۱۲۸ بیت تبدیل شود؛ به عبارت دیگر اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۱۲۸امین (راست‌ترین) بیت متناظر با کم‌ارزش‌ترین بیت ۱۶امین (راست‌ترین) بایت خواهد بود.

شکل ۲ در زیر مراحل الف و ب از مورد ۴ تابع گردساز Φ در تابع درهم‌ساز اختصاصی۲ (RIPEMD-128) را نشان می‌دهد. (نیمه‌ی دیگر یا همان مراحل پ و ت مشابه هستند.) در تابع گردساز Φ مراحل الف تا ت از مورد ۴، ۶۴ بار (i=0, … , 63) استفاده می‌شوند.



شکل ۲- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۲

۹ تابع درهم‌ساز اختصاصی ۳ (SHA-1)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره۱-۹۵۹۸ :۱۳۸۶ توصیف شده است، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، در صورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۳ را بیان می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر $۲^{۶۴}-۱$ بیت، اعمال کرد.
شناسه‌ی تابع درهم‌ساز استاندارد ISO/IECبرای تابع درهم‌ساز اختصاصی ۳ برابر ۳۳ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۳ که در این بند بیان می‌شود به طور معمول SHA-1 نامیده می‌شود. [2].

۹-۱ پارامترها، توابع و ثابت‌ها

۹-۱-۱ پارامترها

برای این تابع درهم‌سازL_1= ۵۱۲ =، L_2= ۱۶۰ و L_H بیشینه ۱۶۰ است.

۹-۱-۲ قاعده‌ی مرتب‌سازی بایت

در مشخصات تابع گردساز این بند فرض شده است که بلوک ورودی به تابع گردساز به صورت یک دنباله از کلمات ۳۲ بیتی است، هر بلوک ۵۱۲ بیتی از ۱۶ تا از این کلمات ساخته می‌شود. یک دنباله از ۶۴ بایت، $B_0, B_1, \dots, B_{63}$، باید به عنوان یک دنباله‌ی ۱۶ کلمه‌ی، $Z_0, Z_1, \dots, Z_{15}$، به صورت آمده در زیر تعبیر شود. هر گروه چهار بایتی متوالی به‌عنوان یک کلمه درنظر گرفته می‌شود، اولین بایت یک کلمه کم‌ارزش‌ترین بایت آن است. بنابراین :

$$Z_i = 2^{24}B_{4i} + 2^{16}B_{4i+1} + 2^{8}B_{4i+2} + B_{4i+3}, \qquad (0 \le i \le 15).$$

برای تبدیل یک کد درهم از یک دنباله از کلمات به یک دنباله بایت، باید معکوس این فرایند دنبال شود.

یادآوری- مرتب‌سازی بایت مشخص شده در اینجا متفاوت از زیربند ۷-۱-۲ است.

۹-۱-۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی کلمات ۳۲ بیتی توصیف می‌شود. دنباله‌ای از توابع $f_0, f_1, \dots, f_{79}$ ، در این تابع گردساز استفاده می‌شود که در آن هر تابع f_i، $0 \le i \le 79$ ، سه کلمه X_0, X_1 و X_2 را به عنوان ورودی گرفته و تنها یک کلمه را به عنوان خروجی برمی‌گرداند.
توابع f_i به صورت زیر تعریف می‌شوند:

$f_i(X_0,X_1,X_2) = (X_0 \wedge X_1) \vee (X_0 \wedge X_2)$, $(0 \le i \le 19)$,
$f_i(X_0,X_1,X_2) = X_0 \oplus X_1 \oplus X_2$, $(20 \le i \le 39)$,
$f_i(X_0,X_1,X_2) = (X_0 \wedge X_1) \vee (X_0 \wedge X_2) \vee (X_1 \wedge X_2)$, $(40 \le i \le 59)$,
$f_i(X_0,X_1,X_2) = X_0 \oplus X_1 \oplus X_2$, $(60 \le i \le 79)$.

۹-۱-۴ ثابت‌ها

دنباله‌ای از کلمات ثابت $C_0, C_1, \ldots, C_{79}$ در این تابع گردساز استفاده می‌شود. در یک نمایش در مبنای شانزده (که در آن با ارزش‌ترین بیت متناظر با چپ‌ترین بیت است) این ثابت‌ها به‌صورت زیر تعریف می‌شوند:

$C_i = 5A827999, \quad (0 \le i \le 19),$
$C_i = 6ED9EBA1, \quad (20 \le i \le 39),$
$C_i = 8F1BBCDC, \quad (40 \le i \le 59),$
$C_i = CA62C1D6, \quad (60 \le i \le 79).$

۹-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۱۶۰بیتی ذیل باشد که در اینجا به عنوان یک دنباله از پنج کلمه Y_0, Y_1, Y_2, Y_3, Y_4 در یک نمایش مبنای شانزدهی نشان داده می‌شود که در آن Y_0 چپ‌ترین ۳۲ بیت از ۱۶۰ بیت است.

$Y_0 = 67452301,$
$Y_1 = EFCDAB89,$
$Y_2 = 98BADCFE,$
$Y_3 = 10325476,$
$Y_4 = C3D2E1F0.$

۹-۲ روش لایه‌گذاری

رشته داده‌ی D نیاز به لایه‌گذاری دارد تا شامل تعدادی، مضربی صحیح از ۵۱۲ بیت گردد. رویه لایه‌گذاری به‌صورت زیر عمل می‌کند:

۱- تک بیت ′1′ به D الحاق می‌شود.

۲- نتیجه‌ی مرحله‌ی قبل به تعدادی بین صفر و ۵۱۱، ′0′ بیت الحاق می‌شود به‌گونه‌ای که طول رشته‌ی نتیجه (به بیت) متناسب با ۴۴۸ به پیمانه ۵۱۲ است. به طور واضح‌تر، اگر طول اصلی D، L_D و r باقیمانده‌ی تقسیم L_D بر ۵۱۲ باشد، آنگاه تعداد صفرهای الحاقی برابر با 447-r (اگر $r \le 447$) یا 959-r (اگر $r > 447$) خواهد بود. نتیجه یک رشته بیت به طول ۶۴ بیت به غیر از مضربی صحیح از ۵۱۲ خواهد بود.

۳- رشته‌ی نتیجه از مرحله‌ی قبل را به نمایش دودویی ۶۴ بیتی L_D الحاق کنید، به‌صورتی که ابتدا با ارزش‌ترین بیت قرار گیرد.

در توصیف تابع گردساز آمده در زیر، هر بلوک داده‌ی ۵۱۲ بیتی D_i، $1 \le i \le q$، به عنوان دنباله‌ای از ۱۶ کلمه، $Z_0, Z_1, \ldots, Z_{15}$، درنظرگرفته می‌شود که در آن Z_0 متناظر با چپ‌ترین ۳۲ بیت D_i است.

یادآوری- الحاق دو رشته‌ی ۳۲ بیتی L_D در مرحله‌ی ۳ به گونه‌ای است که این دو رشته‌ی ۳۲ بیتی به طور مستقیم به عنوان کلمات Z_{14} و Z_{15} از بلوک داده‌ی آخر، استفاده می‌شوند؛ بر اساس قاعده‌ی مرتب‌سازی بایت در بند ۹-۱-۲، کم ارزش‌ترین بایت L_D چپ‌ترین بایت و با ارزش‌ترین بایت L_D راست‌ترین بایت است.

۹-۳ توصیف تابع گردساز

تابع گردساز Φ مانند زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W, X_0, X_1, X_2, X_3, X_4, Z_0, Z_1, \ldots, Z_{79}$ برای نمایش ۸۶ کلمه‌ی متمایز استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۵۱۲ بیت (اولین) ورودی به Φ، در $Z_0, Z_1, \ldots, Z_{15}$ قرار گیرند که در آن Z_0 شامل چپ‌ترین ۳۲ بیت این ۵۱۲ بیت است. همچنین فرض کنید که ۱۶۰ بیت (دومین) ورودی به Φ، در Y_0, Y_1, Y_2, Y_3, Y_4.

۲- برای $i=$ ۱۶ تا ۷۹ قرار دهید:

$$Z_i := S^1(Z_{i-3} \oplus Z_{i-8} \oplus Z_{i-14} \oplus Z_{i-16})$$

۳- قرار دهید : $X_0:=Y_0$ ، $X_1:=Y_1$ ، $X_2:=Y_2$ ، $X_3:=Y_3$ و $X_4:=Y_4$

۴- برای صفر $=i$ تا ۷۹ دو مرحله‌ی زیر را انجام دهید:

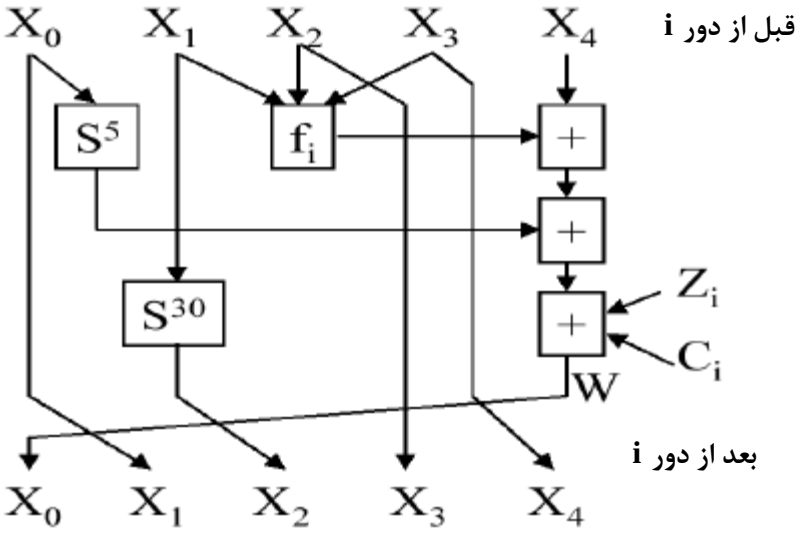
الف- $W := S5(X_0) \uplus fi(X_1, X_2, X_3) \uplus X4 \uplus Zi \uplus Ci$ ؛

ب- $X_4 := X_3; X_3 := X_2; X_2 := S^{30}(X_1); X_1 := X_0; X_0 := W$ ؛

۵- قرار دهید : $Y_0 := Y_0 \uplus X_0$، $Y_1 := Y_1 \uplus X_1$ ، $Y_2 := Y_2 \uplus X_2$ ، $Y_3 := Y_3 \uplus X_3$ و $Y_4 := Y_4 \uplus X_4$.

۶- پنج کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4 ، خروجی تابع گردساز Φ را نمایش می‌دهند. بعد از تکرار آخرین تابع گردساز، ۵کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4 باید با استفاده از معکوس رویه مشخص شده در ۹-۱-۲ ، به دنباله‌ای از ۲۰ بایت تبدیل شود، که در آن Y_0 باید جای خود را به چهار بایت اول، Y_1 به چهار بایت بعدی و به همین ترتیب تا آخر بدهند. بنابراین اولین (چپ‌ترین) بایت متناظر با کم‌ارزش‌ترین بایت Y_0 و ۲۰امین (راست‌ترین) بایت متناظر باارزش‌ترین بایت Y_4 خواهد بود. ۲۰ بایت باید با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۱۶۰ بیت تبدیل شود؛ به عبارت دیگر اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۱۶۰امین (راست‌ترین) بیت متناظر با کم‌ارزش‌ترین بیت ۲۰امین (راست‌ترین) بایت خواهد بود.

شکل ۳ در زیر مراحل الف و ب از مورد ۴ تابع گردساز Φ در تابع درهم‌ساز اختصاصی ۳ (SHA-1) را نشان می‌دهد. در تابع گردساز Φ مراحل الف و ب از مورد ۴، ۸۰ بار ($i=0\ldots79$) استفاده می‌شوند.



شکل ۳- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۳

۱۰ تابع درهم‌ساز اختصاصی ۴ (SHA-256)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸: ۱۳۸۶ توصیف شده، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، درصورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۴ را بیان می‌کند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر $۲^{۶۴}-۱$ بیت، اعمال کرد.

شناسه‌ی تابع درهم‌ساز استاندارد ISO/IEC برای تابع درهم‌ساز اختصاصی ۴ برابر ۳۴ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۴ که در این بند بیان می‌شود به طور معمول SHA-256 نامیده می‌شود [2].

۱-۱۰ پارامترها، توابع و ثابت‌ها

۱-۱-۱۰ پارامترها

برای این تابع درهم‌سازL_1= ۵۱۲، L_2= ۲۵۶ و L_H بیشینه ۲۵۶ است.

۲-۱-۱۰ قاعده‌ی مرتب‌سازی بایت

قاعـده‌ی مرتب‌سـازی بایت برای این تابع درهم‌ساز همانند قاعده‌ی مرتب‌سازی بایت بیان شده در زیربند ۹-۱-۲ است.

۱۰-۱-۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی کلمات ۳۲ بیتی توصیف می‌شود. دنباله‌ای از توابع e_5 ، e_4 ، e_3 ، e_2 ، e_1 ، e_0 ، در این تابع گردساز استفاده می‌شود که در آن e_0 و e_1 هرکدام سه کلمه X_0 , X_1, X_2 و e_5 ، e_4 ، e_3 ، e_2 هرکدام کلمه‌ی X_0 را به عنوان ورودی می‌گیرند و هرکدام از این شش تابع تنها یک کلمه را به عنوان خروجی برمی‌گردانند.

توابع e_5 ، e_4 ، e_3 ، e_2 ، e_1 ، e_0 مانند زیر تعریف می‌شوند:

$e_0(X_0, X_1, X_2) = (X_0 \Lambda X_1) \oplus (X_0 \Lambda X_2)$,
$e_1(X_0, X_1, X_2) = (X_0 \Lambda X_1) \oplus (X_0 \Lambda X_2) \oplus (X_1 \Lambda X_2)$,
$e_2(X_0) = S'^{2}(X_0) \oplus S'^{13}(X_0) \oplus S'^{22}(X_0)$,
$e_3(X_0) = S'^{6}(X_0) \oplus S'^{11}(X_0) \oplus S'^{25}(X_0)$,
$e_4(X_0) = S'^{7}(X_0) \oplus S'^{18}(X_0) \oplus R^{3}(X_0)$,
$e_5(X_0) = S'^{17}(X_0) \oplus S'^{19}(X_0) \oplus R^{10}(X_0)$.

۱۰-۱-۴ ثابت‌ها

دنباله‌ای از کلمات ثابت $C_0, C_1, \ldots, C_{63}$ در این تابع گردساز استفاده می‌شود. در یک نمایش در مبنای شانزده (که در آن باارزش‌ترین بیت متناظر با چپ‌ترین بیت است) این ثابت‌ها به‌صورت زیر تعریف می‌شوند، که در آن کلمات به ترتیب $C_0, C_1, \ldots, C_{63}$ قرار گرفته‌اند:

```
428a2f98 71374491 b5c0fbcf e9b5dba5 3956c25b 59f111f1 923f82a4 ab1c5ed5
d807aa98 12835b01 243185be 550c7dc3 72be5d74 80deb1fe 9bdc06a7 c19bf174
e49b69c1 efbe4786 0fc19dc6 240ca1cc 2de92c6f 4a7484aa 5cb0a9dc 76f988da
983e5152 a831c66d b00327c8 bf597fc7 c6e00bf3 d5a79147 06ca6351 14292967
27b70a85 2e1b2138 4d2c6dfc 53380d13 650a7354 766a0abb 81c2c92e 92722c85
a2bfe8a1 a81a664b c24b8b70 c76c51a3 d192e819 d6990624 f40e3585 106aa070
19a4c116 1e376c08 2748774c 34b0bcb5 391c0cb3 4ed8aa4a 5b9cca4f 682e6ff3
748f82ee 78a5636f 84c87814 8cc70208 90befffa a4506ceb bef9a3f7 c67178f2
```

یادآوری- این مقادیر اولین سی و دو بیتی بخش‌های کسری ریشه‌ی سوم از اولین عدد اول شصت و چهار هستند.

۱۰-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۲۵۶ بیتی ذیل باشد، که در اینجا به عنوان یک دنباله از هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ در یک نمایش مبنای شانزدهی نشان داده می‌شود که در آن Y_0 چپ‌ترین ۳۲ بیت از ۲۵۶ بیت است.

Y_0 = 6a09e667,
Y_1 = bb67ae85,

Y_2 = 3c6ef372,
Y_3 = a54ff53a,
Y_4 = 510e527f,
Y_5 = 9b05688c,
Y_6 = 1f83d9ab,
Y_7 = 5be0cd19.

یادآوری- این مقادیر با گرفتن بخش‌های کسری ریشه‌ی دوم اولین هشت عدد اول به‌دست می‌آیند.

۱۰-۲ روش لایه‌گذاری

روش لایه‌گذاری مورد استفاده‌ی این تابع درهم‌ساز مشابه روش لایه‌گذاری تعریف شده در زیر بند ۹-۲ است.

۱۰-۳ توصیف تابع گردساز

تابع گردساز Φ مانند زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W_1, W_2, X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, Z_0, Z_1, \dots, Z_{63}$ برای نمایش ۷۴ کلمه‌ی متمایز استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۵۱۲ بیت (اولین) ورودی به Φ، در $Z_0, Z_1, \dots, Z_{15}$ قرار گیرند که در آن Z_0 شامل چپ‌ترین ۳۲ بیت این ۵۱۲ بیت است. همچنین فرض کنید که ۲۵۶ بیت (دومین) ورودی به Φ، در هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ قرار گیرد.

۲- برای $i=16$ تا ۶۳ قرار دهید:

$Z_i := e_5(Z_{i-2}) \boxplus Z_{i-7} \boxplus e_4(Z_{i-15}) \boxplus Z_{i-16}$

۳- قرار دهید: $X_0:=Y_0$ ، $X_1:=Y_1$ ، $X_2:=Y_2$ ، $X_3:=Y_3$ ، $X_4:=Y_4$ ، $X_5:=Y_5$ ، $X_6:=Y_6$ ، $X_7:=Y_7$.

۴- برای صفر $i=$ تا ۶۳ سه مرحله‌ی زیر را انجام دهید:

الف- $W_1 := X_7 \boxplus e_3(X_4) \boxplus e_0(X_4, X_5, X_6) \boxplus C_i \boxplus Z_i$؛

ب- $W_2 := e_2(X_0) \boxplus e_1(X_0, X_1, X_2)$؛

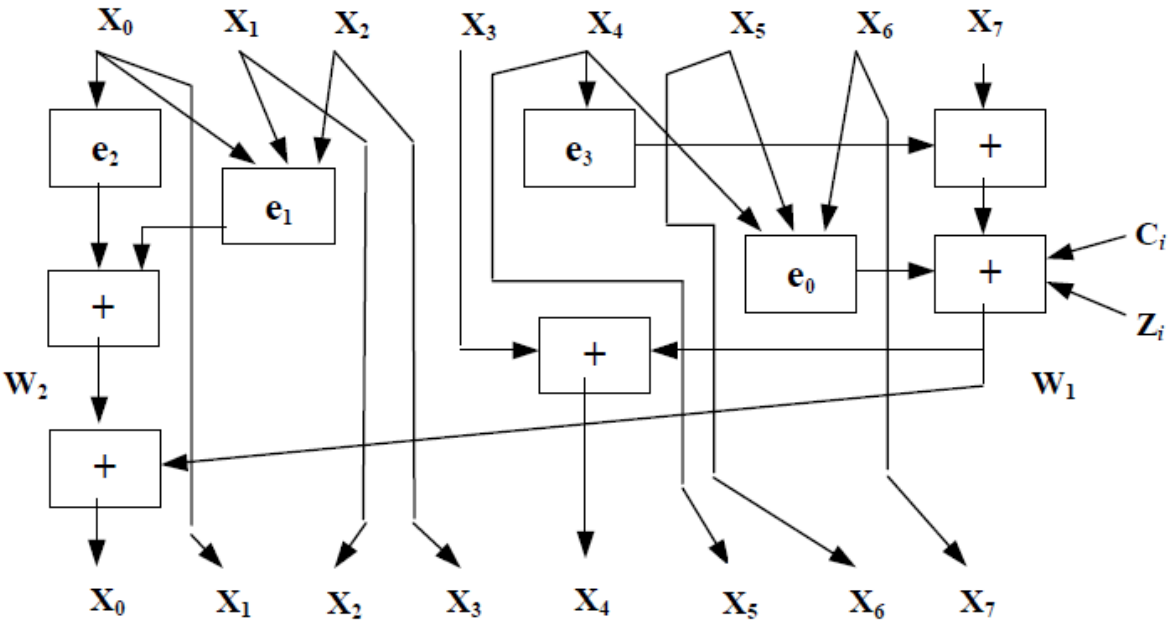
پ- $X_7 := X_6; X_6 := X_5; X_5 := X_4; X_4 := X_3 \boxplus W_1; X_3 := X_2; X_2 := X_1; X_1 := X_0; X_0 := W_1 \boxplus W_2$

۵- قرار دهید:

$Y_0 := Y_0 \boxplus X_0$ ، $Y_1 := Y_1 \boxplus X_1$ ، $Y_2 := Y_2 \boxplus X_2$ ، $Y_3 := Y_3 \boxplus X_3$ ، $Y_4 := Y_4 \boxplus X_4$ ، $Y_5 := Y_5 \boxplus X_5$ ، $Y_6 := Y_6 \boxplus X_6$ و $Y_7 := Y_7 \boxplus X_7$.

۶- هشت کلمه‌ی $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ خروجی تابع گردساز Φ را نمایش می‌دهند. بعد از تکرار آخرین تابع گردساز، هشت کلمه‌ی $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ باید با استفاده از معکوس رویه مشخص شده در ۱۰-۱-۲ ، به دنباله‌ای از ۳۲ بایت تبدیل شود که در آن Y_0 باید جای خود را به چهار بایت اول، Y_1 به چهار بایت بعدی و به همین ترتیب تا آخر بدهند. بنابراین اولین (چپ‌ترین) بایت متناظر با کم‌ارزش‌ترین بایت Y_0 و ۳۲امین (راست‌ترین) بایت متناظر باارزش‌ترین بایت Y_7 خواهد بود. ۳۲ بایت باید با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۲۵۶ بیت تبدیل شود؛ به عبارت دیگر

اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۲۵۶ امین (راست‌ترین) بیت متناظر با کم‌ارزش‌ترین بیت ۳۲امین (راست‌ترین) بایت خواهد بود.
شکل ۴ در زیر مراحل الف، ب و پ از مورد ۴ تابع گردساز Φ در تابع درهم‌ساز اختصاصی۴ (SHA-256) را نشان می‌دهد. در تابع گردساز Φ مراحل الف، ب و پ از مورد ۴، ۶۴ بار ($i=0, \ldots, 63$) استفاده می‌شوند.



شکل ۴- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۴

۱۱ تابع درهم‌ساز اختصاصی ۵ (SHA-512)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸: ۱۳۸۶ توصیف شده، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه، و تابع گردساز مشخص شده در اینجا، درصورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۵ را بیان می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر $2^{128}-1$ بیت، اعمال کرد.
شناسه‌ی تابع درهم‌ساز استاندارد ISO/IEC برای تابع درهم‌ساز اختصاصی ۵ برابر ۳۵ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۵ که در این بند بیان می‌شود به طور معمول SHA-512 نامیده می‌شود. [2].

۱۱-۱ پارامترها، توابع و ثابت‌ها

۱۱-۱-۱ پارامترها

برای این تابع درهم‌ساز$L_1=$ ۱۰۲۴، $L_2=$ ۵۱۲ و L_H بیشینه۵۱۲ است.

۱۱-۱-۲ قاعده‌ی مرتب‌سازی بایت

در مشخصات تابع گردساز این بند فرض شده است که بلوک ورودی به تابع گردساز به صورت یک دنباله از کلمات ۶۴ بیتی است، هر بلوک ۱۰۲۴بیتی از ۱۶ تا از این کلمات ساخته می‌شود. یک دنباله از ۱۲۸ بایت، $B_0, B_1, \ldots, B_{127}$ ، باید به عنوان یک دنباله‌ی ۱۶ کلمه‌ای، $Z_0, Z_1, \ldots, Z_{15}$، به‌صورت آمده در زیر تعبیر شود. هر گروه هشت بایتی متوالی به عنوان یک کلمه در نظر گرفته می‌شود، اولین بایت یک کلمه کم‌ارزش‌ترین بایت آن است. بنابراین :

$$Z_i = 2^{56}B_{8i} + 2^{48}B_{8i+1} + 2^{40}B_{8i+2} + 2^{32}B_{8i+3} + 2^{24}B_{8i+4} + 2^{16}B_{8i+5} + 2^{8}B_{8i+6} + B_{8i+7} \quad (0 \le i \le 15).$$

برای تبدیل یک کد درهم از یک دنباله از کلمات به یک دنباله بایت، باید معکوس این فرایند دنبال شود.

۱۱-۱-۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی کلمات ۶۴ بیتی توصیف می‌شود. دنباله‌ای از توابع d_0 ، d_1 ، d_2 ، d_3 ، d_4 ، d_5 ، در این تابع گردساز استفاده می‌شود، که در آن d_0 و d_1 هرکدام سه کلمه‌ی ۶۴ بیتی X_0, X_1, X_2 را به عنوان ورودی و d_3 ، d_4 ، d_5 و d_2 هرکدام کلمه‌ی ۶۴ بیتی X_0 را به عنوان ورودی می‌گیرند و هر کدام از این شش تابع تنها یک کلمه‌ی ۶۴ بیتی را به عنوان خروجی برمی‌گردانند.

توابع d_5 ، d_4 ، d_3 ، d_2 ، d_1 و d_0 به‌صورت زیر تعریف می‌شوند:

$$d_0(X_0, X_1, X_2) = (X_0 \Lambda X_1) \oplus (X_0 \Lambda X_2),$$
$$d_1(X_0, X_1, X_2) = (X_0 \Lambda X_1) \oplus (X_0 \Lambda X_2) \oplus (X_1 \Lambda X_2),$$
$$d_2(X_0) = S'^{28}(X_0) \oplus S'^{34}(X_0) \oplus S'^{39}(X_0),$$
$$d_3(X_0) = S'^{14}(X_0) \oplus S'^{18}(X_0) \oplus S'^{41}(X_0),$$
$$d_4(X_0) = S'^{1}(X_0) \oplus S'^{8}(X_0) \oplus R^{7}(X_0),$$
$$d_5(X_0) = S'^{19}(X_0) \oplus S'^{61}(X_0) \oplus R^{6}(X_0).$$

۱۱-۱-۴ ثابت‌ها

دنباله‌ای از کلمات ثابت $C_0, C_1, \ldots, C_{79}$ در این تابع گردساز استفاده می‌شود. در یک نمایش در مبنای شانزده (که در آن با ارزش‌ترین بیت متناظر با چپ‌ترین بیت است.) این ثابت‌ها به‌صورت زیر تعریف می‌شوند، که در آن کلمات به ترتیب $C_0, C_1, \ldots, C_{79}$ قرار گرفته‌اند:

```
428a2f98d728ae22 7137449123ef65cd b5c0fbcfec4d3b2f e9b5dba58189dbbc
3956c25bf348b538 59f111f1b605d019 923f82a4af194f9b ab1c5ed5da6d8118
d807aa98a3030242 12835b0145706fbe 243185be4ee4b28c 550c7dc3d5ffb4e2
72be5d74f27b896f 80deb1fe3b1696b1 9bdc06a725c71235 c19bf174cf692694
e49b69c19ef14ad2 efbe4786384f25e3 0fc19dc68b8cd5b5 240ca1cc77ac9c65
2de92c6f592b0275 4a7484aa6ea6e483 5cb0a9dcbd41fbd4 76f988da831153b5
983e5152ee66dfab a831c66d2db43210 b00327c898fb213f bf597fc7beef0ee4
```

```
c6e00bf33da88fc2 d5a79147930aa725 06ca6351e003826f 142929670a0e6e70
27b70a8546d22ffc 2e1b21385c26c926 4d2c6dfc5ac42aed 53380d139d95b3df
650a73548baf63de 766a0abb3c77b2a8 81c2c92e47edaee6 92722c851482353b
a2bfe8a14cf10364 a81a664bbc423001 c24b8b70d0f89791 c76c51a30654be30
d192e819d6ef5218 d69906245565a910 f40e35855771202a 106aa07032bbd1b8
19a4c116b8d2d0c8 1e376c085141ab53 2748774cdf8eeb99 34b0bcb5e19b48a8
391c0cb3c5c95a63 4ed8aa4ae3418acb 5b9cca4f7763e373 682e6ff3d6b2b8a3
748f82ee5defb2fc 78a5636f43172f60 84c87814a1f0ab72 8cc702081a6439ec
90befffa23631e28 a4506cebde82bde9 bef9a3f7b2c67915 c67178f2e372532b
ca273eceea26619c d186b8c721c0c207 eada7dd6cde0eb1e f57d4f7fee6ed178
06f067aa72176fba 0a637dc5a2c898a6 113f9804bef90dae 1b710b35131c471b
28db77f523047d84 32caab7b40c72493 3c9ebe0a15c9bebc 431d67c49c100d4c
4cc5d4becb3e42b6 597f299cfc657e2a 5fcb6fab3ad6faec 6c44198c4a475817
```

یادآوری- این مقادیر اولین شصت و چهار بیتی بخش‌های کسری ریشه‌ی سوم عدد اول اولین هشتاد هستند.

۱۱-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۵۱۲ بیتی ذیل باشد، که در اینجا به عنوان یک دنباله از هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ در یک نمایش مبنای شانزده‌ای نشان داده می‌شود، که در آن Y_0 چپ‌ترین ۶۴ بیت از ۵۱۲ بیت است.

Y_0 = 6a09e667f3bcc908,
Y_1 = bb67ae8584caa73b,
Y_2 = 3c6ef372fe94f82b,
Y_3 = a54ff53a5f1d36f1,
Y_4 = 510e527fade682d1,
Y_5 = 9b05688c2b3e6c1f,
Y_6 = 1f83d9abfb41bd6b,
Y_7 = 5be0cd19137e2179.

یادآوری- این مقادیر با گرفتن بخش‌های کسری ریشه‌ی دوم عدد اول اولین هشت به‌دست می‌آیند.

۱۱-۲ روش لایه‌گذاری

رشته داده‌ی D نیاز به لایه‌گذاری دارد تا شامل تعدادی بیت باشد که مضربی از ۱۰۲۴بیت گردد. رویه لایه‌گذاری به صورت زیر عمل می‌کند:

۱- تک بیت $'1'$ به D الحاق می‌شود.

۲- نتیجه‌ی مرحله‌ی قبل به تعدادی بین صفر و ۱۰۲۳، $'0'$ بیت الحاق می‌شود به گونه‌ای که طول رشته‌ی نتیجه (به بیت) متناسب با ۸۹۶ به پیمانه ۱۰۲۴ است. به طور واضح‌تر، اگر طول اصلی D، L_D و r

باقیمانده‌ی تقسیم L_D بر ۱۰۲۴ باشـد، آنگاه تعداد صفرهای الحاقی برابر 897-r (اگر $r \leq 895$) و یا 1919-r (اگر $r > 895$) خواهد بود. نتیجه یک رشته بیت به طول 128 بیت به غیر از مضربی صحیح از 1024 خواهد بود.

۳- رشته‌ی نتیجه از مرحله‌ی قبل را به نمایش دودویی 128بیتی L_D الحاق کنید، به‌صورتی که ابتدا با ارزش‌ترین بیت قرار گیرد.

در توصیف تابع گردساز آمده در زیر، هر بلوک داده‌ی 1024بیتی D_i، $1 \leq i \leq q$، به عنوان دنباله‌ای از ۱۶ کلمه، $Z_0, Z_1, \ldots, Z_{15}$، درنظرگرفته می‌شود که در آن Z_0 متناظر با چپ‌ترین ۶۴ بیت D_i است.

یادآوری- الحاق دو رشته‌ی ۱۲۸بیتی L_D در مرحله‌ی ۳ به گونه‌ای است که این دو رشته‌ی ۶۴ بیتی به طور مستقیم به عنوان کلمات Z_{14} و Z_{15} از بلوک داده‌ی آخر، استفاده می‌شوند؛ براساس قاعده‌ی مرتب‌سازی بایت در بند ۱۱-۱-۲، کم‌ارزش‌ترین بایت L_D چپ‌ترین بایت، و با ارزش‌ترین بایت L_D راست‌ترین بایت است.

۱۱-۳ توصیف تابع گردساز

تابع گردساز Φ مانند زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W_1, W_2, X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, Z_0, Z_1, \ldots, Z_{79}$ برای نمایش ۹۰ کلمه‌ی متمایز استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۱۰۲۴بیت (اولین) ورودی به Φ، در $Z_0, Z_1, \ldots, Z_{15}$ قرار گیرند که در آن Z_0 شامل چپ‌ترین ۶۴ بیت این ۱۰۲۴بیت است. همچنین فرض کنید که ۵۱۲ بیت (دومین) ورودی به Φ، در هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ قرار گیرد.

۲- برای $i=$ ۱۶ تا ۷۹ قرار دهید:

$$Z_i := d_5(Z_{i-2}) \uplus Z_{i-7} \uplus d_4(Z_{i-15}) \uplus Z_{i-16}$$

۳- قرار دهید: $X_0:=Y_0$ ، $X_1:=Y_1$ ، $X_2:=Y_2$، $X_3:=Y_3$، $X_4:=Y_4$، $X_5:=Y_5$، $X_6:=Y_6$، $X_7:=Y_7$.

۴- برای صفر $=i$ تا ۶۳ سه مرحله‌ی زیر را انجام دهید:

الف- $W_1 := X_7 \uplus d_3(X_4) \uplus d_0(X_4, X_5, X_6) \uplus C_i \uplus Z_i$؛

ب- $W_2 := d_2(X_0) \uplus d_1(X_0, X_1, X_2)$؛

پ- $X_7 := X_6; X_6 := X_5; X_5 := X_4; X_4 := X_3 \uplus W_1; X_3 := X_2; X_2 := X_1; X_1 := X_0; X_0 := W_1 \uplus W_2$

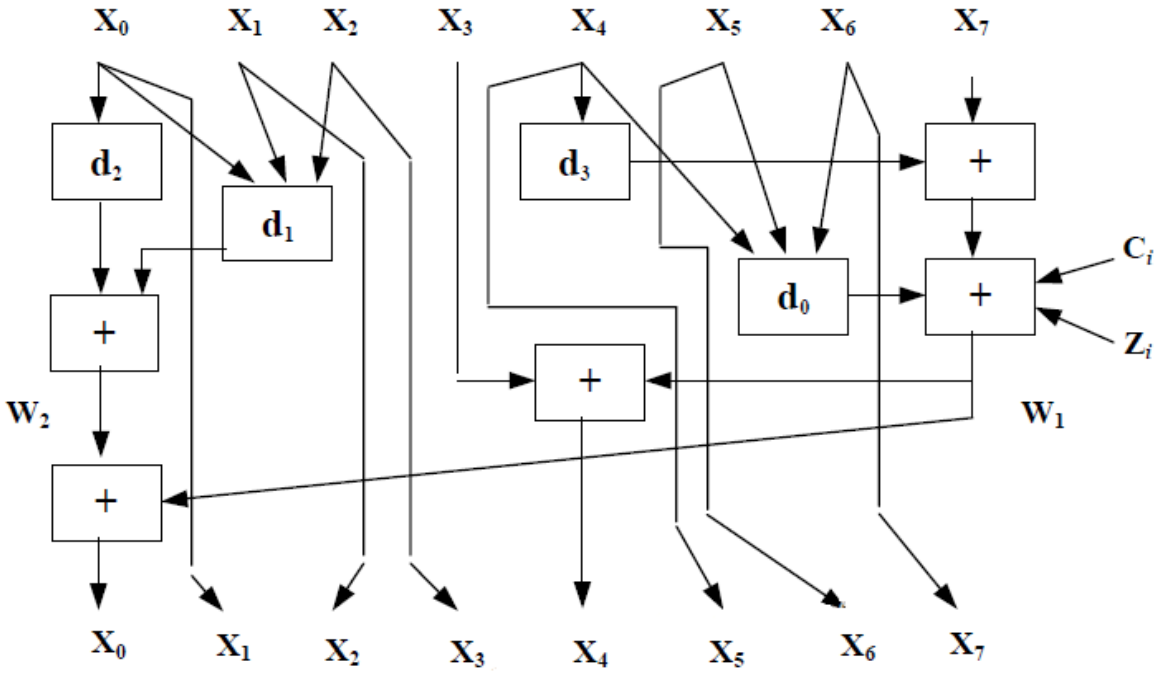
۵- قرار دهید:

$Y_0 := Y_0 \uplus X_0$ ، $Y_1 := Y_1 \uplus X_1$ ، $Y_2 := Y_2 \uplus X_2$ ، $Y_3 := Y_3 \uplus X_3$ ، $Y_4 := Y_4 \uplus X_4$ ، $Y_5 := Y_5 \uplus X_5$ ، $Y_6 := Y_6 \uplus X_6$ و $Y_7 := Y_7 \uplus X_7$.

۶- هشت کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4، خروجی تابع گردساز Φ را نمایش می‌دهند. بعد از تکرار آخرین تابع گردساز، هشت کلمه‌ی Y_0, Y_1, Y_2, Y_3, Y_4 باید با استفاده از معکوس رویه مشخص شده در ۱۱-۱-۲ ، به دنباله‌ای از ۶۴ بایت تبدیل شود که در آن Y_0 باید جای خود را به هشت بایت اول، Y_1 به هشت بایت بعدی و به همین ترتیب تا آخر بدهند. بنابراین اولین (چپ‌ترین) بایت متناظر با کم‌ارزش‌ترین بایت Y_0 و

۶۴امین (راست‌ترین) بایت متناظر باارزش‌ترین بایت Y_7 خواهد بود. ۶۴ بایت باید با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۵۱۲ بیت تبدیل شود؛ به عبارت دیگر اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۵۱۲امین (راست‌ترین) بیت متناظر با کم‌ارزش‌ترین بیت ۶۴امین (راست‌ترین) بایت خواهد بود.

شکل ۵ در زیر مراحل الف، ب و پ از مورد ۴ تابع گردساز Φ در تابع درهم‌ساز اختصاصی ۵ (SHA-512) را نشان می‌دهد. در تابع گردساز Φ مراحل الف، ب و پ از مورد ۴، ۸۰ بار ($i=0, \ldots, 79$) استفاده می‌شوند.



شکل ۵- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۵

۱۲ تابع درهم‌ساز اختصاصی ۶ (SHA-384)

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸ : سال ۱۳۸۶ توصیف شده است، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، در صورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۶ را بیان می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر $2^{128}-1$ بیت، اعمال کرد.

شناساگری تابع درهم‌ساز استاندارد ISO/IECبرای تابع درهم‌ساز اختصاصی ۶ برابر ۳۶ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۶ که در این بند بیان می‌شود به طور معمول SHA-384 نامیده می‌شود. [2].

۱-۱۲ پارامترها، توابع و ثابت‌ها

۱-۱-۱۲ پارامترها

برای این تابع درهم‌سازL_1= ۱۰۲۴، L_2= ۵۱۲ و L_H = ۳۸۴ است.

۲-۱-۱۲ قاعده‌ی مرتب‌سازی بایت

قاعده‌ی مرتب‌سازی بایت برای این تابع درهم‌ساز همانند قاعده‌ی مرتب‌سازی تابع درهم ساز بند ۱۱ است.

۳-۱-۱۲ توابع

توابع برای این تابع درهم‌ساز همانند توابع درهم‌ساز بند ۱۱ است.

۴-۱-۱۲ ثابت‌ها

ثابت‌ها برای این تابع درهم‌ساز همانند ثابت‌های تابع درهم‌ساز بند ۱۱ است.

۵-۱-۱۲ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV، باید همواره رشته‌ی ۵۱۲ بیتی ذیل باشد، که در اینجا به عنوان یک دنباله از هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ در یک نمایش مبنای شانزدهی نشان داده می‌شود، که در آن Y_0 چپ‌ترین ۶۴ بیت از ۵۱۲ بیت است.

Y_0 = cbbb9d5dc1059ed8,
Y_1 = 629a292a367cd507,
Y_2 = 9159015a3070dd17,
Y_3 = 152fecd8f70e5939,
Y_4 = 67332667ffc00b31,
Y_5 = 8eb44a8768581511,
Y_6 = db0c2e0d64f98fa7,
Y_7 = 47b5481dbefa4fa4-

یادآوری- این مقادیر با گرفتن بخش‌های کسری ریشه‌ی دوم عدد اول نهمین شانزده به‌دست می‌آیند.

۲-۱۲ روش لایه‌گذاری

روش لایه‌گذاری برای این تابع درهم‌ساز همانند روش لایه‌گذاری بیان شده در بند ۱۱ است.

۳-۱۲ توصیف تابع گردساز

تابع گردسازی که باید برای این تابع درهم‌ساز استفاده شود همانند تابع گردساز بیان شده در بند ۱۱-۳ است.

درهم‌ساز ۳۸۴ بیتی نهایی با کوتاه کردن خروجی درهم‌ساز مبتنی بر SHA-512 به چپ‌ترین ۳۸۴ بیت آن، به‌دست می‌آید.

۱۳ تابع درهم‌ساز اختصاصی ۷ (گرداب[1])

در این بند به بیان یک روش لایه‌گذاری، یک مقدار اولیه و یک تابع گردساز برای استفاده در مدل کلی برای توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸ : سال ۱۳۸۶توصیف شده است، می‌پردازیم. روش لایه‌گذاری، مقدار اولیه و تابع گردساز مشخص شده در اینجا، در صورت استفاده در مدل کلی بالا، با یکدیگر تابع درهم‌ساز اختصاصی ۷ را بیان می‌کنند. این تابع درهم‌ساز اختصاصی را می‌توان به تمامی رشته‌های داده‌ای D، شامل حدکثر $2^{256}-1$ بیت، اعمال کرد.
شناسه‌ی تابع درهم‌ساز استاندارد ISO/IECبرای تابع درهم‌ساز اختصاصی ۷ برابر ۳۷ (در مبنای شانزده) است.

یادآوری- تابع درهم‌ساز اختصاصی ۷ که در این بند بیان می‌شود به طور معمول WHIRPOOL نامیده می‌شود[4].

۱-۱۳ پارامترها، توابع و ثابت‌ها

۱-۱-۱۳ پارامترها

برای این تابع درهم‌ساز L_1= ۵۱۲، L_2= ۵۱۲ و L_H بیشینه ۵۱۲ است.

۲-۱-۱۳ قاعده‌ی مرتب‌سازی بایت

در مشخصات تابع گردساز این بند فرض شده است که بلوک ورودی به تابع گردساز به‌صورت یک ماتریس M (تمام ماتریس‌ها در اینجا ۸ در ۸ با ورودی‌های انتخابی از میدان $GF(2^8)$ هستند.) است، هر بلوک ۵۱۲ بیتی از چنین ماتریسی ساخته می‌شود. دنباله‌ای از ۶۴ بایت، $B = (B_0, B_1, \dots, B_{63})$ ، باید به عنوان یک ماتریس M به روش آمده در ادامه تعبیر شود. ورودی سطر و ستون اول ماتریس چپ‌ترین بایت (که چپ‌ترین بایت متناظر با ارزش‌ترین بایت است.) دنباله‌ی B (یا همان B0)، ورودی سطر اول و ستون دوم ماتریس دومین بایت چپ B (یا همان B1)، ... ، و ورودی سطر و ستون هشتادم راست‌ترین بایت B (یا همان B_{63}) باشد. این عمل با استفاده از تابع c_0 مشخص شده در زیربند ۱۳-۱-۳ انجام می‌شود.
برای تبدیل یک کد درهم از یک ماتریس به یک دنباله بایت، باید فرایند معکوس تابع c_0 دنبال شود.

۳-۱-۱۳ توابع

برای تسهیل پیاده‌سازی نرم‌افزاری، تابع گردساز Φ از لحاظ عملیات روی ماتریس M توصیف می‌شود. دنباله‌ای از توابع c_4 ، c_3 ، c_2 ، c_1 ، c_0 در این تابع گردساز استفاده می‌شود. این توابع به صورت زیر تعریف می‌شوند. تابع c_0 یک دنباله‌ی ۶۴ بایتی ، $B = (B_0, B_1, \dots, B_{63})$ ، را به عنوان ورودی می‌گیرد و ماتریس $Z'=(z'_{ij})$ را به عنوان خروجی تولید می‌کند که در آن

$z'_{ij} = B_{8i+j,}$ $\quad (0 \geq i, j \geq 7)$

این به آن معناست که $Z' = c_0(B)$ اگر و تنها اگر $0 \geq i, j \geq 7$ $z'_{ij} = B_{8i+j}$

[1]-WHIRPOOL

تابع c_1 ماتریس $X''=(x''_{ij})$ را به عنوان ورودی می‌گیرد و ماتریس دیگر $W'=(w'_{ij})$ را به عنوان خروجی تولید می‌کند که در آن

$w'ij = s[x''ij], \qquad (0 \geq i, j \geq 7)$

و s یک جعبه‌ی جایگزینی غیرخطی است. این به آن معناست که $W'=c_1(X'')$ اگر و تنها اگر

$w'ij = s[x''ij] \qquad 0 \geq i, j \geq 7)$

جعبه‌ی s عنصر $x \in GF(2^8)$ را با عنصر دیگر $s[x] \in GF(2^8)$ جایگزین می‌کند؛ همان گونه که در جدول ۲ مشخص شده (عناصر واقع در ستون اول " باارزش‌ترین نیمه" از x و عناصر واقع در سطر اول " کم ارزش‌ترین نیمه" از x هستند؛ به عنوان مثال، اگر x=01010110=56 (در مبنای شانزده) باشد آنگاه s[x]=01001001=49 (در مبنای شانزده).

جدول ۲- جعبه‌ی s

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	18	23	C6	E8	87	B8	01	4F	36	A6	D2	F5	79	6F	91	52
1	60	BC	9B	8E	A3	0C	7B	35	1D	E0	D7	C2	2E	4B	FE	57
2	15	77	37	LS	9F	F0	4A	DA	58	C9	29	0A	B1	A0	6B	85
3	BD	5D	10	F4	CB	3E	05	67	E4	27	41	8B	A7	7D	95	D8
4	FB	EE	7C	66	DD	17	47	9E	CA	2D	BF	07	AD	5A	83	33
5	63	02	AA	71	C8	19	49	D9	F2	E3	5B	88	9A	26	32	B0
6	E9	0F	D5	80	BE	CD	34	48	FF	7A	90	5F	20	68	1A	AE
7	B4	54	93	22	64	F1	73	12	40	08	C3	EC	DB	A1	8D	3D
8	97	00	CF	2B	76	82	D6	1B	B5	AF	6A	50	45	F3	30	EF
9	3F	55	A2	EA	65	BA	2F	30	DE	1C	FD	4D	92	75	06	8A
A	B2	E6	0E	1F	62	D4	A8	96	F9	C5	25	59	84	72	39	4C
B	5E	78	38	8C	D1	A5	E2	61	B3	21	9C	1E	43	C7	FC	04
C	51	99	6d	0D	FA	DF	7E	24	3B	AB	CE	11	8F	4E	B7	EB
D	3C	81	94	F7	B9	13	2C	D3	E7	6E	C4	03	56	44	7F	A9
E	2A	BB	C1	53	DC	0B	9D	6C	31	74	F6	46	AC	89	14	E1
F	16	3A	69	09	70	B6	D0	ED	CC	42	98	A4	28	5C	F8	86

تابع c_2 ماتریس $X''= (x''_{ij})$ را به عنوان ورودی می‌گیرد و ماتریس دیگر $W'= (w'_{ij})$ را به عنوان خروجی تولید می‌کند که در آن

$w'_{ij} = x''_{(i-j) \bmod 8,j}, \qquad ((0 \geq i, j \geq 7)$

این به آن معناست که $W'= c_2(X'')$ اگر و تنها اگر $w'_{ij} = x''_{(i-j) \bmod 8,j} \quad (0 \geq i, j \geq 7$

تابع c_3 ماتریس X'' را به عنوان ورودی گرفته و ماتریس دیگر W' را به عنوان خروجی تولید می‌کند که در آن

$W' = X'' \bullet C'',$

و C'' ماتریس ۸ در ۸ گردشی با ورودی‌های انتخابی از میدان $GF(2^8)$ است، همان طور که در زیر آمده است:

$$C'' = \begin{bmatrix} 01 & 01 & 04 & 01 & 08 & 05 & 02 & 09 \\ 09 & 01 & 01 & 04 & 01 & 08 & 05 & 02 \\ 02 & 09 & 01 & 01 & 04 & 01 & 08 & 05 \\ 05 & 02 & 09 & 01 & 01 & 04 & 01 & 08 \\ 08 & 05 & 02 & 09 & 01 & 01 & 04 & 01 \\ 01 & 08 & 05 & 02 & 09 & 01 & 01 & 04 \\ 04 & 01 & 08 & 05 & 02 & 09 & 01 & 01 \\ 01 & 04 & 01 & 08 & 05 & 02 & 09 & 01 \end{bmatrix}$$

این به آن معناست که $W' = c_3(X'')$ اگر و تنها اگر $W' = X'' \bullet C''$.

تابع c_4 دو ماتریس $X'' = (x''_{ij})$ و $Y'' = (y''_{ij})$ را به عنوان ورودی می‌گیرد و تنها ماتریس $W' = (w'_{ij})$ را به عنوان خروجی تولید می‌کند که در آن

$$w'_{ij} = x''_{ij} \oplus y'_{ij}, \qquad (0 \geq i, j \geq 7)$$

این به آن معناست که $W' = c_2(X'')$ اگر و تنها اگر $w'_{ij} = x''_{ij} \oplus y'_{ij}$ $(0 \geq i, j \geq 7)$

۱۳-۱-۴ ثابت‌ها

دنباله‌ای از ماتریس‌های ثابت $A^r = (A^r_{ij})$ $(0 < r \leq 10)$ در این تابع گردساز استفاده می‌شود. ثابت گردساز برای rامین گردش، ماتریسی است که به‌صورت زیر تعریف می‌شود:

$$A^r_{0j} = s[8(r - 1) + j], \qquad (0 \geq j \geq 7)$$
$$A^r_{ij} = 0, \qquad (`1 \geq i \geq 7, 0 \geq j \geq 7)$$

۱۳-۱-۵ مقدار اولیه

مقدار اولیه IV رشته‌ی ۵۱۲ بیتی ′0′ است.

۱۳-۲ روش لایه‌گذاری

رشته داده‌ی D نیاز به لایه‌گذاری دارد تا شامل تعدادی، مضربی صحیح از ۵۱۲، بیت گردد. رویه لایه‌گذاری به‌صورت زیر عمل می‌کند:

۱- تک بیت ′1′ به D الحاق می‌شود.

۲- نتیجه‌ی مرحله‌ی قبل به تعدادی بین صفر و ۱۰۲۳، ′0′ بیت الحاق می‌شود به گونه‌ای که طول رشته‌ی نتیجه (به بیت) مضرب فردی از ۲۵۶ گردد.

۳- اگر طول اصلی D، L_D باشد، رشته‌ی نتیجه از مرحله‌ی قبل را به نمایش دودویی ۲۵۶ بیتی L_D الحاق کنید، به‌صورتی که ابتدا باارزش‌ترین بیت قرار گیرد.

در توصیف تابع گردساز زیر، هر بلوک داده‌ی ۵۱۲بیتی D_i، $1 \leq i \leq q$، به عنوان ماتریس $Z' = (z'_{ij})$ $(0 \leq i, j \leq 7)$، همان طور که در زیربند ۱۳-۱-۲ مشخص شده است، در نظر گرفته می‌شود که در آن z'_{00} متناظر با چپ‌ترین ۸ بیت D_i و z'_{77} متناظر با راست‌ترین ۸ بیت D_i است.

یادآوری- الحاق دو رشته‌ی ۲۵۶ بیتی L_D در مرحله‌ی ۳ به گونه‌ای است که این دو رشته‌ی ۲۵۶ بیتی به طور مستقیم به عنوان دومین نیمه‌ی آخرین ماتریس داده، استفاده می‌شوند؛ براساس قاعده‌ی مرتب‌سازی بایت در بند ۱۳-۱-۲، باارزش‌ترین بایت L_D در سطر پنجم و ستون اول و باارزش‌ترین بایت L_D در هشتمین سطر و هشتمین ستون ماتریس است.

۱۳-۳ توصیف تابع گردساز

تابع گردساز Φ مانند زیر عمل می‌کند. به یاد داشته باشید که در این توصیف، از نمادهای $W', X'', K_0, K_1, \ldots, K_{10}$ برای نمایش ۱۳ ماتریس متمایز با ورودی‌های انتخابی از میدان $GF(2^8)$ ، استفاده می‌کنیم که شامل مقادیر لازم در محاسبات هستند.

۱- فرض کنید که ۵۱۲ بیت (اولین) ورودی به Φ، در یک ماتریس Z' با ورودی‌های انتخابی از میدان$GF(2^8)$ که با استفاده از قاعده‌ی مرتب‌سازی بایت مشخص شده در زیربند ۱۳-۱-۲ شکل یافته‌اند، قرار گیرند. همچنین فرض کنید که ۵۱۲ بیت (دومین) ورودی به Φ، در یک ماتریس Y' با ورودی‌های انتخابی از میدان $GF(2^8)$ قرار می‌گیرند.

۲- قرار دهید : $K_0 = Y'$، همچنین برای $i = ۱$ تا ۱۰ قرار دهید:

$K_i := c_4(c_3(c_2(c_1(K_{i-1}))), A^i)$.

یادآوری- این مرحله ماتریس Y' را به دنباله‌ای از کلیدهای گردساز $K0, \ldots, K10$ بسط می‌دهد.

۳- قرار دهید : $X'' := c_4(Z', K_0)$ و برای $j = ۱$ تا ۱۰ دو مرحله‌ی زیر را انجام دهید:

الف- $W' := c_4(c_3(c_2(c_1(X'')))), K_j)$ ؛

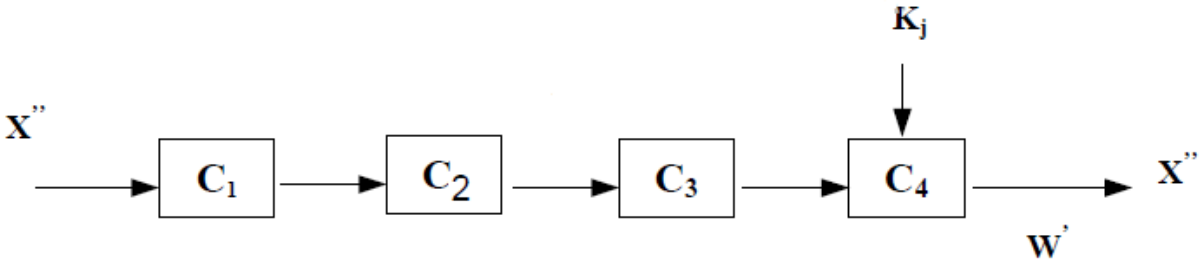
ب- $X'' := W'$.

۴- قرار دهید : $Y' := W' \oplus K_0 \oplus Z'$

۵- ماتریس Y' خروجی تابع گردساز Φ را نمایش می‌دهد. بعد از تکرار آخرین تابع گردساز، ماتریس Y' باید با استفاده از معکوس رویه مشخص شده در ۱۳-۱-۲، به دنباله‌ای از ۶۴ بایت تبدیل شود که در آن ورودی سطر و ستون اول ماتریس باید جای خود را به اولین بایت، ورودی سطر اول و ستون دوم ماتریس به بایت بعدی و ... ورودی سطر و ستون هشتم ماتریس جای خود را به بایت آخر بدهند. ۶۴ بایت باید با استفاده از معکوس رویه مشخص شده در بند ۶، به رشته‌ای از ۵۱۲ بیت تبدیل شود؛ به عبارت دیگر اولین (چپ‌ترین) بیت متناظر باارزش‌ترین بیت اولین (چپ‌ترین) بایت و ۵۱۲امین (راست‌ترین) بیت متناظر با کم‌ارزش‌ترین بیت ۶۴امین (راست‌ترین) بایت خواهد بود.

شکل ۶ در زیر مراحل الف و ب از مورد ۳ تابع گردساز Φ در تابع درهم‌ساز اختصاصی۷ (WHIRLPOOL) را نشان می‌دهد. در تابع گردساز Φ مراحل نشان داده شده در شکل ۶، ۱۰ بار ($i=0, \ldots, 10$) استفاده

می‌شوند.



شکل ۶- بخش تابع گردساز در تابع درهم‌ساز اختصاصی ۶

۱۴ تابع درهم‌ساز اختصاصی ۸ (SHA-224)

در این بند یک روش لایه‌گذاری، یک مقدار اولیه، و یک تابع گردساز را برای استفاده در مدل کلی توابع درهم‌ساز که در استاندارد ملی شماره ۱-۹۵۹۸ : سال ۱۳۸۶ شرح داده شده، تعین می‌کنیم. این روش لایه‌گذاری، مقدار اولیه، و تابع گردساز که در اینجا تعیین می‌شود وقتی در مدل کلی فوق استفاده شود، با هم تابع اختصاصی درهم‌ساز ۸ را مشخص می‌کنند. این تابع درهم‌ساز اختصاصی می‌تواند برای هر رشته داده مانند D که بیشینه $۲^{۶۴}-۱$ بیت طول دارد، به کار رود.

شناسه تابع درهم‌ساز استاندارد ISO/IEC برای تابع درهم‌ساز اختصاصی ۸ برابر با ۳۸ است (در مبنای شانزده).

یادآوری- تابع اختصاصی درهم‌ساز ۸ که در این بند معرفی شده عموماً SHA-224،[2] نامیده می‌شود.

۱۳-۴ پارامترها، توابع و ثابت ها

۱۴-۱-۱ پارامترها

برای این تابع درهم‌ساز L_1=۵۱۲ ، L_2=۲۵۶ و L_H=۲۲۴ است.

۱۴-۱-۲ قرارداد ترتیب بایت

قرارداد ترتیب بایت برای این تابع درهم‌ساز، مانند تابع درهم‌ساز بند ۱۰ است.

۱۴-۱-۳ توابع

توابع برای این تابع درهم‌ساز، به همان صورت تابع درهم‌ساز بند ۱۰ است.

۱۴-۱-۴ ثابت ها

ثابت‌ها برای این تابع درهم‌ساز، به همان صورت تابع درهم‌ساز بند ۱۰ است.

۱۴-۱-۵ مقدار اولیه

برای این تابع گردساز مقدار اولیه، IV ، باید همواره به‌صورت رشته‌ی ۲۵۶ بیتی زیر باشد که در اینجا به‌صورت دنباله‌ای از هشت حرف $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ در یک نمایش در مبنای شانزده طوری نشان داده شده که Y_0 مشخص کننده ۳۲ بیت سمت چپ از ۲۵۶ بیت است:

Y_0 = c1059ed8

Y_1 = 367cd507

Y_2 = 3070dd17

Y_3 = f70e5939

Y_4 = ffc00b31

Y_5 = 68581511

Y_6 = 64f98fa7

Y_7 = befa4fa4

یادآوری- این مقادیر ۳۲ بیت مرتبه پایین مقادیر مشخص شده در زیربند ۱۲-۱-۵ هستند.

۱۴-۲ روش لایه‌گذاری

یک روش لایه‌گذاری که با این تابع درهم‌ساز استفاده می شود، باید مشابه روش لایه‌گذاری معرفی شده در بند ۱۰-۲ باشد.

۱۴-۳ توصیف تابع گردساز

تابع سازی که با این تابع درهم‌ساز استفاده می‌شود، باید مشابه تابع گردساز معرفی شده در ۱۰-۳ باشد.
۲۲۴ بیت درهم‌ساز نهایی به‌وسیله بریدن ۲۲۴ بیت سمت چپ خروجی درهم‌ساز مبنی بر SHA-256 به‌دست می‌آید.

پیوست الف

(اطلاعاتی)

مثال‌ها

این پیوست شامل مثال‌هایی برای محاسبه توابع درهم‌ساز اختصاصی ۱ تا ۸ است. برای هر تابع درهم‌ساز، مقادیر میانه که در طول عملیات تابع درهم‌ساز مشتق شده، در بعضی مثال‌ها آورده شده است.

در سراسر این پیوست به رمزگذاری ASCII رشته‌های داده ارجاع می‌دهیم؛ که معادل رمزگذاری مورد استفاده در ISO 646 است.

الف-۱ تابع درهم‌ساز اختصاصی ۱

یادآوری- مرجع [۳] شامل توصیف شبه کد تابع درهم‌ساز اختصاصی ۱.

الف-۱-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
9C 11 85 A5 C5 E9 FC 54 61 28 08 97 7E E8 F5 48 B2 25 8D 31
```

الف-۱-۲ مثال ۲

در این مثال رشته-داده شامل تنها یک بایت است؛ معادل کد ASCII حرف ’a‘ .

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
0B DC 9D 2D 25 6B 3E E9 DA AE 34 7B E6 F4 DC 83 5A 46 7F FE
```

الف-۱-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی شامل معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
80636261 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000018 00000000
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X'_0, X'_1, X'_2, X'_3, X'_4$ آمده است.

```
67452301, EFCDAB89, 98BADCFE, 10325476, C3D2E1F0, 67452301, EFCDAB89, 98BADCFE, 10325476, C3D2E1F0
C3D2E1F0, 3115FC67, EFCDAB89, EB73FA62, 10325476, C3D2E1F0, DDD63FB8, EFCDAB89, EB73FA62, 10325476
10325476, B41192D5, 3115FC67, 36AE27BF, EB73FA62, 10325476, 322E7AE3, DDD63FB8, 36AE27BF, EB73FA62
```

EB73FA62, 3A35DC50, B41192D5, 57F19CC4, 36AE27BF, EB73FA62, 883EE903, 322E7AE3, 58FEE377, 36AE27BF
36AE27BF, D3786413, 3A35DC50, 464B56D0, 57F19CC4, 36AE27BF, 92B2B79B, 883EE903, B9EB8CC8, 58FEE377
57F19CC4, 0E946720, D3786413, D77140E8, 464B56D0, 58FEE377, F9091FF2, 92B2B79B, FBA40E20, B9EB8CC8
464B56D0, D52BF632, 0E946720, E1904F4D, D77140E8, B9EB8CC8, E5B09992, F9091FF2, CADE6E4A, FBA40E20
D77140E8, 150BD8A8, D52BF632, 519C803A, E1904F4D, FBA40E20, 8B2D9FB3, E5B09992, 247FCBE4, CADE6E4A
E1904F4D, 3D6F601F, 150BD8A8, AFD8CB54, 519C803A, CADE6E4A, E755F422, 8B2D9FB3, C2664B96, 247FCBE4
519C803A, B7B60384, 3D6F601F, 2F62A054, AFD8CB54, 247FCBE4, 5922D09E, E755F422, B67ECE2C, C2664B96
AFD8CB54, B85A0A3F, B7B60384, BD807CF5, 2F62A054, C2664B96, CF24E72C, 5922D09E, 57D08B9D, B67ECE2C
2F62A054, 7F8B38E5, B85A0A3F, D80E12DE, BD807CF5, B67ECE2C, CA6A1C75, CF24E72C, 8B427964, 57D08B9D
BD807CF5, 9DACA495, 7F8B38E5, 6828FEE1, D80E12DE, 57D08B9D, 227F6D84, CA6A1C75, 939CB33C, 8B427964
D80E12DE, BC05F46F, 9DACA495, 2CE395FE, 6828FEE1, 8B427964, 5D801685, 227F6D84, A871D729, 939CB33C
6828FEE1, 1494F053, BC05F46F, B2925676, 2CE395FE, 939CB33C, B3C3F4D5, 5D801685, FDB61089, A871D729
2CE395FE, 85861D02, 1494F053, 17D1BEF0, B2925676, A871D729, 3D16242D, B3C3F4D5, 005A1576, FDB61089
B2925676, 597BF629, 85861D02, 53C14C52, 17D1BEF0, FDB61089, FF459078, 3D16242D, 0FD356CF, 005A1576
17D1BEF0, 6347EF78, 597BF629, 18740A16, 53C14C52, 005A1576, 927E40A8, FF459078, 5890B4F4, 0FD356CF
53C14C52, 45C8FA44, 6347EF78, EFD8A565, 18740A16, 0FD356CF, ACBB994E, 927E40A8, 1641E3FD, 5890B4F4
18740A16, AD2956AF, 45C8FA44, 1FBDE18D, EFD8A565, 5890B4F4, AD30AD24, ACBB994E, F902A249, 1641E3FD
EFD8A565, 5EAF16B7, AD2956AF, 23E91117, 1FBDE18D, 1641E3FD, 6261732E, AD30AD24, EE653AB2, F902A249
1FBDE18D, 41730D4B, 5EAF16B7, A55ABEB4, 23E91117, F902A249, 45ED27AF, 6261732E, C2B492B4, EE653AB2
23E91117, FC0CCBD3, 41730D4B, BC5ADD7A, A55ABEB4, EE653AB2, 243C5668, 45ED27AF, 85CCB989, C2B492B4
A55ABEB4, 042ECC93, FC0CCBD3, CC352D05, BC5ADD7A, C2B492B4, 82F89BD1, 243C5668, B49EBD17, 85CCB989
BC5ADD7A, 4D4D4377, 042ECC93, 332F4FF0, CC352D05, 85CCB989, 5FC74686, 82F89BD1, F159A090, B49EBD17
CC352D05, 5207002B, 4D4D4377, BB324C10, 332F4FF0, B49EBD17, B2720031, 5FC74686, E26F460B, F159A090
332F4FF0, 388278F5, 5207002B, 350DDD35, BB324C10, F159A090, 58A100F8, B2720031, 1D1A197F, E26F460B
BB324C10, 62879D70, 388278F5, 1C00AD48, 350DDD35, E26F460B, 5992068B, 58A100F8, C800C6C9, 1D1A197F
350DDD35, A30A1FD9, 62879D70, 09E3D4E2, 1C00AD48, 1D1A197F, CC290DCA, 5992068B, 8403E162, C800C6C9
1C00AD48, BDA2B31B, A30A1FD9, 1E75C18A, 09E3D4E2, C800C6C9, 863D625E, CC290DCA, 481A2D66, 8403E162
09E3D4E2, F7211DEE, BDA2B31B, 287F668C, 1E75C18A, 8403E162, 6061B5A5, 863D625E, A4372B30, 481A2D66
1E75C18A, B6A665C6, F7211DEE, 8ACC6EF6, 287F668C, 481A2D66, AA98ADB5, 6061B5A5, F5897A18, A4372B30
287F668C, 2D30FA02, B6A665C6, 8477BBDC, 8ACC6EF6, A4372B30, 2999255A, AA98ADB5, 86D69581, F5897A18
8ACC6EF6, C76D12F9, 2D30FA02, 99971ADA, 8477BBDC, F5897A18, 98237631, 2999255A, 62B6D6AA, 86D69581
8477BBDC, 516F84DF, C76D12F9, C3E808B4, 99971ADA, 86D69581, 6C472A90, 98237631, 649568A6, 62B6D6AA
99971ADA, F3FA5B05, 516F84DF, B44BE71D, C3E808B4, 62B6D6AA, 2EAD5672, 6C472A90, 8DD8C660, 649568A6
C3E808B4, D539625E, F3FA5B05, BE137D45, B44BE71D, 649568A6, C5CB48BA, 2EAD5672, 1CAA41B1, 8DD8C660
B44BE71D, D8500C99, D539625E, E96C17CF, BE137D45, 8DD8C660, 05286DFB, C5CB48BA, B559C8BA, 1CAA41B1
BE137D45, 7ECDE5B2, D8500C99, E5897B54, E96C17CF, 1CAA41B1, 88396DD2, 05286DFB, 2D22EB17, B559C8BA
E96C17CF, 681D30B9, 7ECDE5B2, 40326761, E5897B54, B559C8BA, 333F2212, 88396DD2, A1B7EC14, 2D22EB17
E5897B54, 960F7BFD, 681D30B9, 3796C9FB, 40326761, 2D22EB17, C699295B, 333F2212, E5B74A20, A1B7EC14
40326761, 6770E498, 960F7BFD, 74C2E5A0, 3796C9FB, A1B7EC14, BFD68874, C699295B, FC8848CC, E5B74A20
3796C9FB, 75EB06C5, 6770E498, 3DEFF658, 74C2E5A0, E5B74A20, BDDF3474, BFD68874, 64A56F1A, FC8848CC
74C2E5A0, 14FA827A, 75EB06C5, C392619D, 3DEFF658, FC8848CC, 8CBC87E9, BDDF3474, 5A21D2FF, 64A56F1A
3DEFF658, 804B0068, 14FA827A, AC1B15D7, C392619D, 64A56F1A, CDDA6EBF, 8CBC87E9, 7CD1D2F7, 5A21D2FF
C392619D, 475BA81B, 804B0068, EA09E853, AC1B15D7, 5A21D2FF, 656C7DA3, CDDA6EBF, F21FA632, 7CD1D2F7
AC1B15D7, D26BC25D, 475BA81B, 2C01A201, EA09E853, 7CD1D2F7, 76D66CA3, 656C7DA3, 69BAFF37, F21FA632
EA09E853, DBC5A2CB, D26BC25D, 6EA06D1D, 2C01A201, F21FA632, C9B17F72, 76D66CA3, B1F68D95, 69BAFF37
2C01A201, 77367F5E, DBC5A2CB, AF097749, 6EA06D1D, 69BAFF37, 65A60151, C9B17F72, 59B28DDB, B1F68D95
6EA06D1D, 8155A6B4, 77367F5E, 168B2F6F, AF097749, B1F68D95, 33F3AC81, 65A60151, C5FDCB26, 59B28DDB
AF097749, C90C4D38, 8155A6B4, D9FD79DC, 168B2F6F, 59B28DDB, 9BFB827D, 33F3AC81, 98054596, C5FDCB26
168B2F6F, 9762713B, C90C4D38, 569AD205, D9FD79DC, C5FDCB26, DDC8130E, 9BFB827D, CEB204CF, 98054596
D9FD79DC, 7EBF9C32, 9762713B, 3134E324, 569AD205, 98054596, C24C2C79, DDC8130E, EE09F66F, CEB204CF
569AD205, 20EFFA01, 7EBF9C32, 89C4EE5D, 3134E324, CEB204CF, F255847E, C24C2C79, 204C3B77, EE09F66F

3134E324, 75B7117F, 20EFFA01, FE70C9FA, 89C4EE5D, EE09F66F, DCD63949, F255847E, 30B1E709, 204C3B77
89C4EE5D, A96BE4C7, 75B7117F, BFE80483, FE70C9FA, 204C3B77, 5B99238D, DCD63949, 5611FBC9, 30B1E709
FE70C9FA, 5E3201FC, A96BE4C7, DC45FDD6, BFE80483, 30B1E709, B43484F4, 5B99238D, 58E52773, 5611FBC9
BFE80483, 2CF95A98, 5E3201FC, AF931EA5, DC45FDD6, 5611FBC9, 52325A09, B43484F4, 648E356E, 58E52773
DC45FDD6, 1393F0C3, 2CF95A98, C807F178, AF931EA5, 58E52773, D015577D, 52325A09, D213D2D0, 648E356E
AF931EA5, BB49CCF7, 1393F0C3, E56A60B3, C807F178, 648E356E, BB9C87C4, D015577D, C9682548, D213D2D0
C807F178, 6A330EB4, BB49CCF7, 4FC30C4E, E56A60B3, D213D2D0, B1BB1A2E, BB9C87C4, 555DF740, C9682548
E56A60B3, 14E58204, 6A330EB4, 2733DEED, 4FC30C4E, C9682548, AC77F96D, B1BB1A2E, 721F12EE, 555DF740
4FC30C4E, 79AAF53E, 14E58204, CC3AD1A8, 2733DEED, 555DF740, 1774D326, AC77F96D, EC68BAC6, 721F12EE
2733DEED, 210769B3, 79AAF53E, 96081053, CC3AD1A8, 721F12EE, A625F112, 1774D326, DFE5B6B1, EC68BAC6
CC3AD1A8, F44B53A7, 210769B3, ABD4F9E6, 96081053, EC68BAC6, 5DCA4D12, A625F112, D34C985D, DFE5B6B1
96081053, 7C1E3640, F44B53A7, 1DA6CC84, ABD4F9E6, DFE5B6B1, EBC4D9C6, 5DCA4D12, 97C44A98, D34C985D
ABD4F9E6, 06B59EE8, 7C1E3640, 2D4E9FD1, 1DA6CC84, D34C985D, 095F37FD, EBC4D9C6, 29344977, 97C44A98
1DA6CC84, C422C3CD, 06B59EE8, 78D901F0, 2D4E9FD1, 97C44A98, 5BBEE487, 095F37FD, 13671BAF, 29344977
2D4E9FD1, AD864025, C422C3CD, D67BA01A, 78D901F0, 29344977, BF5B2529, 5BBEE487, 7CDFF425, 13671BAF
78D901F0, 29A83BB5, AD864025, 8B0F3710, D67BA01A, 13671BAF, FB5747C5, BF5B2529, FB921D6E, 7CDFF425
D67BA01A, 626E3910, 29A83BB5, 190096B6, 8B0F3710, 7CDFF425, DD935A5F, FB5747C5, 6C94A6FD, FB921D6E
8B0F3710, A719D8BC, 626E3910, A0EED4A6, 190096B6, FB921D6E, 27754F3A, DD935A5F, 5D1F17ED, 6C94A6FD
190096B6, BA84C782, A719D8BC, B8E44189, A0EED4A6, 6C94A6FD, 4F5CA4A5, 27754F3A, 4D697F76, 5D1F17ED
A0EED4A6, 9F6887A9, BA84C782, 6762F29C, B8E44189, 5D1F17ED, 325AFE7E, 4F5CA4A5, D53CE89D, 4D697F76
B8E44189, 3A88288C, 9F6887A9, 131E0AEA, 6762F29C, 4D697F76, 86AFE021, 325AFE7E, 7292953D, D53CE89D
6762F29C, AB23F78F, 3A88288C, A21EA67D, 131E0AEA, D53CE89D, C97F9EA1, 86AFE021, 6BF9F8C9, 7292953D
131E0AEA, 7299044A, AB23F78F, 20A230EA, A21EA67D, 7292953D, 9F60751C, C97F9EA1, BF80861A, 6BF9F8C9
A21EA67D, 6A3F10CF, 7299044A, 8FDE3EAC, 20A230EA, 6BF9F8C9, 1E9CE713, 9F60751C, FE7A8725, BF80861A
20A230EA, 1A1B904D, 6A3F10CF, 641129CA, 8FDE3EAC, BF80861A, C13F038A, 1E9CE713, 81D4727D, FE7A8725
8FDE3EAC, 0B2CDC01, 1A1B904D, FC433DA8, 641129CA, FE7A8725, BF627814, C13F038A, 739C4C7A, 81D4727D
641129CA, D563BFDC, 0B2CDC01, 6E413468, FC433DA8, 81D4727D, 5FCCBADE, BF627814, FC0E2B04, 739C4C7A

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

8E B2 08 F7 E0 5D 98 7A 9B 04 4A 8E 98 C6 B0 87 F1 5A 0B FC

الف-۱-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´
کد درهم رشته‌ی ۱۶۰ بیتی ذیل است:

5D 06 89 EF 49 D2 FA E5 72 B8 81 B1 23 A8 5F FA 21 59 5F 36

الف-۱-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII
´abcdefghijklmnopqrstuvwxyz´
کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

F7 1C 27 10 9C 69 2C 1B 56 BB DC EB 5B 9D 28 65 B3 70 8D BC

الف-۱-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
B0 E2 0B 6E 31 16 64 02 86 ED 3A 87 A5 71 30 79 B2 1F 51 89
```

الف-۱-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار ‘1234567890’

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
9B 75 2E 45 57 3D 4B 39 F4 DB D3 32 3C AB 82 BF 63 32 6B FB
```

الف-۱-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

بعد از فرایند لایه‌گذاری، بلوک ۱۶کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
64636261 65646362 66656463 67666564 68676665 69686766 6A696867 6B6A6968
6C6B6A69 6D6C6B6A 6E6D6C6B 6F6E6D6C 706F6E6D 71706F6E 00000080 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 000001C0 00000000
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X'_0, X'_1, X'_2, X'_3, X'_4$ که در طول پردازش اولین بلوک به‌دست آمده، آورده شده است.

```
67452301, EFCDAB89, 98BADCFE, 10325476, C3D2E1F0, 67452301, EFCDAB89, 98BADCFE, 10325476, C3D2E1F0
C3D2E1F0, 3115FB87, EFCDAB89, EB73FA62, 10325476, C3D2E1F0, 463DA521, EFCDAB89, EB73FA62, 10325476
10325476, CC21EC2E, 3115FB87, 36AE27BF, EB73FA62, 10325476, DB247A12, 463DA521, 36AE27BF, EB73FA62
EB73FA62, DFEB9B7A, CC21EC2E, 57EE1CC4, 36AE27BF, EB73FA62, 1D166A23, DB247A12, F6948518, 36AE27BF
36AE27BF, 2363912E, DFEB9B7A, 87B0BB30, 57EE1CC4, 36AE27BF, CE7A12F6, 1D166A23, 91E84B6C, F6948518
57EE1CC4, A1B60DC7, 2363912E, AE6DEB7F, 87B0BB30, F6948518, 57FF19DD, CE7A12F6, 59A88C74, 91E84B6C
87B0BB30, 96AC7C1E, A1B60DC7, 8E44B88D, AE6DEB7F, 91E84B6C, 01A9FEFA, 57FF19DD, E84BDB39, 59A88C74
AE6DEB7F, 6AE46154, 96AC7C1E, D8371E86, 8E44B88D, 59A88C74, 5D9A609C, 01A9FEFA, FC67755F, E84BDB39
8E44B88D, 3CF61F09, 6AE46154, B1F07A5A, D8371E86, E84BDB39, 030F7FE7, 5D9A609C, A7FBE806, FC67755F
D8371E86, 696F0D9A, 3CF61F09, 918551AB, B1F07A5A, FC67755F, 7456C8E3, 030F7FE7, 69827176, A7FBE806
B1F07A5A, AB957B91, 696F0D9A, D87C24F3, 918551AB, A7FBE806, F64C4453, 7456C8E3, 3DFF9C0C, 69827176
918551AB, 9FF4A064, AB957B91, BC3669A5, D87C24F3, 69827176, 22A5FE6E, F64C4453, 5B238DD1, 3DFF9C0C
D87C24F3, 912FE998, 9FF4A064, 55EE46AE, BC3669A5, 3DFF9C0C, 8D7E53E4, 22A5FE6E, 31114FD9, 5B238DD1
BC3669A5, C45F164E, 912FE998, D281927F, 55EE46AE, 5B238DD1, 695B23B7, 8D7E53E4, 97F9B88A, 31114FD9
```

55EE46AE, 2211A508, C45F164E, BFA66244, D281927F, 31114FD9, 6FAA776F, 695B23B7, F94F9235, 97F9B88A
D281927F, 80B1F3DE, 2211A508, 7C593B11, BFA66244, 97F9B88A, 4D94F720, 6FAA776F, 6C8EDDA5, F94F9235
BFA66244, 3AA6A8F5, 80B1F3DE, 46942088, 7C593B11, F94F9235, D81C6137, 4D94F720, A9DDBDBE, 6C8EDDA5
7C593B11, 9E4C4BF6, 3AA6A8F5, C7CF7A02, 46942088, 6C8EDDA5, B2ECCABD, D81C6137, 53DC8136, A9DDBDBE
46942088, F929216E, 9E4C4BF6, 9AA3D4EA, C7CF7A02, A9DDBDBE, A96B1820, B2ECCABD, 7184DF60, 53DC8136
C7CF7A02, D9AEEFAF, F929216E, 312FDA79, 9AA3D4EA, 53DC8136, 5A5E09B3, A96B1820, B32AF6CB, 7184DF60
9AA3D4EA, 8BB34505, D9AEEFAF, A485BBE4, 312FDA79, 7184DF60, 616711FA, 5A5E09B3, AC6082A5, B32AF6CB
312FDA79, 07067302, 8BB34505, BBBEBF66, A485BBE4, B32AF6CB, F4F47116, 616711FA, 7826CD69, AC6082A5
A485BBE4, 51997747, 07067302, CD14162E, BBBEBF66, AC6082A5, FAE97297, F4F47116, 9C47E985, 7826CD69
BBBEBF66, C213132C, 51997747, 19CC081C, CD14162E, 7826CD69, 887E5A3F, FAE97297, D1C45BD3, 9C47E985
CD14162E, 29D001F0, C213132C, 65DD1D46, 19CC081C, 9C47E985, 187068EF, 887E5A3F, A5CA5FEB, D1C45BD3
19CC081C, 2B59B58A, 29D001F0, 4C4CB308, 65DD1D46, D1C45BD3, 56C66FD3, 187068EF, F968FE21, A5CA5FEB
65DD1D46, C45681A6, 2B59B58A, 4007C0A7, 4C4CB308, A5CA5FEB, D718432A, 56C66FD3, C1A3BC61, F968FE21
4C4CB308, 2E32CA16, C45681A6, 66D628AD, 4007C0A7, F968FE21, 775BA27D, D718432A, 19BF4D5B, C1A3BC61
4007C0A7, 5C712D51, 2E32CA16, 5A069B11, 66D628AD, C1A3BC61, 6243D22F, 775BA27D, 610CAB5C, 19BF4D5B
66D628AD, 989BC126, 5C712D51, CB2858B8, 5A069B11, 19BF4D5B, 44DCD35A, 6243D22F, 6E89F5DD, 610CAB5C
5A069B11, 9EE4CA1F, 989BC126, C4B54571, CB2858B8, 610CAB5C, 8FBE3F7E, 44DCD35A, 0F48BD89, 6E89F5DD
CB2858B8, F417F849, 9EE4CA1F, 6F049A62, C4B54571, 6E89F5DD, DA718428, 8FBE3F7E, 734D6913, 0F48BD89
C4B54571, 75239882, F417F849, 93287E7B, 6F049A62, 0F48BD89, 91573E0A, DA718428, F8FDFA3E, 734D6913
6F049A62, 3AC6B69F, 75239882, 5FE127D0, 93287E7B, 734D6913, 2A5224A6, 91573E0A, C610A369, F8FDFA3E
93287E7B, 0B7C24AC, 3AC6B69F, 8E6209D4, 5FE127D0, F8FDFA3E, 8128FFB7, 2A5224A6, 5CF82A45, C610A369
5FE127D0, 2854DCE0, 0B7C24AC, 1ADA7CEB, 8E6209D4, C610A369, FF374DFD, 8128FFB7, 489298A9, 5CF82A45
8E6209D4, 267080E2, 2854DCE0, F092B02D, 1ADA7CEB, 5CF82A45, C5E0CCD7, FF374DFD, A3FEDE04, 489298A9
1ADA7CEB, 7806D96F, 267080E2, 537380A1, F092B02D, 489298A9, 31860C44, C5E0CCD7, DD37F7FC, A3FEDE04
F092B02D, 52638496, 7806D96F, C2038899, 537380A1, A3FEDE04, CEE7092B, 31860C44, 83335F17, DD37F7FC
537380A1, 59FC5CDB, 52638496, 1B65BDE0, C2038899, DD37F7FC, 46827AAE, CEE7092B, 183110C6, 83335F17
C2038899, 8AE30FBE, 59FC5CDB, 8E125949, 1B65BDE0, 83335F17, A757A907, 46827AAE, 9C24AF3B, 183110C6
1B65BDE0, 4F4AEBED, 8AE30FBE, F1736D67, 8E125949, 183110C6, E90F38FC, A757A907, 09EAB91A, 9C24AF3B
8E125949, 65BBCCCC, 4F4AEBED, 8C3EFA2B, F1736D67, 9C24AF3B, EC65CB85, E90F38FC, 5EA41E9D, 09EAB91A
F1736D67, 0B3B88C1, 65BBCCCC, 2BAFB53D, 8C3EFA2B, 09EAB91A, 54B06FBD, EC65CB85, 3CE3F3A4, 5EA41E9D
8C3EFA2B, 6DF30989, 0B3B88C1, EF333196, 2BAFB53D, 5EA41E9D, D8D6F0E3, 54B06FBD, 972E17B1, 3CE3F3A4
2BAFB53D, 156421AC, 6DF30989, EE23042C, EF333196, 3CE3F3A4, B30DA892, D8D6F0E3, C1BEF552, 972E17B1
EF333196, 6F54F9CA, 156421AC, CC2625B7, EE23042C, 972E17B1, F526A85A, B30DA892, 5BC38F63, C1BEF552
EE23042C, A5D28921, 6F54F9CA, 9086B055, CC2625B7, C1BEF552, 5F5587DB, F526A85A, 36A24ACC, 5BC38F63
CC2625B7, 2959D915, A5D28921, 53E729BD, 9086B055, 5BC38F63, 9FABAC24, 5F5587DB, 9AA16BD4, 36A24ACC
9086B055, 4EFF0384, 2959D915, 4A248697, 53E729BD, 36A24ACC, 52E4FB9B, 9FABAC24, 561F6D7D, 9AA16BD4
53E729BD, 17292945, 4EFF0384, 676454A5, 4A248697, 9AA16BD4, E13C3BDA, 52E4FB9B, AEB0927E, 561F6D7D
4A248697, 5FE71F22, 17292945, FC0E113B, 676454A5, 561F6D7D, 71244E49, E13C3BDA, 93EE6D4B, AEB0927E
676454A5, DC06A80F, 5FE71F22, A4A5145C, FC0E113B, AEB0927E, AA49234C, 71244E49, F0EF6B84, 93EE6D4B
FC0E113B, 5BD21FC5, DC06A80F, 9C7C897F, A4A5145C, 93EE6D4B, 42532D95, AA49234C, 913925C4, F0EF6B84
A4A5145C, 5587BC4F, 5BD21FC5, 1AA03F70, 9C7C897F, F0EF6B84, CDA86FD0, 42532D95, 248D32A9, 913925C4
9C7C897F, A1755F6B, 5587BC4F, 487F156F, 1AA03F70, 913925C4, 69C12F76, CDA86FD0, 4CB65509, 248D32A9
1AA03F70, 100A6B19, A1755F6B, 1EF13D56, 487F156F, 248D32A9, 44272219, 69C12F76, A1BF4336, 4CB65509
487F156F, AA2CFD07, 100A6B19, D57DAE85, 1EF13D56, 4CB65509, CBD360C3, 44272219, 04BDD9A7, A1BF4336
1EF13D56, 28246D22, AA2CFD07, 29AC6440, D57DAE85, A1BF4336, 27A64C2D, CBD360C3, 9C886510, 04BDD9A7
D57DAE85, 4909C2BD, 28246D22, B3F41EA8, 29AC6440, 04BDD9A7, CCB70B88, 27A64C2D, 4D830F2F, 9C886510
29AC6440, 9020271B, 4909C2BD, 91B488A0, B3F41EA8, 9C886510, 2020C0FC, CCB70B88, 9930B49E, 4D830F2F
B3F41EA8, A557D838, 9020271B, 270AF524, 91B488A0, 4D830F2F, 7541E108, 2020C0FC, DC2E2332, 9930B49E
91B488A0, F879D1F8, A557D838, 809C6E40, 270AF524, 9930B49E, 0A66EBF9, 7541E108, 8303F080, DC2E2332
270AF524, 39BAC08A, F879D1F8, 5F60E295, 809C6E40, DC2E2332, A0AB24D8, 0A66EBF9, 078421D5, 8303F080
809C6E40, DF212B9C, 39BAC08A, E747E3E1, 5F60E295, 8303F080, 44C068DD, A0AB24D8, 9BAFE429, 078421D5

5F60E295, 46F2CD86, DF212B9C, EB0228E6, E747E3E1, 078421D5, 3F8B3B48, 44C068DD, AC936282, 9BAFE429
E747E3E1, A17766F4, 46F2CD86, 84AE737C, EB0228E6, 9BAFE429, 873A41C4, 3F8B3B48, 01A37513, AC936282
EB0228E6, FC20AA01, A17766F4, CB36191B, 84AE737C, AC936282, A2969EB4, 873A41C4, 2CED20FE, 01A37513
84AE737C, 93A30DD9, FC20AA01, DD9BD285, CB36191B, 01A37513, 7B345F4F, A2969EB4, E907121C, 2CED20FE
CB36191B, 98554E1C, 93A30DD9, 82A807F0, DD9BD285, 2CED20FE, 07B2EA78, 7B345F4F, 5A7AD28A, E907121C
DD9BD285, 79D46BD1, 98554E1C, 8C37664E, 82A807F0, E907121C, 93451653, 07B2EA78, D17D3DEC, 5A7AD28A
82A807F0, 5FBC55DB, 79D46BD1, 55387261, 8C37664E, 5A7AD28A, AA0DF949, 93451653, CBA9E01E, D17D3DEC
8C37664E, DEF23A3B, 5FBC55DB, 51AF45E7, 55387261, D17D3DEC, 030FFB9A, AA0DF949, 14594E4D, CBA9E01E
55387261, 287DB1EB, DEF23A3B, F1576D7E, 51AF45E7, CBA9E01E, 0D9CD217, 030FFB9A, 37E526A8, 14594E4D
51AF45E7, CF955B8E, 287DB1EB, C8E8EF7B, F1576D7E, 14594E4D, BECE1BBD, 0D9CD217, 3FEE680C, 37E526A8
F1576D7E, 83B6B7E8, CF955B8E, F6C7ACA1, C8E8EF7B, 37E526A8, D97CFEEC, BECE1BBD, 73485C36, 3FEE680C
C8E8EF7B, 7943C443, 83B6B7E8, 556E3B3E, F6C7ACA1, 3FEE680C, DBEA79F5, D97CFEEC, 386EF6FB, 73485C36
F6C7ACA1, F336AA45, 7943C443, DADFA20E, 556E3B3E, 73485C36, 91704BDB, DBEA79F5, F3FBB365, 386EF6FB
556E3B3E, 2FF847D6, F336AA45, 0F110DE5, DADFA20E, 386EF6FB, 40CBA97D, 91704BDB, A9E7D76F, F3FBB365
DADFA20E, 33FE64C9, 2FF847D6, DAA917CC, 0F110DE5, F3FBB365, B0BD2456, 40CBA97D, C12F6E45, A9E7D76F
0F110DE5, 78378FE9, 33FE64C9, E11F58BF, DAA917CC, A9E7D76F, CA09D415, B0BD2456, 2EA5F503, C12F6E45

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X'_0, X'_1, X'_2, X'_3, X'_4$ که در طول پردازش دومین بلوک به‌دست آمده، آورده شده است.

52720555, 3B09A402, 94C343B1, 9CEDC3EA, 9039D740, 52720555, 3B09A402, 94C343B1, 9CEDC3EA, 9039D740
9039D740, 59874B6C, 3B09A402, 0D0EC653, 9CEDC3EA, 9039D740, 7FA6C9AF, 3B09A402, 0D0EC653, 9CEDC3EA
9CEDC3EA, 1D0D43D8, 59874B6C, 269008EC, 0D0EC653, 9CEDC3EA, 149F92B4, 7FA6C9AF, 269008EC, 0D0EC653
0D0EC653, EF3045D6, 1D0D43D8, 1D2DB166, 269008EC, 0D0EC653, 0E887E05, 149F92B4, 9B26BDFE, 269008EC
269008EC, 1E6BC8AD, EF3045D6, 350F6074, 1D2DB166, 269008EC, 6E8757AC, 0E887E05, 7E4AD052, 9B26BDFE
1D2DB166, 79CC70E3, 1E6BC8AD, C1175BBC, 350F6074, 9B26BDFE, 32C1290B, 6E8757AC, 21F8143A, 7E4AD052
350F6074, 13A4B937, 79CC70E3, AF22B479, C1175BBC, 7E4AD052, 8EB02C5A, 32C1290B, 1D5EB1BA, 21F8143A
C1175BBC, EE066CB9, 13A4B937, 31C38DE7, AF22B479, 21F8143A, 719EB9D9, 8EB02C5A, 04A42CCB, 1D5EB1BA
AF22B479, A08AFF93, EE066CB9, 92E4DC4E, 31C38DE7, 1D5EB1BA, 3D5B8A9A, 719EB9D9, C0B16A3A, 04A42CCB
31C38DE7, 89E27A43, A08AFF93, 19B2E7B8, 92E4DC4E, 04A42CCB, 47DEA0A3, 3D5B8A9A, 7AE765C6, C0B16A3A
92E4DC4E, 50EEC8A1, 89E27A43, 2BFE4E82, 19B2E7B8, C0B16A3A, A6AACEE1, 47DEA0A3, 6E2A68F5, 7AE765C6
19B2E7B8, 0FDE892D, 50EEC8A1, 89E90E27, 2BFE4E82, 7AE765C6, 4456D048, A6AACEE1, 7A828D1F, 6E2A68F5
2BFE4E82, 47B046C8, 0FDE892D, BB228543, 89E90E27, 6E2A68F5, 072D166E, 4456D048, AB3B869A, 7A828D1F
89E90E27, 5C8F582E, 47B046C8, 7A24B43F, BB228543, 7A828D1F, B37A11D1, 072D166E, 5B412111, AB3B869A
BB228543, 3D7F05B8, 5C8F582E, C11B211E, 7A24B43F, AB3B869A, 654CBE94, B37A11D1, B459B81C, 5B412111
7A24B43F, 962BCAF7, 3D7F05B8, 3D60B972, C11B211E, 5B412111, 6AFF9ABA, 654CBE94, E84746CD, B459B81C
C11B211E, 1A459D2E, 962BCAF7, FC16E0F5, 3D60B972, B459B81C, EE0E390E, 6AFF9ABA, 32FA5195, E84746CD
3D60B972, 1622907A, 1A459D2E, AF2BDE58, FC16E0F5, E84746CD, 569023C2, EE0E390E, FE6AE9AB, 32FA5195
FC16E0F5, B75B2E49, 1622907A, 1674B869, AF2BDE58, 32FA5195, 5C2944E8, 569023C2, 38E43BB8, FE6AE9AB
AF2BDE58, 6F16D4C4, B75B2E49, 8A41E858, 1674B869, FE6AE9AB, 103CE067, 5C2944E8, 408F095A, 38E43BB8
1674B869, 46FDEE89, 6F16D4C4, 6CB926DD, 8A41E858, 38E43BB8, AB641473, 103CE067, A513A170, 408F095A
8A41E858, E9F89F50, 46FDEE89, 5B5311BC, 6CB926DD, 408F095A, 25643DBF, AB641473, F3819C40, A513A170
6CB926DD, EC9A614C, E9F89F50, F7BA251B, 5B5311BC, A513A170, E60A5336, 25643DBF, 9051CEAD, F3819C40
5B5311BC, D525F69D, EC9A614C, E27D43A7, F7BA251B, F3819C40, FF4D318D, E60A5336, 90F6FC95, 9051CEAD
F7BA251B, EDFBF331, D525F69D, 698533B2, E27D43A7, 9051CEAD, 6D5A28DD, FF4D318D, 294CDB98, 90F6FC95
E27D43A7, 93C5E732, EDFBF331, 97DA7754, 698533B2, 90F6FC95, 855C140A, 6D5A28DD, 34C637FD, 294CDB98
698533B2, 24907FDF, 93C5E732, EFCCC7B7, 97DA7754, 294CDB98, 79C1BC35, 855C140A, 68A375B5, 34C637FD
97DA7754, E2193F3E, 24907FDF, 179CCA4F, EFCCC7B7, 34C637FD, B2D5EF34, 79C1BC35, 70502A15, 68A375B5
EFCCC7B7, D3AD6006, E2193F3E, 41FF7C92, 179CCA4F, 68A375B5, DB87209A, B2D5EF34, 06F0D5E7, 70502A15
179CCA4F, 6B8BFAB4, D3AD6006, 64FCFB88, 41FF7C92, 70502A15, 4DEC84F2, DB87209A, 57BCD2CB, 06F0D5E7

41FF7C92, 5052D6EF, 6B8BFAB4, B5801B4E, 64FCFB88, 06F0D5E7, D4F6A30D, 4DEC84F2, 1C826B6E, 57BCD2CB
64FCFB88, FF36EBC8, 5052D6EF, 2FEAD1AE, B5801B4E, 57BCD2CB, 0191C9F0, D4F6A30D, B213C937, 1C826B6E
B5801B4E, 5A010C53, FF36EBC8, 4B5BBD41, 2FEAD1AE, 1C826B6E, 20FBAB36, 0191C9F0, DA8C3753, B213C937
2FEAD1AE, 952BFB5D, 5A010C53, DBAF23FC, 4B5BBD41, B213C937, 7E796493, 20FBAB36, 4727C006, DA8C3753
4B5BBD41, FE05BEE3, 952BFB5D, 04314D68, DBAF23FC, DA8C3753, C9EABB3E, 7E796493, EEACD883, 4727C006
DBAF23FC, 2256AF69, FE05BEE3, AFED7654, 04314D68, 4727C006, B44977A5, C9EABB3E, E5924DF9, EEACD883
04314D68, 5285B0D3, 2256AF69, 16FB8FF8, AFED7654, EEACD883, 287580C6, B44977A5, AAECFB27, E5924DF9
AFED7654, 1DFB856C, 5285B0D3, 5ABDA489, 16FB8FF8, E5924DF9, 1E1DBD16, 287580C6, 25DE96D1, AAECFB27
16FB8FF8, 32974404, 1DFB856C, 16C34D4A, 5ABDA489, AAECFB27, FBEB21BA, 1E1DBD16, D60318A1, 25DE96D1
5ABDA489, 90AC71CE, 32974404, EE15B077, 16C34D4A, 25DE96D1, B74BF3E2, FBEB21BA, 76F45878, D60318A1
16C34D4A, 849CCC12, 90AC71CE, 5D1010CA, EE15B077, D60318A1, 755BEDDF, B74BF3E2, AC86EBEF, 76F45878
EE15B077, 340EBE92, 849CCC12, B1C73A42, 5D1010CA, 76F45878, 3CD099C6, 755BEDDF, 2FCF8ADD, AC86EBEF
5D1010CA, F531E5F5, 340EBE92, 73304A12, B1C73A42, AC86EBEF, A19BBAA2, 3CD099C6, 6FB77DD5, 2FCF8ADD
B1C73A42, 27528557, F531E5F5, 3AFA48D0, 73304A12, 2FCF8ADD, EFC554F1, A19BBAA2, 426718F3, 6FB77DD5
73304A12, E4AFA69F, 27528557, C797D7D4, 3AFA48D0, 6FB77DD5, F56F1485, EFC554F1, 6EEA8A86, 426718F3
3AFA48D0, E3462C93, E4AFA69F, 4A155C9D, C797D7D4, 426718F3, E0A1480A, F56F1485, 1553C7BF, 6EEA8A86
C797D7D4, 3CF5CD85, E3462C93, BE9A7F92, 4A155C9D, 6EEA8A86, 9F80007D, E0A1480A, BC5217D5, 1553C7BF
4A155C9D, B6C756F9, 3CF5CD85, 18B24F8D, BE9A7F92, 1553C7BF, 090898BE, 9F80007D, 85202B82, BC5217D5
BE9A7F92, CC2AB627, B6C756F9, D73614F3, 18B24F8D, BC5217D5, A0CD75A2, 090898BE, 0001F67E, 85202B82
18B24F8D, E5471921, CC2AB627, 1D5BE6DB, D73614F3, 85202B82, 95FE46E6, A0CD75A2, 2262F824, 0001F67E
D73614F3, E8FEFBC6, E5471921, AAD89F30, 1D5BE6DB, 0001F67E, 4B55D832, 95FE46E6, 35D68A83, 2262F824
1D5BE6DB, 788FFBE7, E8FEFBC6, 1C648795, AAD89F30, 2262F824, 681302D4, 4B55D832, F91B9A57, 35D68A83
AAD89F30, FA97F1BB, 788FFBE7, FBEF1BA3, 1C648795, 35D68A83, 860F8E32, 681302D4, 5760C92D, F91B9A57
1C648795, 2FE154B4, FA97F1BB, 3FEF9DE2, FBEF1BA3, F91B9A57, CA3DDAC0, 860F8E32, 4C0B51A0, 5760C92D
FBEF1BA3, D884695B, 2FE154B4, 5FC6EFEA, 3FEF9DE2, 5760C92D, 7E790793, CA3DDAC0, 3E38CA18, 4C0B51A0
3FEF9DE2, A09357E9, D884695B, 8552D0BF, 5FC6EFEA, 4C0B51A0, 4E0DF927, 7E790793, F76B0328, 3E38CA18
5FC6EFEA, 019B9791, A09357E9, 11A56F62, 8552D0BF, 3E38CA18, 311DFB90, 4E0DF927, E41E4DF9, F76B0328
8552D0BF, 70DB6FDF, 019B9791, 4D5FA682, 11A56F62, F76B0328, 24FA9DC7, 311DFB90, 37E49D38, E41E4DF9
11A56F62, 82F104B4, 70DB6FDF, 6E5E4406, 4D5FA682, E41E4DF9, CE45E142, 24FA9DC7, 77EE40C4, 37E49D38
4D5FA682, BFAB29F8, 82F104B4, 6DBF7DC3, 6E5E4406, 37E49D38, 9C4F267F, CE45E142, EA771C93, 77EE40C4
6E5E4406, 880198A9, BFAB29F8, C412D20B, 6DBF7DC3, 77EE40C4, 06880805, 9C4F267F, 17850B39, EA771C93
6DBF7DC3, 917C197C, 880198A9, ACA7E2FE, C412D20B, EA771C93, 7625BD09, 06880805, 3C99FE71, 17850B39
C412D20B, 03E7992A, 917C197C, 0662A620, ACA7E2FE, 17850B39, 8720C8E7, 7625BD09, 2020141A, 3C99FE71
ACA7E2FE, 824CEF7A, 03E7992A, F065F245, 0662A620, 3C99FE71, CBB7DA7A, 8720C8E7, 96F425D8, 2020141A
0662A620, AF16F218, 824CEF7A, 9E64A80F, F065F245, 2020141A, 88851068, CBB7DA7A, 83239E1C, 96F425D8
F065F245, EFC8943D, AF16F218, 33BDEA09, 9E64A80F, 96F425D8, C85C4EB8, 88851068, DF69EB2E, 83239E1C
9E64A80F, C80FF53B, EFC8943D, 5BC862BC, 33BDEA09, 83239E1C, 57BF18E2, C85C4EB8, 1441A222, DF69EB2E
33BDEA09, 28DF9E36, C80FF53B, 2250F7BF, 5BC862BC, DF69EB2E, 48932C1A, 57BF18E2, 713AE321, 1441A222
5BC862BC, 6E1D8950, 28DF9E36, 3FD4EF20, 2250F7BF, 1441A222, 15C7B0BD, 48932C1A, FC63895E, 713AE321
2250F7BF, 21EEE621, 6E1D8950, 7E78D8A3, 3FD4EF20, 713AE321, FCBC9E78, 15C7B0BD, 4CB06922, FC63895E
3FD4EF20, 561379BA, 21EEE621, 762541B8, 7E78D8A3, FC63895E, DD28EA60, FCBC9E78, 1EC2F457, 4CB06922
7E78D8A3, 4D0255C5, 561379BA, BB988487, 762541B8, 4CB06922, CF1BB810, DD28EA60, F279E3F2, 1EC2F457
762541B8, 966845EC, 4D0255C5, 4DE6E958, BB988487, 1EC2F457, 5D899D62, CF1BB810, A3A98374, F279E3F2
BB988487, D922DEB8, 966845EC, 09571534, 4DE6E958, F279E3F2, F1144141, 5D899D62, 6EE0433C, A3A98374
4DE6E958, B919B2A3, D922DEB8, A117B259, 09571534, A3A98374, 940BBA12, F1144141, 26758976, 6EE0433C
09571534, D3CF80F9, B919B2A3, 8B7AE364, A117B259, 6EE0433C, 33DDA9B5, 940BBA12, 510507C4, 26758976
A117B259, F548EA98, D3CF80F9, 66CA8EE4, 8B7AE364, 26758976, DCE0B562, 33DDA9B5, 2EE84A50, 510507C4
8B7AE364, A1D3372D, F548EA98, 3E03E74F, 66CA8EE4, 510507C4, C103FBE9, DCE0B562, 76A6D4CF, 2EE84A50
66CA8EE4, 6578D66C, A1D3372D, 23AA63D5, 3E03E74F, 2EE84A50, 832961D9, C103FBE9, 82D58B73, 76A6D4CF
3E03E74F, 57C29604, 6578D66C, 4CDCB687, 23AA63D5, 76A6D4CF, B183744E, 832961D9, 0FEFA704, 82D58B73
23AA63D5, 27F5E937, 57C29604, E359B195, 4CDCB687, 82D58B73, E710A112, B183744E, A587660C, 0FEFA704

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
12 A0 53 38 4A 9C 0C 88 E4 05 A0 6C 27 DC F4 9A DA 62 EB 2B
```

الف-۱-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای 10^6 بار تکرار می‌شود.

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
52 78 32 43 C1 69 7B DB E1 6D 37 F9 7F 68 F0 83 25 DC 15 28
```

الف-۱-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سر خط[1] بعد از اولین n)

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
6f 3f a3 9b 6b 50 3c 38 4f 91 9a 49 a7 aa 5c 2c 08 bd fb 45
```

الف-۱-۱۱ مثال ۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
94 c2 64 11 54 04 e6 33 79 0d fc c8 7b 58 7d 36 77 06 7d 9f
```

الف-۲ تابع درهم‌ساز اختصاصی ۲

الف-۲-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
CD F2 62 13 A1 50 DC 3E CB 61 0F 18 F6 B3 8B 46
```

1- Line break

الف-۲-۲ مثال ۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ‘a’ .

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
86 BE 7A FA 33 9D 0F C7 CF C7 85 E7 2F 57 8D 33
```

الف-۲-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ‘abc’ است. این معادل رشته بیت ‘01100001 01100010 01100011’ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
80636261 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000018 00000000
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X'_0, X'_1, X'_2, X'_3$ آمده است.

```
67452301, EFCDAB89, 98BADCFE, 10325476, 67452301, EFCDAB89, 98BADCFE, 10325476
10325476, 6D431A77, EFCDAB89, 98BADCFE, 10325476, 70376F40, EFCDAB89, 98BADCFE
98BADCFE, B05D8A99, 6D431A77, EFCDAB89, 98BADCFE, 989F6BB0, 70376F40, EFCDAB89
EFCDAB89, 0C32E5C7, B05D8A99, 6D431A77, EFCDAB89, 39B14904, 989F6BB0, 70376F40
6D431A77, A20B2C0F, 0C32E5C7, B05D8A99, 70376F40, 671C03CC, 39B14904, 989F6BB0
B05D8A99, 74EBB911, A20B2C0F, 0C32E5C7, 989F6BB0, BFD55C42, 671C03CC, 39B14904
0C32E5C7, 2FFB728B, 74EBB911, A20B2C0F, 39B14904, A12F346F, BFD55C42, 671C03CC
A20B2C0F, A766AE02, 2FFB728B, 74EBB911, 671C03CC, 989C2210, A12F346F, BFD55C42
74EBB911, 03234F3D, A766AE02, 2FFB728B, BFD55C42, 0F95FBEA, 989C2210, A12F346F
2FFB728B, 52662805, 03234F3D, A766AE02, A12F346F, 068D5115, 0F95FBEA, 989C2210
A766AE02, E778A4C3, 52662805, 03234F3D, 989C2210, AFCD27FC, 068D5115, 0F95FBEA
03234F3D, 1C7F5769, E778A4C3, 52662805, 0F95FBEA, CBD1F3F8, AFCD27FC, 068D5115
52662805, 95765642, 1C7F5769, E778A4C3, 068D5115, CFFE405F, CBD1F3F8, AFCD27FC
E778A4C3, 35F37B70, 95765642, 1C7F5769, AFCD27FC, 2B55C9C3, CFFE405F, CBD1F3F8
1C7F5769, 398F8F52, 35F37B70, 95765642, CBD1F3F8, DD6A43FB, 2B55C9C3, CFFE405F
95765642, 13F3C36B, 398F8F52, 35F37B70, CFFE405F, 049B909E, DD6A43FB, 2B55C9C3
35F37B70, 058D8BB5, 13F3C36B, 398F8F52, 2B55C9C3, 3713BFFD, 049B909E, DD6A43FB
398F8F52, FCBE3664, 058D8BB5, 13F3C36B, DD6A43FB, 82ADDB53, 3713BFFD, 049B909E
```

13F3C36B, F7F306A6, FCBE3664, 058D8BB5, 049B909E, CC1D8105, 82ADDB53, 3713BFFD

058D8BB5, 34CC3963, F7F306A6, FCBE3664, 3713BFFD, BE09159A, CC1D8105, 82ADDB53

FCBE3664, 416E8BA0, 34CC3963, F7F306A6, 82ADDB53, 541AE568, BE09159A, CC1D8105

F7F306A6, EDE91870, 416E8BA0, 34CC3963, CC1D8105, 27D40F94, 541AE568, BE09159A

34CC3963, C352C547, EDE91870, 416E8BA0, BE09159A, 675C363A, 27D40F94, 541AE568

416E8BA0, 5D5EEE28, C352C547, EDE91870, 541AE568, 77F3A38B, 675C363A, 27D40F94

EDE91870, 6CC4BEF2, 5D5EEE28, C352C547, 27D40F94, 84D73C44, 77F3A38B, 675C363A

C352C547, E140970B, 6CC4BEF2, 5D5EEE28, 675C363A, D2958F37, 84D73C44, 77F3A38B

5D5EEE28, 79F631A9, E140970B, 6CC4BEF2, 77F3A38B, FC39C927, D2958F37, 84D73C44

6CC4BEF2, 038E0E91, 79F631A9, E140970B, 84D73C44, E3A5A4DE, FC39C927, D2958F37

E140970B, 1B942D52, 038E0E91, 79F631A9, D2958F37, 4BA3A889, E3A5A4DE, FC39C927

79F631A9, 496AECFD, 1B942D52, 038E0E91, FC39C927, A964BA74, 4BA3A889, E3A5A4DE

038E0E91, FE6CD56F, 496AECFD, 1B942D52, E3A5A4DE, 7AF9DBB0, A964BA74, 4BA3A889

1B942D52, 2E94F501, FE6CD56F, 496AECFD, 4BA3A889, 7DA68EA9, 7AF9DBB0, A964BA74

496AECFD, 584E8E58, 2E94F501, FE6CD56F, A964BA74, 9C7247E5, 7DA68EA9, 7AF9DBB0

FE6CD56F, 41A17EFA, 584E8E58, 2E94F501, 7AF9DBB0, 0130312B, 9C7247E5, 7DA68EA9

2E94F501, 8981C6CD, 41A17EFA, 584E8E58, 7DA68EA9, 90552232, 0130312B, 9C7247E5

584E8E58, 400A93E1, 8981C6CD, 41A17EFA, 9C7247E5, 99C1FBA4, 90552232, 0130312B

41A17EFA, 841F817F, 400A93E1, 8981C6CD, 0130312B, 9D481CD2, 99C1FBA4, 90552232

8981C6CD, 659379BE, 841F817F, 400A93E1, 90552232, F5AABE07, 9D481CD2, 99C1FBA4

400A93E1, AB3D9A70, 659379BE, 841F817F, 99C1FBA4, C3AFB7E6, F5AABE07, 9D481CD2

841F817F, D3D21DC8, AB3D9A70, 659379BE, 9D481CD2, 473E2B79, C3AFB7E6, F5AABE07

659379BE, 38C8D29D, D3D21DC8, AB3D9A70, F5AABE07, C4CAFF99, 473E2B79, C3AFB7E6

AB3D9A70, 738B9B0F, 38C8D29D, D3D21DC8, C3AFB7E6, A2879AA4, C4CAFF99, 473E2B79

D3D21DC8, 8528B83E, 738B9B0F, 38C8D29D, 473E2B79, 56565EDB, A2879AA4, C4CAFF99

38C8D29D, 7345AF18, 8528B83E, 738B9B0F, C4CAFF99, E7A4BD86, 56565EDB, A2879AA4

738B9B0F, FFCCC52B, 7345AF18, 8528B83E, A2879AA4, 974B9E10, E7A4BD86, 56565EDB

8528B83E, A77E902B, FFCCC52B, 7345AF18, 56565EDB, 96CC5AE1, 974B9E10, E7A4BD86

7345AF18, CB9C6C83, A77E902B, FFCCC52B, E7A4BD86, 57E6A772, 96CC5AE1, 974B9E10

FFCCC52B, 38A2DA83, CB9C6C83, A77E902B, 974B9E10, F10B6CF5, 57E6A772, 96CC5AE1

A77E902B, 487F9401, 38A2DA83, CB9C6C83, 96CC5AE1, 90426E6B, F10B6CF5, 57E6A772

CB9C6C83, C7184576, 487F9401, 38A2DA83, 57E6A772, 0066E6BE, 90426E6B, F10B6CF5

```
38A2DA83, 56D619B1, C7184576, 487F9401, F10B6CF5, 22D17257, 0066E6BE, 90426E6B
487F9401, 3A35A3C5, 56D619B1, C7184576, 90426E6B, 016777A4, 22D17257, 0066E6BE
C7184576, B5517538, 3A35A3C5, 56D619B1, 0066E6BE, 9A8DC5A0, 016777A4, 22D17257
56D619B1, 4609C4C2, B5517538, 3A35A3C5, 22D17257, A9C46E68, 9A8DC5A0, 016777A4
3A35A3C5, D5C2B699, 4609C4C2, B5517538, 016777A4, 13B0D540, A9C46E68, 9A8DC5A0
B5517538, 342AF741, D5C2B699, 4609C4C2, 9A8DC5A0, 983D8B08, 13B0D540, A9C46E68
4609C4C2, 38286DDA, 342AF741, D5C2B699, A9C46E68, 96084F4E, 983D8B08, 13B0D540
D5C2B699, 9BCEEC0A, 38286DDA, 342AF741, 13B0D540, D25FDBB1, 96084F4E, 983D8B08
342AF741, 5803DF3A, 9BCEEC0A, 38286DDA, 983D8B08, 35EA6FE0, D25FDBB1, 96084F4E
38286DDA, E1B026EB, 5803DF3A, 9BCEEC0A, 96084F4E, B862709F, 35EA6FE0, D25FDBB1
9BCEEC0A, 31587C22, E1B026EB, 5803DF3A, D25FDBB1, C02839EB, B862709F, 35EA6FE0
5803DF3A, 9B25E1DC, 31587C22, E1B026EB, 35EA6FE0, 00245200, C02839EB, B862709F
E1B026EB, 2205379E, 9B25E1DC, 31587C22, B862709F, CB116A95, 00245200, C02839EB
31587C22, 5E3334A3, 2205379E, 9B25E1DC, C02839EB, B90EE1BF, CB116A95, 00245200
9B25E1DC, 56F80FA9, 5E3334A3, 2205379E, 00245200, 64132D32, B90EE1BF, CB116A95
```

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
C1 4A 12 19 9C 66 E4 BA 84 63 6B 0F 69 14 4C 77
```

الف-۲-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´
کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
9E 32 7B 3D 6E 52 30 62 AF C1 13 2D 7D F9 D1 B8
```

الف-۲-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII
´abcdefghijklmnopqrstuvwxyz´
کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
FD 2A A6 07 F7 1D C8 F5 10 71 49 22 B3 71 83 4E
```

الف-۲-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
D1 E9 59 EB 17 9C 91 1F AE A4 62 4C 60 C5 C7 02
```

الف-۲-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار ‘1234567890’

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
3F 45 EF 19 47 32 C2 DB B2 C4 A2 C7 69 79 5F A3
```

الف-۲-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
64636261 65646362 66656463 67666564 68676665 69686766 6A696867 6B6A6968
6C6B6A69 6D6C6B6A 6E6D6C6B 6F6E6D6C 706F6E6D 71706F6E 00000080 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 000001C0 00000000
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X'_0, X'_1, X'_2, X'_3$ که در طول پردازش اولین بلوک به‌دست آمده، آورده شده است.

```
67452301, EFCDAB89, 98BADCFE, 10325476, 67452301, EFCDAB89, 98BADCFE, 10325476
10325476, 6D431997, EFCDAB89, 98BADCFE, 10325476, D89ED5A9, EFCDAB89, 98BADCFE
98BADCFE, C9AE23F2, 6D431997, EFCDAB89, 98BADCFE, 69B10AC1, D89ED5A9, EFCDAB89
EFCDAB89, 69A6A520, C9AE23F2, 6D431997, EFCDAB89, B661DB9C, 69B10AC1, D89ED5A9
6D431997, FB032247, 69A6A520, C9AE23F2, D89ED5A9, ABACC2AF, B661DB9C, 69B10AC1
C9AE23F2, 16C49226, FB032247, 69A6A520, 69B10AC1, D412CAD1, ABACC2AF, B661DB9C
69A6A520, 77A099B7, 16C49226, FB032247, B661DB9C, E2DEDF22, D412CAD1, ABACC2AF
FB032247, 3B9BAEB7, 77A099B7, 16C49226, ABACC2AF, CFB03688, E2DEDF22, D412CAD1
16C49226, DA61AB82, 3B9BAEB7, 77A099B7, D412CAD1, 72599389, CFB03688, E2DEDF22
77A099B7, 54C888CC, DA61AB82, 3B9BAEB7, E2DEDF22, CF3CD682, 72599389, CFB03688
3B9BAEB7, F2635347, 54C888CC, DA61AB82, CFB03688, B235784E, CF3CD682, 72599389
DA61AB82, E2CAC9B4, F2635347, 54C888CC, 72599389, 881678DF, B235784E, CF3CD682
```

54C888CC, 9596C718, E2CAC9B4, F2635347, CF3CD682, E815373B, 881678DF, B235784E

F2635347, 9DD54912, 9596C718, E2CAC9B4, B235784E, BD994B56, E815373B, 881678DF

E2CAC9B4, 2E8539A7, 9DD54912, 9596C718, 881678DF, B0055655, BD994B56, E815373B

9596C718, 2303C213, 2E8539A7, 9DD54912, E815373B, CC87EF5A, B0055655, BD994B56

9DD54912, EA79BE25, 2303C213, 2E8539A7, BD994B56, 6B24384D, CC87EF5A, B0055655

2E8539A7, 23D7CB45, EA79BE25, 2303C213, B0055655, 93E7329F, 6B24384D, CC87EF5A

2303C213, F028EF04, 23D7CB45, EA79BE25, CC87EF5A, 35B95AE7, 93E7329F, 6B24384D

EA79BE25, 48863F19, F028EF04, 23D7CB45, 6B24384D, 06C6536D, 35B95AE7, 93E7329F

23D7CB45, 514C81B6, 48863F19, F028EF04, 93E7329F, FF1C5DC7, 06C6536D, 35B95AE7

F028EF04, 6102CE67, 514C81B6, 48863F19, 35B95AE7, D0D541F1, FF1C5DC7, 06C6536D

48863F19, 330485FD, 6102CE67, 514C81B6, 06C6536D, A94C0DD9, D0D541F1, FF1C5DC7

514C81B6, 289E8C82, 330485FD, 6102CE67, FF1C5DC7, DEDC1E39, A94C0DD9, D0D541F1

6102CE67, 13CC3A1D, 289E8C82, 330485FD, D0D541F1, 12D926C0, DEDC1E39, A94C0DD9

330485FD, 40A226A6, 13CC3A1D, 289E8C82, A94C0DD9, ED7EDA63, 12D926C0, DEDC1E39

289E8C82, 70BFB1A8, 40A226A6, 13CC3A1D, DEDC1E39, 9E52219C, ED7EDA63, 12D926C0

13CC3A1D, CE1D1A37, 70BFB1A8, 40A226A6, 12D926C0, F5D22339, 9E52219C, ED7EDA63

40A226A6, EC9F7830, CE1D1A37, 70BFB1A8, ED7EDA63, 0BC5B4FC, F5D22339, 9E52219C

70BFB1A8, 3CF2D6EE, EC9F7830, CE1D1A37, 9E52219C, FCFBD391, 0BC5B4FC, F5D22339

CE1D1A37, F0C1F95C, 3CF2D6EE, EC9F7830, F5D22339, 2B6A389B, FCFBD391, 0BC5B4FC

EC9F7830, 9A351A9D, F0C1F95C, 3CF2D6EE, 0BC5B4FC, FBF85B05, 2B6A389B, FCFBD391

3CF2D6EE, 138B0685, 9A351A9D, F0C1F95C, FCFBD391, F7BBBE8B, FBF85B05, 2B6A389B

F0C1F95C, EA3574D1, 138B0685, 9A351A9D, 2B6A389B, C8592ACC, F7BBBE8B, FBF85B05

9A351A9D, 4719C849, EA3574D1, 138B0685, FBF85B05, FE2D3EFA, C8592ACC, F7BBBE8B

138B0685, 57F52A13, 4719C849, EA3574D1, F7BBBE8B, 5411CC34, FE2D3EFA, C8592ACC

EA3574D1, 4751F880, 57F52A13, 4719C849, C8592ACC, DC8ED546, 5411CC34, FE2D3EFA

4719C849, 80605BAF, 4751F880, 57F52A13, FE2D3EFA, 55C1E317, DC8ED546, 5411CC34

57F52A13, 1E53AD4A, 80605BAF, 4751F880, 5411CC34, 0B92E4F0, 55C1E317, DC8ED546

4751F880, 1ABEED79, 1E53AD4A, 80605BAF, DC8ED546, 5E192900, 0B92E4F0, 55C1E317

80605BAF, 75EACBB7, 1ABEED79, 1E53AD4A, 55C1E317, 186EB0CF, 5E192900, 0B92E4F0

1E53AD4A, 08AC1056, 75EACBB7, 1ABEED79, 0B92E4F0, 8F3A64E3, 186EB0CF, 5E192900

1ABEED79, 9BDB7A88, 08AC1056, 75EACBB7, 5E192900, 3701E7B3, 8F3A64E3, 186EB0CF

75EACBB7, ADF32F05, 9BDB7A88, 08AC1056, 186EB0CF, 6CE969E9, 3701E7B3, 8F3A64E3

08AC1056, 2277B80D, ADF32F05, 9BDB7A88, 8F3A64E3, EE7224D5, 6CE969E9, 3701E7B3

9BDB7A88, 535DBB9A, 2277B80D, ADF32F05, 3701E7B3, 3E849D0F, EE7224D5, 6CE969E9

ADF32F05, 2A494EC5, 535DBB9A, 2277B80D, 6CE969E9, DDBD8EE7, 3E849D0F, EE7224D5

2277B80D, 693C7A09, 2A494EC5, 535DBB9A, EE7224D5, C3DDAC40, DDBD8EE7, 3E849D0F

535DBB9A, 148A5796, 693C7A09, 2A494EC5, 3E849D0F, 5E0E10B9, C3DDAC40, DDBD8EE7

2A494EC5, D2932448, 148A5796, 693C7A09, DDBD8EE7, 1CCB75AF, 5E0E10B9, C3DDAC40

693C7A09, 39CA97B6, D2932448, 148A5796, C3DDAC40, 27F81499, 1CCB75AF, 5E0E10B9

148A5796, 770BCE98, 39CA97B6, D2932448, 5E0E10B9, 82843491, 27F81499, 1CCB75AF

D2932448, 8C4DC6AF, 770BCE98, 39CA97B6, 1CCB75AF, 4E4E13E9, 82843491, 27F81499

39CA97B6, 048CC517, 8C4DC6AF, 770BCE98, 27F81499, 03BD1BD9, 4E4E13E9, 82843491

770BCE98, 419960CF, 048CC517, 8C4DC6AF, 82843491, 6FA999B7, 03BD1BD9, 4E4E13E9

8C4DC6AF, 407700EE, 419960CF, 048CC517, 4E4E13E9, 37B18629, 6FA999B7, 03BD1BD9

048CC517, E60ABEC4, 407700EE, 419960CF, 03BD1BD9, 9EA44395, 37B18629, 6FA999B7

419960CF, 0E248A8B, E60ABEC4, 407700EE, 6FA999B7, F877D28C, 9EA44395, 37B18629

407700EE, 10667792, 0E248A8B, E60ABEC4, 37B18629, F63EA862, F877D28C, 9EA44395

E60ABEC4, 646BB7A8, 10667792, 0E248A8B, 9EA44395, 424072F0, F63EA862, F877D28C

0E248A8B, 625CCE22, 646BB7A8, 10667792, F877D28C, 3B7642B8, 424072F0, F63EA862

10667792, 8E0E1101, 625CCE22, 646BB7A8, F63EA862, CD620F4E, 3B7642B8, 424072F0

646BB7A8, C23D3583, 8E0E1101, 625CCE22, 424072F0, BFAA1A02, CD620F4E, 3B7642B8

625CCE22, 81DE3DC5, C23D3583, 8E0E1101, 3B7642B8, 1BA7FD36, BFAA1A02, CD620F4E

8E0E1101, D24E4181, 81DE3DC5, C23D3583, CD620F4E, E62BB2A4, 1BA7FD36, BFAA1A02

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X'_0, X'_1, X'_2, X'_3$ که در طول پردازش دومین بلوک به‌دست آمده، آورده شده است.

31560350, 285A21CF, 846C181B, 553B61B8, 31560350, 285A21CF, 846C181B, 553B61B8

553B61B8, 1ADDE153, 285A21CF, 846C181B, 553B61B8, 56C8C102, 285A21CF, 846C181B

846C181B, CE8FC309, 1ADDE153, 285A21CF, 846C181B, 702249A4, 56C8C102, 285A21CF

285A21CF, 0DD8403A, CE8FC309, 1ADDE153, 285A21CF, 22CB0A97, 702249A4, 56C8C102

1ADDE153, 4842F01E, 0DD8403A, CE8FC309, 56C8C102, 35B2DCDF, 22CB0A97, 702249A4

CE8FC309, BE6A9014, 4842F01E, 0DD8403A, 702249A4, D2EFFB4A, 35B2DCDF, 22CB0A97

0DD8403A, 7FE339CA, BE6A9014, 4842F01E, 22CB0A97, 59EA6C60, D2EFFB4A, 35B2DCDF

4842F01E, D1CCFD4B, 7FE339CA, BE6A9014, 35B2DCDF, 82DEA3AE, 59EA6C60, D2EFFB4A

BE6A9014, 108966B1, D1CCFD4B, 7FE339CA, D2EFFB4A, 4481FDE2, 82DEA3AE, 59EA6C60

7FE339CA, 899223E8, 108966B1, D1CCFD4B, 59EA6C60, 13BB8F73, 4481FDE2, 82DEA3AE

D1CCFD4B, 5E3B9917, 899223E8, 108966B1, 82DEA3AE, 946BD478, 13BB8F73, 4481FDE2

108966B1, 7666663B, 5E3B9917, 899223E8, 4481FDE2, BD0605EA, 946BD478, 13BB8F73

899223E8, A1BAD92C, 7666663B, 5E3B9917, 13BB8F73, 36F99153, BD0605EA, 946BD478

5E3B9917, DE527A04, A1BAD92C, 7666663B, 946BD478, EB4AE872, 36F99153, BD0605EA

7666663B, E52F1533, DE527A04, A1BAD92C, BD0605EA, 7C346442, EB4AE872, 36F99153

A1BAD92C, 5C3C2C22, E52F1533, DE527A04, 36F99153, AFA320AD, 7C346442, EB4AE872

DE527A04, FC1C4108, 5C3C2C22, E52F1533, EB4AE872, B4905651, AFA320AD, 7C346442

E52F1533, 0A03E84B, FC1C4108, 5C3C2C22, 7C346442, 02E94FA1, B4905651, AFA320AD

5C3C2C22, FB74BD26, 0A03E84B, FC1C4108, AFA320AD, E08D1799, 02E94FA1, B4905651

FC1C4108, C78DC5C4, FB74BD26, 0A03E84B, B4905651, 69AFAA80, E08D1799, 02E94FA1

0A03E84B, ACF60434, C78DC5C4, FB74BD26, 02E94FA1, FA665E46, 69AFAA80, E08D1799

FB74BD26, 58F751E0, ACF60434, C78DC5C4, E08D1799, 269AB7E3, FA665E46, 69AFAA80

C78DC5C4, EB75C7CB, 58F751E0, ACF60434, 69AFAA80, 0F06388B, 269AB7E3, FA665E46

ACF60434, 83C0A8B7, EB75C7CB, 58F751E0, FA665E46, FD44FBD5, 0F06388B, 269AB7E3

58F751E0, 27C87178, 83C0A8B7, EB75C7CB, 269AB7E3, DBBC0190, FD44FBD5, 0F06388B

EB75C7CB, B7B9163F, 27C87178, 83C0A8B7, 0F06388B, D0E3FC2B, DBBC0190, FD44FBD5

83C0A8B7, 0FA1C6DC, B7B9163F, 27C87178, FD44FBD5, 7D87B4BA, D0E3FC2B, DBBC0190

27C87178, 2CC60316, 0FA1C6DC, B7B9163F, DBBC0190, 68367FDB, 7D87B4BA, D0E3FC2B

B7B9163F, 08029C44, 2CC60316, 0FA1C6DC, D0E3FC2B, 53AB5439, 68367FDB, 7D87B4BA

0FA1C6DC, F693A10E, 08029C44, 2CC60316, 7D87B4BA, E78B75B5, 53AB5439, 68367FDB

2CC60316, 356224B9, F693A10E, 08029C44, 68367FDB, 830530DF, E78B75B5, 53AB5439

08029C44, 669F7869, 356224B9, F693A10E, 53AB5439, 67FCB1AC, 830530DF, E78B75B5

F693A10E, 7B70C168, 669F7869, 356224B9, E78B75B5, 757BB243, 67FCB1AC, 830530DF

356224B9, 037FB19C, 7B70C168, 669F7869, 830530DF, F0CA8878, 757BB243, 67FCB1AC

669F7869, 9B0A10B3, 037FB19C, 7B70C168, 67FCB1AC, FA10CB33, F0CA8878, 757BB243

7B70C168, 9D015956, 9B0A10B3, 037FB19C, 757BB243, 5487E56C, FA10CB33, F0CA8878

037FB19C, 6A7DE5F4, 9D015956, 9B0A10B3, F0CA8878, A5D33699, 5487E56C, FA10CB33

9B0A10B3, E522D913, 6A7DE5F4, 9D015956, FA10CB33, BEB495BC, A5D33699, 5487E56C

9D015956, 0EFD42E5, E522D913, 6A7DE5F4, 5487E56C, 05202F93, BEB495BC, A5D33699

6A7DE5F4, 7902100B, 0EFD42E5, E522D913, A5D33699, BACE7DD9, 05202F93, BEB495BC

E522D913, 1ACEFABC, 7902100B, 0EFD42E5, BEB495BC, 08D045DD, BACE7DD9, 05202F93

0EFD42E5, E07378FF, 1ACEFABC, 7902100B, 05202F93, 5448A3A0, 08D045DD, BACE7DD9

7902100B, 489C7A1A, E07378FF, 1ACEFABC, BACE7DD9, D98BE3AA, 5448A3A0, 08D045DD

1ACEFABC, C02A45A5, 489C7A1A, E07378FF, 08D045DD, 12EC982F, D98BE3AA, 5448A3A0

E07378FF, 3068DDE8, C02A45A5, 489C7A1A, 5448A3A0, 4A1EB2B2, 12EC982F, D98BE3AA

489C7A1A, D5DD5018, 3068DDE8, C02A45A5, D98BE3AA, D677AAA8, 4A1EB2B2, 12EC982F

C02A45A5, B9D75D76, D5DD5018, 3068DDE8, 12EC982F, 5AA89133, D677AAA8, 4A1EB2B2

3068DDE8, 51A9B2DD, B9D75D76, D5DD5018, 4A1EB2B2, 49BCE169, 5AA89133, D677AAA8

D5DD5018, 36F589C4, 51A9B2DD, B9D75D76, D677AAA8, CF4FA8D2, 49BCE169, 5AA89133

B9D75D76, B5C60EAF, 36F589C4, 51A9B2DD, 5AA89133, C1985969, CF4FA8D2, 49BCE169

51A9B2DD, 725DF80C, B5C60EAF, 36F589C4, 49BCE169, 427440B4, C1985969, CF4FA8D2

36F589C4, 3F7A2507, 725DF80C, B5C60EAF, CF4FA8D2, 60927896, 427440B4, C1985969

B5C60EAF, 9D539EB6, 3F7A2507, 725DF80C, C1985969, 7050ED96, 60927896, 427440B4

725DF80C, 5A249895, 9D539EB6, 3F7A2507, 427440B4, CBC74513, 7050ED96, 60927896

3F7A2507, A7CECDCD, 5A249895, 9D539EB6, 60927896, 8431C75E, CBC74513, 7050ED96

9D539EB6, F8DCD12B, A7CECDCD, 5A249895, 7050ED96, 0E3A1C68, 8431C75E, CBC74513

5A249895, 3E30DB2A, F8DCD12B, A7CECDCD, CBC74513, 62EEEC87, 0E3A1C68, 8431C75E

A7CECDCD, A25D36CE, 3E30DB2A, F8DCD12B, 8431C75E, 2B1F312D, 62EEEC87, 0E3A1C68

F8DCD12B, A92CF759, A25D36CE, 3E30DB2A, 0E3A1C68, FB124197, 2B1F312D, 62EEEC87

3E30DB2A, 0CD0BA66, A92CF759, A25D36CE, 62EEEC87, DB8A5C11, FB124197, 2B1F312D

A25D36CE, AF62D775, 0CD0BA66, A92CF759, 2B1F312D, EC3264DC, DB8A5C11, FB124197

A92CF759, 69D4E1DF, AF62D775, 0CD0BA66, FB124197, 9AA87F7C, EC3264DC, DB8A5C11

0CD0BA66, 0EE66339, 69D4E1DF, AF62D775, DB8A5C11, 04512915, 9AA87F7C, EC3264DC

AF62D775, 5C5B5FBD, 0EE66339, 69D4E1DF, EC3264DC, C763272A, 04512915, 9AA87F7C

69D4E1DF, 0D80E8CF, 5C5B5FBD, 0EE66339, 9AA87F7C, CCD7DF45, C763272A, 04512915

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

A1 AA 06 89 D0 FA FA 2D DC 22 E8 8B 49 13 3A 06

الف-۲-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای $۱۰^۶$ بار تکرار می‌شود.

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
4A 7F 57 23 F9 54 EB A1 21 6C 9D 8F 63 20 43 1F
```

الف-۲-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سرخط بعد از اولین n)

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
d4 ec c9 13 e1 df 77 6b f4 8d e9 d5 5b 1f 25 46
```

الف-۲-۱۱ مثال۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۱۲۸ بیتی ذیل است:

```
13 fc 13 e8 ef ff 34 7d e1 93 ff 46 db ac cf d4
```

الف-۳ تابع درهم‌ساز اختصاصی ۳

الف-۳-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
DA 39 A3 EE 5E 6B 4B 0D 32 55 BF EF 95 60 18 90 AF D8 07 09
```

الف-۳-۲ مثال ۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ’a‘ .

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
86 F7 E4 37 FA A5 A7 FC E1 5D 1D DC B9 EA EA EA 37 76 67 B8
```

الف-۳-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای X_0, X_1, X_2, X_3, X_4 آمده است.

0116FC33, 67452301, 7BF36AE2, 98BADCFE, 10325476

8990536D, 0116FC33, 59D148C0, 7BF36AE2, 98BADCFE

A1390F08, 8990536D, C045BF0C, 59D148C0, 7BF36AE2

CDD8E11B, A1390F08, 626414DB, C045BF0C, 59D148C0

CFD499DE, CDD8E11B, 284E43C2, 626414DB, C045BF0C

3FC7CA40, CFD499DE, F3763846, 284E43C2, 626414DB

993E30C1, 3FC7CA40, B3F52677, F3763846, 284E43C2

9E8C07D4, 993E30C1, 0FF1F290, B3F52677, F3763846

4B6AE328, 9E8C07D4, 664F8C30, 0FF1F290, B3F52677

8351F929, 4B6AE328, 27A301F5, 664F8C30, 0FF1F290

FBDA9E89, 8351F929, 12DAB8CA, 27A301F5, 664F8C30

63188FE4, FBDA9E89, 60D47E4A, 12DAB8CA, 27A301F5

4607B664, 63188FE4, 7EF6A7A2, 60D47E4A, 12DAB8CA

9128F695, 4607B664, 18C623F9, 7EF6A7A2, 60D47E4A

196BEE77, 9128F695, 1181ED99, 18C623F9, 7EF6A7A2

20BDD62F, 196BEE77, 644A3DA5, 1181ED99, 18C623F9

4E925823, 20BDD62F, C65AFB9D, 644A3DA5, 1181ED99

82AA6728, 4E925823, C82F758B, C65AFB9D, 644A3DA5

DC64901D, 82AA6728, D3A49608, C82F758B, C65AFB9D

FD9E1D7D, DC64901D, 20AA99CA, D3A49608, C82F758B

1A37B0CA, FD9E1D7D, 77192407, 20AA99CA, D3A49608

33A23BFC, 1A37B0CA, 7F67875F, 77192407, 20AA99CA

21283486, 33A23BFC, 868DEC32, 7F67875F, 77192407

D541F12D, 21283486, 0CE88EFF, 868DEC32, 7F67875F

C7567DC6, D541F12D, 884A0D21, 0CE88EFF, 868DEC32

48413BA4, C7567DC6, 75507C4B, 884A0D21, 0CE88EFF

BE35FBD5, 48413BA4, B1D59F71, 75507C4B, 884A0D21

4AA84D97, BE35FBD5, 12104EE9, B1D59F71, 75507C4B

8370B52E, 4AA84D97, 6F8D7EF5, 12104EE9, B1D59F71

C5FBAF5D, 8370B52E, D2AA1365, 6F8D7EF5, 12104EE9

1267B407, C5FBAF5D, A0DC2D4B, D2AA1365, 6F8D7EF5

3B845D33, 1267B407, 717EEBD7, A0DC2D4B, D2AA1365

046FAA0A, 3B845D33, C499ED01, 717EEBD7, A0DC2D4B

2C0EBC11, 046FAA0A, CEE1174C, C499ED01, 717EEBD7

21796AD4, 2C0EBC11, 811BEA82, CEE1174C, C499ED01

DCBBB0CB, 21796AD4, 4B03AF04, 811BEA82, CEE1174C

0F511FD8, DCBBB0CB, 085E5AB5, 4B03AF04, 811BEA82

DC63973F, 0F511FD8, F72EEC32, 085E5AB5, 4B03AF04

4C986405, DC63973F, 03D447F6, F72EEC32, 085E5AB5

32DE1CBA, 4C986405, F718E5CF, 03D447F6, F72EEC32

FC87DEDF, 32DE1CBA, 53261901, F718E5CF, 03D447F6

970A0D5C, FC87DEDF, 8CB7872E, 53261901, F718E5CF

7F193DC5, 970A0D5C, FF21F7B7, 8CB7872E, 53261901

EE1B1AAF, 7F193DC5, 25C28357, FF21F7B7, 8CB7872E

40F28E09, EE1B1AAF, 5FC64F71, 25C28357, FF21F7B7

1C51E1F2, 40F28E09, FB86C6AB, 5FC64F71, 25C28357

A01B846C, 1C51E1F2, 503CA382, FB86C6AB, 5FC64F71

BEAD02CA, A01B846C, 8714787C, 503CA382, FB86C6AB

BAF39337, BEAD02CA, 2806E11B, 8714787C, 503CA382

120731C5, BAF39337, AFAB40B2, 2806E11B, 8714787C

641DB2CE, 120731C5, EEBCE4CD, AFAB40B2, 2806E11B

3847AD66, 641DB2CE, 4481CC71, EEBCE4CD, AFAB40B2

E490436D, 3847AD66, 99076CB3, 4481CC71, EEBCE4CD

27E9F1D8, E490436D, 8E11EB59, 99076CB3, 4481CC71

7B71F76D, 27E9F1D8, 792410DB, 8E11EB59, 99076CB3

5E6456AF, 7B71F76D, 09FA7C76, 792410DB, 8E11EB59

C846093F, 5E6456AF, 5EDC7DDB, 09FA7C76, 792410DB

D262FF50, C846093F, D79915AB, 5EDC7DDB, 09FA7C76

09D785FD, D262FF50, F211824F, D79915AB, 5EDC7DDB

3F52DE5A, 09D785FD, 3498BFD4, F211824F, D79915AB

D756C147, 3F52DE5A, 4275E17F, 3498BFD4, F211824F

548C9CB2, D756C147, 8FD4B796, 4275E17F, 3498BFD4

B66C020B, 548C9CB2, F5D5B051, 8FD4B796, 4275E17F

6B61C9E1, B66C020B, 9523272C, F5D5B051, 8FD4B796

19DFA7AC, 6B61C9E1, ED9B0082, 9523272C, F5D5B051

101655F9, 19DFA7AC, 5AD87278, ED9B0082, 9523272C

0C3DF2B4, 101655F9, 0677E9EB, 5AD87278, ED9B0082

78DD4D2B, 0C3DF2B4, 4405957E, 0677E9EB, 5AD87278

497093C0, 78DD4D2B, 030F7CAD, 4405957E, 0677E9EB

3F2588C2, 497093C0, DE37534A, 030F7CAD, 4405957E

C199F8C7, 3F2588C2, 125C24F0, DE37534A, 030F7CAD

39859DE7, C199F8C7, 8FC96230, 125C24F0, DE37534A

EDB42DE4, 39859DE7, F0667E31, 8FC96230, 125C24F0

11793F6F, EDB42DE4, CE616779, F0667E31, 8FC96230

5EE76897, 11793F6F, 3B6D0B79, CE616779, F0667E31

63F7DAB7, 5EE76897, C45E4FDB, 3B6D0B79, CE616779

A079B7D9, 63F7DAB7, D7B9DA25, C45E4FDB, 3B6D0B79

860D21CC, A079B7D9, D8FDF6AD, D7B9DA25, C45E4FDB

5738D5E1, 860D21CC, 681E6DF6, D8FDF6AD, D7B9DA25

42541B35, 5738D5E1, 21834873, 681E6DF6, D8FDF6AD

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

A9 99 3E 36 47 06 81 6A BA 3E 25 71 78 50 C2 6C 9C D0 D8 9D

الف-۳-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII

´message digest´

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
C1 22 52 CE DA 8B E8 99 4D 5F A0 29 0A 47 23 1C 1D 16 AA E3
```

الف-۳-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII

´abcdefghijklmnopqrstuvwxyz´

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
32 D1 0C 7B 8C F9 65 70 CA 04 CE 37 F2 A1 9D 84 24 0D 3A 89
```

الف-۳-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
76 1C 45 7B F7 3B 14 D2 7E 9E 92 65 C4 6F 4B 4D DA 11 F9 40
```

الف-۳-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار

‘1234567890’

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
50 AB F5 70 6A 15 09 90 A0 8B 2C 5E A4 0F A0 E5 85 55 47 32
```

الف-۳-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
61626364 62636465 63646566 64656667 65666768 66676869 6768696A 68696A6B
696A6B6C 6A6B6C6D 6B6C6D6E 6C6D6E6F 6D6E6F70 6E6F7071 80000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

00000000 00000000 00000000 00000000 00000000 00000000 00000000 000001C0

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای X_0, X_1, X_2, X_3, X_4 که در طول پردازش اولین بلوک به‌دست آمده، آورده شده است.

0116FC17, 67452301, 7BF36AE2, 98BADCFE, 10325476

EBF3B452, 0116FC17, 59D148C0, 7BF36AE2, 98BADCFE

5109913A, EBF3B452, C045BF05, 59D148C0, 7BF36AE2

2C4F6EAC, 5109913A, BAFCED14, C045BF05, 59D148C0

33F4AE5B, 2C4F6EAC, 9442644E, BAFCED14, C045BF05

96B85189, 33F4AE5B, 0B13DBAB, 9442644E, BAFCED14

DB04CB58, 96B85189, CCFD2B96, 0B13DBAB, 9442644E

45833F0F, DB04CB58, 65AE1462, CCFD2B96, 0B13DBAB

C565C35E, 45833F0F, 36C132D6, 65AE1462, CCFD2B96

6350AFDA, C565C35E, D160CFC3, 36C132D6, 65AE1462

8993EA77, 6350AFDA, B15970D7, D160CFC3, 36C132D6

E19ECAA2, 8993EA77, 98D42BF6, B15970D7, D160CFC3

8603481E, E19ECAA2, E264FA9D, 98D42BF6, B15970D7

32F94A85, 8603481E, B867B2A8, E264FA9D, 98D42BF6

B2E7A8BE, 32F94A85, A180D207, B867B2A8, E264FA9D

42637E39, B2E7A8BE, 4CBE52A1, A180D207, B867B2A8

6B068048, 42637E39, ACB9EA2F, 4CBE52A1, A180D207

426B9C35, 6B068048, 5098DF8E, ACB9EA2F, 4CBE52A1

944B1BD1, 426B9C35, 1AC1A012, 5098DF8E, ACB9EA2F

6C445652, 944B1BD1, 509AE70D, 1AC1A012, 5098DF8E

95836DA5, 6C445652, 6512C6F4, 509AE70D, 1AC1A012

09511177, 95836DA5, 9B111594, 6512C6F4, 509AE70D

E2B92DC4, 09511177, 6560DB69, 9B111594, 6512C6F4

FD224575, E2B92DC4, C254445D, 6560DB69, 9B111594

EEB82D9A, FD224575, 38AE4B71, C254445D, 6560DB69

5A142C1A, EEB82D9A, 7F48915D, 38AE4B71, C254445D

2972F7C7, 5A142C1A, BBAE0B66, 7F48915D, 38AE4B71

D526A644, 2972F7C7, 96850B06, BBAE0B66, 7F48915D

E1122421, D526A644, CA5CBDF1, 96850B06, BBAE0B66

05B457B2, E1122421, 3549A991, CA5CBDF1, 96850B06

A9C84BEC, 05B457B2, 78448908, 3549A991, CA5CBDF1

52E31F60, A9C84BEC, 816D15EC, 78448908, 3549A991

5AF3242C, 52E31F60, 2A7212FB, 816D15EC, 78448908

31C756A9, 5AF3242C, 14B8C7D8, 2A7212FB, 816D15EC

E9AC987C, 31C756A9, 16BCC90B, 14B8C7D8, 2A7212FB

AB7C32EE, E9AC987C, 4C71D5AA, 16BCC90B, 14B8C7D8

5933FC99, AB7C32EE, 3A6B261F, 4C71D5AA, 16BCC90B

43F87AE9, 5933FC99, AADF0CBB, 3A6B261F, 4C71D5AA

24957F22, 43F87AE9, 564CFF26, AADF0CBB, 3A6B261F

ADEB7478, 24957F22, 50FE1EBA, 564CFF26, AADF0CBB

D70E5010, ADEB7478, 89255FC8, 50FE1EBA, 564CFF26

79BCFB08, D70E5010, 2B7ADD1E, 89255FC8, 50FE1EBA

F9BCB8DE, 79BCFB08, 35C39404, 2B7ADD1E, 89255FC8

633E9561, F9BCB8DE, 1E6F3EC2, 35C39404, 2B7ADD1E

98C1EA64, 633E9561, BE6F2E37, 1E6F3EC2, 35C39404

C6EA241E, 98C1EA64, 58CFA558, BE6F2E37, 1E6F3EC2

A2AD4F02, C6EA241E, 26307A99, 58CFA558, BE6F2E37

C8A69090, A2AD4F02, B1BA8907, 26307A99, 58CFA558

88341600, C8A69090, A8AB53C0, B1BA8907, 26307A99

7E846F58, 88341600, 3229A424, A8AB53C0, B1BA8907

86E358BA, 7E846F58, 220D0580, 3229A424, A8AB53C0

8D2E76C8, 86E358BA, 1FA11BD6, 220D0580, 3229A424

CE892E10, 8D2E76C8, A1B8D62E, 1FA11BD6, 220D0580

EDEA95B1, CE892E10, 234B9DB2, A1B8D62E, 1FA11BD6

36D1230A, EDEA95B1, 33A24B84, 234B9DB2, A1B8D62E

776C3910, 36D1230A, 7B7AA56C, 33A24B84, 234B9DB2

A681B723, 776C3910, 8DB448C2, 7B7AA56C, 33A24B84

AC0A794F, A681B723, 1DDB0E44, 8DB448C2, 7B7AA56C

F03D3782, AC0A794F, E9A06DC8, 1DDB0E44, 8DB448C2

9EF775C3, F03D3782, EB029E53, E9A06DC8, 1DDB0E44

36254B13, 9EF775C3, BC0F4DE0, EB029E53, E9A06DC8

4080D4DC, 36254B13, E7BDDD70, BC0F4DE0, EB029E53

2BFAF7A8, 4080D4DC, CD8952C4, E7BDDD70, BC0F4DE0

513F9CA0, 2BFAF7A8, 10203537, CD8952C4, E7BDDD70

E5895C81, 513F9CA0, 0AFEBDEA, 10203537, CD8952C4

1037D2D5, E5895C81, 144FE728, 0AFEBDEA, 10203537

14A82DA9, 1037D2D5, 79625720, 144FE728, 0AFEBDEA

6D17C9FD, 14A82DA9, 440DF4B5, 79625720, 144FE728

2C7B07BD, 6D17C9FD, 452A0B6A, 440DF4B5, 79625720

FDF6EFFF, 2C7B07BD, 5B45F27F, 452A0B6A, 440DF4B5

112B96E3, FDF6EFFF, 4B1EC1EF, 5B45F27F, 452A0B6A

84065712, 112B96E3, FF7DBBFF, 4B1EC1EF, 5B45F27F

AB89FB71, 84065712, C44AE5B8, FF7DBBFF, 4B1EC1EF

C5210E35, AB89FB71, A10195C4, C44AE5B8, FF7DBBFF

352D9F4B, C5210E35, 6AE27EDC, A10195C4, C44AE5B8

1A0E0E0A, 352D9F4B, 7148438D, 6AE27EDC, A10195C4

D0D47349, 1A0E0E0A, CD4B67D2, 7148438D, 6AE27EDC

AD38620D, D0D47349, 86838382, CD4B67D2, 7148438D

D3AD7C25, AD38620D, 74351CD2, 86838382, CD4B67D2

8CE34517, D3AD7C25, 6B4E1883, 74351CD2, 86838382

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای X_0, X_1, X_2, X_3, X_4 که در طول پردازش دومین بلوک به‌دست آمده، آورده شده است.

2DF257E9, F4286818, B0DEC9EB, 0408F581, 84677148

4D3DC58F, 2DF257E9, 3D0A1A06, B0DEC9EB, 0408F581

C352BB05, 4D3DC58F, 4B7C95FA, 3D0A1A06, B0DEC9EB

EEF743C6, C352BB05, D34F7163, 4B7C95FA, 3D0A1A06

41E34277, EEF743C6, 70D4AEC1, D34F7163, 4B7C95FA

5443915C, 41E34277, BBBDD0F1, 70D4AEC1, D34F7163

E7FA0377, 5443915C, D078D09D, BBBDD0F1, 70D4AEC1

C6946813, E7FA0377, 1510E457, D078D09D, BBBDD0F1

FDDE1DE1, C6946813, F9FE80DD, 1510E457, D078D09D

B8538ACA, FDDE1DE1, F1A51A04, F9FE80DD, 1510E457

6BA94F63, B8538ACA, 7F778778, F1A51A04, F9FE80DD

43A2792F, 6BA94F63, AE14E2B2, 7F778778, F1A51A04

FECD7BBF, 43A2792F, DAEA53D8, AE14E2B2, 7F778778

A2604CA8, FECD7BBF, D0E89E4B, DAEA53D8, AE14E2B2

258B0BAA, A2604CA8, FFB35EEF, D0E89E4B, DAEA53D8

D9772360, 258B0BAA, 2898132A, FFB35EEF, D0E89E4B

5507DB6E, D9772360, 8962C2EA, 2898132A, FFB35EEF

A51B58BC, 5507DB6E, 365DC8D8, 8962C2EA, 2898132A

C2EB709F, A51B58BC, 9541F6DB, 365DC8D8, 8962C2EA

D8992153, C2EB709F, 2946D62F, 9541F6DB, 365DC8D8

37482F5F, D8992153, F0BADC27, 2946D62F, 9541F6DB

EE8700BD, 37482F5F, F6264854, F0BADC27, 2946D62F

9AD594B9, EE8700BD, CDD20BD7, F6264854, F0BADC27

8FBAA5B9, 9AD594B9, 7BA1C02F, CDD20BD7, F6264854

88FB5867, 8FBAA5B9, 66B5652E, 7BA1C02F, CDD20BD7

EEC50521, 88FB5867, 63EEA96E, 66B5652E, 7BA1C02F

50BCE434, EEC50521, E23ED619, 63EEA96E, 66B5652E

5C416DAF, 50BCE434, 7BB14148, E23ED619, 63EEA96E

2429BE5F, 5C416DAF, 142F390D, 7BB14148, E23ED619

0A2FB108, 2429BE5F, D7105B6B, 142F390D, 7BB14148

17986223, 0A2FB108, C90A6F97, D7105B6B, 142F390D

8A4AF384, 17986223, 028BEC42, C90A6F97, D7105B6B

6B629993, 8A4AF384, C5E61888, 028BEC42, C90A6F97

F15F04F3, 6B629993, 2292BCE1, C5E61888, 028BEC42

295CC25B, F15F04F3, DAD8A664, 2292BCE1, C5E61888

696DA404, 295CC25B, FC57C13C, DAD8A664, 2292BCE1

CEF5AE12, 696DA404, CA573096, FC57C13C, DAD8A664

87D5B80C, CEF5AE12, 1A5B6901, CA573096, FC57C13C

84E2A5F2, 87D5B80C, B3BD6B84, 1A5B6901, CA573096

03BB6310, 84E2A5F2, 21F56E03, B3BD6B84, 1A5B6901

C2D8F75F, 03BB6310, A138A97C, 21F56E03, B3BD6B84

BFB25768, C2D8F75F, 00EED8C4, A138A97C, 21F56E03

28589152, BFB25768, F0B63DD7, 00EED8C4, A138A97C

EC1D3D61, 28589152, 2FEC95DA, F0B63DD7, 00EED8C4

3CAED7AF, EC1D3D61, 8A162454, 2FEC95DA, F0B63DD7

C3D033EA, 3CAED7AF, 7B074F58, 8A162454, 2FEC95DA

7316056A, C3D033EA, CF2BB5EB, 7B074F58, 8A162454

46F93B68, 7316056A, B0F40CFA, CF2BB5EB, 7B074F58

DC8E7F26, 46F93B68, 9CC5815A, B0F40CFA, CF2BB5EB

850D411C, DC8E7F26, 11BE4EDA, 9CC5815A, B0F40CFA

7E4672C0, 850D411C, B7239FC9, 11BE4EDA, 9CC5815A

89FBD41D, 7E4672C0, 21435047, B7239FC9, 11BE4EDA

1797E228, 89FBD41D, 1F919CB0, 21435047, B7239FC9

431D65BC, 1797E228, 627EF507, 1F919CB0, 21435047

2BDBB8CB, 431D65BC, 05E5F88A, 627EF507, 1F919CB0

6DA72E7F, 2BDBB8CB, 10C7596F, 05E5F88A, 627EF507

A8495A9B, 6DA72E7F, CAF6EE32, 10C7596F, 05E5F88A

E785655A, A8495A9B, DB69CB9F, CAF6EE32, 10C7596F

5B086C42, E785655A, EA1256A6, DB69CB9F, CAF6EE32

A65818F7, 5B086C42, B9E15956, EA1256A6, DB69CB9F

7AAB101B, A65818F7, 96C21B10, B9E15956, EA1256A6

93614C9C, 7AAB101B, E996063D, 96C21B10, B9E15956

F66D9BF4, 93614C9C, DEAAC406, E996063D, 96C21B10

D504902B, F66D9BF4, 24D85327, DEAAC406, E996063D

```
60A9DA62, D504902B, 3D9B66FD, 24D85327, DEAAC406
8B687819, 60A9DA62, F541240A, 3D9B66FD, 24D85327
083E90C3, 8B687819, 982A7698, F541240A, 3D9B66FD
F6226BBF, 083E90C3, 62DA1E06, 982A7698, F541240A
76C0563B, F6226BBF, C20FA430, 62DA1E06, 982A7698
989DD165, 76C0563B, FD889AEF, C20FA430, 62DA1E06
8B2C7573, 989DD165, DDB0158E, FD889AEF, C20FA430
AE1B8E7B, 8B2C7573, 66277459, DDB0158E, FD889AEF
CA1840DE, AE1B8E7B, E2CB1D5C, 66277459, DDB0158E
16F3BABB, CA1840DE, EB86E39E, E2CB1D5C, 66277459
D28D83AD, 16F3BABB, B2861037, EB86E39E, E2CB1D5C
6BC02DFE, D28D83AD, C5BCEEAE, B2861037, EB86E39E
D3A6E275, 6BC02DFE, 74A360EB, C5BCEEAE, B2861037
DA955482, D3A6E275, 9AF00B7F, 74A360EB, C5BCEEAE
58C0AAC0, DA955482, 74E9B89D, 9AF00B7F, 74A360EB
906FD62C, 58C0AAC0, B6A55520, 74E9B89D, 9AF00B7F
```

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
84 98 3E 44 1C 3B D2 6E BA AE 4A A1 F9 51 29 E5 E5 46 70 F1
```

الف-۳-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای 10^6 بار تکرار می‌شود.

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
34 AA 97 3C D4 C4 DA A4 F6 1E EB 2B DB AD 27 31 65 34 01 6F
```

الف-۳-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سرخط بعد از اولین n)

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
a4 9b 24 46 a0 2c 64 5b f4 19 f9 95 b6 70 91 25 3a 04 a2 59
```

الف-۳-۱۱ مثال ۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۱۶۰ بیتی ذیل است:

```
37 bc 52 21 ad e3 bc 09 ca d1 5e 47 84 f3 c7 05 14 54 b1 b3
```

الف-۴ تابع درهم‌ساز اختصاصی ۴

الف-۴-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
e3b0c442 98fc1c14 9afbf4c8 996fb924 27ae41e4 649b934c a495991b 7852b855
```

الف-۴-۲ مثال ۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ’a‘ .

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
ca978112 ca1bbdca fac231b3 9a23dc4d a786eff8 147c4e72 b9807785 afee48bb
```

الف-۴-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ آمده است.

```
 init: 6a09e667 bb67ae85 3c6ef372 a54ff53a 510e527f 9b05688c 1f83d9ab 5be0cd19
   0 5d6aebcd 6a09e667 bb67ae85 3c6ef372 fa2a4622 510e527f 9b05688c 1f83d9ab
   1 5a6ad9ad 5d6aebcd 6a09e667 bb67ae85 78ce7989 fa2a4622 510e527f 9b05688c
   2 c8c347a7 5a6ad9ad 5d6aebcd 6a09e667 f92939eb 78ce7989 fa2a4622 510e527f
   3 d550f666 c8c347a7 5a6ad9ad 5d6aebcd 24e00850 f92939eb 78ce7989 fa2a4622
   4 04409a6a d550f666 c8c347a7 5a6ad9ad 43ada245 24e00850 f92939eb 78ce7989
```

5 2b4209f5 04409a6a d550f666 c8c347a7 714260ad 43ada245 24e00850 f92939eb

6 e5030380 2b4209f5 04409a6a d550f666 9b27a401 714260ad 43ada245 24e00850

7 85a07b5f e5030380 2b4209f5 04409a6a 0c657a79 9b27a401 714260ad 43ada245

8 8e04ecb9 85a07b5f e5030380 2b4209f5 32ca2d8c 0c657a79 9b27a401 714260ad

9 8c87346b 8e04ecb9 85a07b5f e5030380 1cc92596 32ca2d8c 0c657a79 9b27a401

10 4798a3f4 8c87346b 8e04ecb9 85a07b5f 436b23e8 1cc92596 32ca2d8c 0c657a79

11 f71fc5a9 4798a3f4 8c87346b 8e04ecb9 816fd6e9 436b23e8 1cc92596 32ca2d8c

12 87912990 f71fc5a9 4798a3f4 8c87346b 1e578218 816fd6e9 436b23e8 1cc92596

13 d932eb16 87912990 f71fc5a9 4798a3f4 745a48de 1e578218 816fd6e9 436b23e8

14 c0645fde d932eb16 87912990 f71fc5a9 0b92f20c 745a48de 1e578218 816fd6e9

15 b0fa238e c0645fde d932eb16 87912990 07590dcd 0b92f20c 745a48de 1e578218

16 21da9a9b b0fa238e c0645fde d932eb16 8034229c 07590dcd 0b92f20c 745a48de

17 c2fbd9d1 21da9a9b b0fa238e c0645fde 846ee454 8034229c 07590dcd 0b92f20c

18 fe777bbf c2fbd9d1 21da9a9b b0fa238e cc899961 846ee454 8034229c 07590dcd

19 e1f20c33 fe777bbf c2fbd9d1 21da9a9b b0638179 cc899961 846ee454 8034229c

20 9dc68b63 e1f20c33 fe777bbf c2fbd9d1 8ada8930 b0638179 cc899961 846ee454

21 c2606d6d 9dc68b63 e1f20c33 fe777bbf e1257970 8ada8930 b0638179 cc899961

22 a7a3623f c2606d6d 9dc68b63 e1f20c33 49f5114a e1257970 8ada8930 b0638179

23 c5d53d8d a7a3623f c2606d6d 9dc68b63 aa47c347 49f5114a e1257970 8ada8930

24 1c2c2838 c5d53d8d a7a3623f c2606d6d 2823ef91 aa47c347 49f5114a e1257970

25 cde8037d 1c2c2838 c5d53d8d a7a3623f 14383d8e 2823ef91 aa47c347 49f5114a

26 b62ec4bc cde8037d 1c2c2838 c5d53d8d c74c6516 14383d8e 2823ef91 aa47c347

27 77d37528 b62ec4bc cde8037d 1c2c2838 edffbff8 c74c6516 14383d8e 2823ef91

28 363482c9 77d37528 b62ec4bc cde8037d 6112a3b7 edffbff8 c74c6516 14383d8e

29 a0060b30 363482c9 77d37528 b62ec4bc ade79437 6112a3b7 edffbff8 c74c6516

30 ea992a22 a0060b30 363482c9 77d37528 0109ab3a ade79437 6112a3b7 edffbff8

31 73b33bf5 ea992a22 a0060b30 363482c9 ba591112 0109ab3a ade79437 6112a3b7

32 98e12507 73b33bf5 ea992a22 a0060b30 9cd9f5f6 ba591112 0109ab3a ade79437

33 fe604df5 98e12507 73b33bf5 ea992a22 59249dd3 9cd9f5f6 ba591112 0109ab3a

34 a9a7738c fe604df5 98e12507 73b33bf5 085f3833 59249dd3 9cd9f5f6 ba591112

35 65a0cfe4 a9a7738c fe604df5 98e12507 f4b002d6 085f3833 59249dd3 9cd9f5f6

36 41a65cb1 65a0cfe4 a9a7738c fe604df5 0772a26b f4b002d6 085f3833 59249dd3

37 34df1604 41a65cb1 65a0cfe4 a9a7738c a507a53d 0772a26b f4b002d6 085f3833

38 6dc57a8a 34df1604 41a65cb1 65a0cfe4 f0781bc8 a507a53d 0772a26b f4b002d6

39 79ea687a 6dc57a8a 34df1604 41a65cb1 1efbc0a0 f0781bc8 a507a53d 0772a26b

40 d6670766 79ea687a 6dc57a8a 34df1604 26352d63 1efbc0a0 f0781bc8 a507a53d

41 df46652f d6670766 79ea687a 6dc57a8a 838b2711 26352d63 1efbc0a0 f0781bc8

42 17aa0dfe df46652f d6670766 79ea687a decd4715 838b2711 26352d63 1efbc0a0

43 9d4baf93 17aa0dfe df46652f d6670766 fda24c2e decd4715 838b2711 26352d63

44 26628815 9d4baf93 17aa0dfe df46652f a80f11f0 fda24c2e decd4715 838b2711

45 72ab4b91 26628815 9d4baf93 17aa0dfe b7755da1 a80f11f0 fda24c2e decd4715

46 a14c14b0 72ab4b91 26628815 9d4baf93 d57b94a9 b7755da1 a80f11f0 fda24c2e

47 4172328d a14c14b0 72ab4b91 26628815 fecf0bc6 d57b94a9 b7755da1 a80f11f0

48 05757ceb 4172328d a14c14b0 72ab4b91 bd714038 fecf0bc6 d57b94a9 b7755da1

49 f11bfaa8 05757ceb 4172328d a14c14b0 6e5c390c bd714038 fecf0bc6 d57b94a9

50 7a0508a1 f11bfaa8 05757ceb 4172328d 52f1ccf7 6e5c390c bd714038 fecf0bc6

51 886e7a22 7a0508a1 f11bfaa8 05757ceb 49231c1e 52f1ccf7 6e5c390c bd714038

52 101fd28f 886e7a22 7a0508a1 f11bfaa8 529e7d00 49231c1e 52f1ccf7 6e5c390c

53 f5702fdb 101fd28f 886e7a22 7a0508a1 9f4787c3 529e7d00 49231c1e 52f1ccf7

54 3ec45cdb f5702fdb 101fd28f 886e7a22 e50e1b4f 9f4787c3 529e7d00 49231c1e

55 38cc9913 3ec45cdb f5702fdb 101fd28f 54cb266b e50e1b4f 9f4787c3 529e7d00

56 fcd1887b 38cc9913 3ec45cdb f5702fdb 9b5e906c 54cb266b e50e1b4f 9f4787c3

57 c062d46f fcd1887b 38cc9913 3ec45cdb 7e44008e 9b5e906c 54cb266b e50e1b4f

58 ffb70472 c062d46f fcd1887b 38cc9913 6d83bfc6 7e44008e 9b5e906c 54cb266b

59 b6ae8fff ffb70472 c062d46f fcd1887b b21bad3d 6d83bfc6 7e44008e 9b5e906c

60 b85e2ce9 b6ae8fff ffb70472 c062d46f 961f4894 b21bad3d 6d83bfc6 7e44008e

61 04d24d6c b85e2ce9 b6ae8fff ffb70472 948d25b6 961f4894 b21bad3d 6d83bfc6

62 d39a2165 04d24d6c b85e2ce9 b6ae8fff fb121210 948d25b6 961f4894 b21bad3d

63 506e3058 d39a2165 04d24d6c b85e2ce9 5ef50f24 fb121210 948d25b6 961f4894

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

$$X0 = 6a09e667 \boxplus 506e3058 = ba7816bf$$

$$X1 = bb67ae85 \boxplus d39a2165 = 8f01cfea$$

$$X2 = 3c6ef372 \boxplus 04d24d6c = 414140de$$

$$X3 = a54ff53a \boxplus b85e2ce9 = 5dae2223$$

$$X4 = 510e527f \boxplus 5ef50f24 = b00361a3$$

$$X5 = 9b05688c \boxplus fb121210 = 96177a9c$$

$X6 = 1f83d9ab \boxplus 948d25b6 = b410ff61$

$X7 = 5be0cd19 \boxplus 961f4894 = f20015ad$

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
ba7816bf 8f01cfea 414140de 5dae2223 b00361a3 96177a9c b410ff61 f20015ad
```

الف-۴-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´
کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
f7846f55 cf23e14e ebeab5b4 e1550cad 5b509e33 48fbc4ef a3a1413d 393cb650
```

الف-۴-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII
´abcdefghijklmnopqrstuvwxyz´
کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

71c480df 93d6ae2f 1efad144 7c66c952 5e316218 cf51fc8d 9ed832f2 daf18b73

الف-۴-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

db4bfcbd 4da0cd85 a60c3c37 d3fbd880 5c77f15f c6b1fdfe 614ee0a7 c8fdb4c0

الف-۴-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار

’1234567890’

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
f371bc4a 311f2b00 9eef952d d83ca80e 2b60026c 8e935592 d0f9c308 453c813e
```

الف-۴-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
61626364 62636465 63646566 64656667 65666768 66676869 6768696a 68696a6b
696a6b6c 6a6b6c6d 6b6c6d6e 6c6d6e6f 6d6e6f70 6e6f7071 80000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 000001c0
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ در پردازش اولین بلوک، آورده شده است.

```
init: 6a09e667 bb67ae85 3c6ef372 a54ff53a 510e527f 9b05688c 1f83d9ab 5be0cd19
  0 5d6aebb1 6a09e667 bb67ae85 3c6ef372 fa2a4606 510e527f 9b05688c 1f83d9ab
  1 2f2d5fcf 5d6aebb1 6a09e667 bb67ae85 4eb1cfce fa2a4606 510e527f 9b05688c
  2 97651825 2f2d5fcf 5d6aebb1 6a09e667 62d5c49e 4eb1cfce fa2a4606 510e527f
  3 4a8d64d5 97651825 2f2d5fcf 5d6aebb1 6494841b 62d5c49e 4eb1cfce fa2a4606
  4 f921c212 4a8d64d5 97651825 2f2d5fcf 05c4f88a 6494841b 62d5c49e 4eb1cfce
  5 55c8ef48 f921c212 4a8d64d5 97651825 7ff91c94 05c4f88a 6494841b 62d5c49e
  6 485835b7 55c8ef48 f921c212 4a8d64d5 39a5b2ca 7ff91c94 05c4f88a 6494841b
  7 d237e6db 485835b7 55c8ef48 f921c212 a401d211 39a5b2ca 7ff91c94 05c4f88a
  8 359f2bce d237e6db 485835b7 55c8ef48 c09ffec4 a401d211 39a5b2ca 7ff91c94
  9 3a474b2b 359f2bce d237e6db 485835b7 9037b3b8 c09ffec4 a401d211 39a5b2ca
 10 b8e2b4cb 3a474b2b 359f2bce d237e6db 443ed29e 9037b3b8 c09ffec4 a401d211
 11 1762215c b8e2b4cb 3a474b2b 359f2bce ee1c97a8 443ed29e 9037b3b8 c09ffec4
 12 101a4861 1762215c b8e2b4cb 3a474b2b 839a0fc9 ee1c97a8 443ed29e 9037b3b8
 13 d68e6457 101a4861 1762215c b8e2b4cb 9243f8af 839a0fc9 ee1c97a8 443ed29e
 14 dd16cbb3 d68e6457 101a4861 1762215c 9162aded 9243f8af 839a0fc9 ee1c97a8
 15 c3486194 dd16cbb3 d68e6457 101a4861 1496a54f 9162aded 9243f8af 839a0fc9
 16 b9dcacb1 c3486194 dd16cbb3 d68e6457 d4f64250 1496a54f 9162aded 9243f8af
 17 046a193e b9dcacb1 c3486194 dd16cbb3 885370b6 d4f64250 1496a54f 9162aded
 18 f402f058 046a193e b9dcacb1 c3486194 6f433549 885370b6 d4f64250 1496a54f
 19 2139187b f402f058 046a193e b9dcacb1 7c304206 6f433549 885370b6 d4f64250
 20 d70ac17d 2139187b f402f058 046a193e 7cc6b262 7c304206 6f433549 885370b6
 21 1b2b66b8 d70ac17d 2139187b f402f058 d560b028 7cc6b262 7c304206 6f433549
 22 ae2e2d4f 1b2b66b8 d70ac17d 2139187b f074fc95 d560b028 7cc6b262 7c304206
```

23 59fce6b9 ae2e2d4f 1b2b66b8 d70ac17d a2c7d51d f074fc95 d560b028 7cc6b262

24 4a885065 59fce6b9 ae2e2d4f 1b2b66b8 763597fb a2c7d51d f074fc95 d560b028

25 573221da 4a885065 59fce6b9 ae2e2d4f 36e74eb4 763597fb a2c7d51d f074fc95

26 128661da 573221da 4a885065 59fce6b9 1162d575 36e74eb4 763597fb a2c7d51d

27 73f858af 128661da 573221da 4a885065 e77c797f 1162d575 36e74eb4 763597fb

28 74bcf468 73f858af 128661da 573221da 72abaecd e77c797f 1162d575 36e74eb4

29 df7151a0 74bcf468 73f858af 128661da 7629c961 72abaecd e77c797f 1162d575

30 eb43f3ed df7151a0 74bcf468 73f858af 0635d880 7629c961 72abaecd e77c797f

31 5581ab07 eb43f3ed df7151a0 74bcf468 df980085 0635d880 7629c961 72abaecd

32 9fc905c8 5581ab07 eb43f3ed df7151a0 a94d2af1 df980085 0635d880 7629c961

33 9ce5a62f 9fc905c8 5581ab07 eb43f3ed 6ef3b6bd a94d2af1 df980085 0635d880

34 1df8e885 9ce5a62f 9fc905c8 5581ab07 2a9e048e 6ef3b6bd a94d2af1 df980085

35 0786dce8 1df8e885 9ce5a62f 9fc905c8 de2a21d1 2a9e048e 6ef3b6bd a94d2af1

36 2c55d3a6 0786dce8 1df8e885 9ce5a62f b067c1af de2a21d1 2a9e048e 6ef3b6bd

37 a985b4be 2c55d3a6 0786dce8 1df8e885 f72bf353 b067c1af de2a21d1 2a9e048e

38 91ac9d5d a985b4be 2c55d3a6 0786dce8 68d8d590 f72bf353 b067c1af de2a21d1

39 7e4d30b8 91ac9d5d a985b4be 2c55d3a6 9f5b9b6d 68d8d590 f72bf353 b067c1af

40 7e056794 7e4d30b8 91ac9d5d a985b4be 423b26c0 9f5b9b6d 68d8d590 f72bf353

41 508a16ab 7e056794 7e4d30b8 91ac9d5d 45459d97 423b26c0 9f5b9b6d 68d8d590

42 b62c7013 508a16ab 7e056794 7e4d30b8 80a92a00 45459d97 423b26c0 9f5b9b6d

43 167361de b62c7013 508a16ab 7e056794 41dd3844 80a92a00 45459d97 423b26c0

44 de71e2f2 167361de b62c7013 508a16ab ff61c636 41dd3844 80a92a00 45459d97

45 18f0d19d de71e2f2 167361de b62c7013 6b88472c ff61c636 41dd3844 80a92a00

46 165be9cd 18f0d19d de71e2f2 167361de a483f080 6b88472c ff61c636 41dd3844

47 13d82741 165be9cd 18f0d19d de71e2f2 a7802a4d a483f080 6b88472c ff61c636

48 017b9d99 13d82741 165be9cd 18f0d19d aeb10b60 a7802a4d a483f080 6b88472c

49 543c99a1 017b9d99 13d82741 165be9cd 16f134b6 aeb10b60 a7802a4d a483f080

50 758ca97a 543c99a1 017b9d99 13d82741 100cf2ea 16f134b6 aeb10b60 a7802a4d

51 81c1cde0 758ca97a 543c99a1 017b9d99 5c47eb7b 100cf2ea 16f134b6 aeb10b60

52 b8d55619 81c1cde0 758ca97a 543c99a1 1c806a61 5c47eb7b 100cf2ea 16f134b6

53 1d6de87a b8d55619 81c1cde0 758ca97a 3443bed4 1c806a61 5c47eb7b 100cf2ea

54 f907b313 1d6de87a b8d55619 81c1cde0 61a41711 3443bed4 1c806a61 5c47eb7b

```
55 9e57c4a0 f907b313 1d6de87a b8d55619 eec13548 61a41711 3443bed4 1c806a61
56 71629856 9e57c4a0 f907b313 1d6de87a 2f6c8c4e eec13548 61a41711 3443bed4
57 7c015a2c 71629856 9e57c4a0 f907b313 cb9d3dd0 2f6c8c4e eec13548 61a41711
58 921fccb6 7c015a2c 71629856 9e57c4a0 43d8a034 cb9d3dd0 2f6c8c4e eec13548
59 e18f259a 921fccb6 7c015a2c 71629856 51e15869 43d8a034 cb9d3dd0 2f6c8c4e
60 bcfce922 e18f259a 921fccb6 7c015a2c 962d8621 51e15869 43d8a034 cb9d3dd0
61 f6f443f8 bcfce922 e18f259a 921fccb6 acc75916 962d8621 51e15869 43d8a034
62 86126910 f6f443f8 bcfce922 e18f259a 2fc08f85 acc75916 962d8621 51e15869
63 1bdc6f6f 86126910 f6f443f8 bcfce922 25d2430a 2fc08f85 acc75916 962d8621
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تابع گردساز در پردازش اولین بلوک را نمایش می‌دهند:

$$X_0 = 6a09e667 \boxplus 1bdc6f6f = 85e655d6$$

$$X_1 = bb67ae85 \boxplus 86126910 = 417a1795$$

$$X_2 = 3c6ef372 \boxplus f6f443f8 = 3363376a$$

$$X_3 = a54ff53a \boxplus bcfce922 = 624cde5c$$

$$X_4 = 510e527f \boxplus 25d2430a = 76e09589$$

$$X_5 = 9b05688c \boxplus 2fc08f85 = cac5f811$$

$$X_6 = 1f83d9ab \boxplus acc75916 = cc4b32c1$$

$$X_7 = 5be0cd19 \boxplus 962d8621 = f20e533a$$

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ در پردازش دومین بلوک، آورده شده است.

```
init: 85e655d6 417a1795 3363376a 624cde5c 76e09589 cac5f811 cc4b32c1 f20e533a
 0 7c20c838 85e655d6 417a1795 3363376a 4670ae6e 76e09589 cac5f811 cc4b32c1
 1 7c3c0f86 7c20c838 85e655d6 417a1795 8c51be64 4670ae6e 76e09589 cac5f811
 2 fd1eebdc 7c3c0f86 7c20c838 85e655d6 af71b9ea 8c51be64 4670ae6e 76e09589
 3 f268faa9 fd1eebdc 7c3c0f86 7c20c838 e20362ef af71b9ea 8c51be64 4670ae6e
 4 185a5d79 f268faa9 fd1eebdc 7c3c0f86 8dff3001 e20362ef af71b9ea 8c51be64
 5 3eeb6c06 185a5d79 f268faa9 fd1eebdc fe20cda6 8dff3001 e20362ef af71b9ea
 6 89bba3f1 3eeb6c06 185a5d79 f268faa9 0a34df03 fe20cda6 8dff3001 e20362ef
 7 bf9a93a0 89bba3f1 3eeb6c06 185a5d79 059abdd1 0a34df03 fe20cda6 8dff3001
 8 2c096744 bf9a93a0 89bba3f1 3eeb6c06 abfa465b 059abdd1 0a34df03 fe20cda6
```

9 2d964e86 2c096744 bf9a93a0 89bba3f1 aa27ed82 abfa465b 059abdd1 0a34df03

10 5b35025b 2d964e86 2c096744 bf9a93a0 10e77723 aa27ed82 abfa465b 059abdd1

11 5eb4ec40 5b35025b 2d964e86 2c096744 e11b4548 10e77723 aa27ed82 abfa465b

12 35ee996d 5eb4ec40 5b35025b 2d964e86 5c24e2a2 e11b4548 10e77723 aa27ed82

13 d74080fa 35ee996d 5eb4ec40 5b35025b 68aa893f 5c24e2a2 e11b4548 10e77723

14 0cea5cbc d74080fa 35ee996d 5eb4ec40 60356548 68aa893f 5c24e2a2 e11b4548

15 16a8cc79 0cea5cbc d74080fa 35ee996d 0fcb1f6f 60356548 68aa893f 5c24e2a2

16 f16f634e 16a8cc79 0cea5cbc d74080fa 8b21cdc1 0fcb1f6f 60356548 68aa893f

17 23dcb6c2 f16f634e 16a8cc79 0cea5cbc ca9182d3 8b21cdc1 0fcb1f6f 60356548

18 dcff40fd 23dcb6c2 f16f634e 16a8cc79 69bf7b95 ca9182d3 8b21cdc1 0fcb1f6f

19 76f1a2bc dcff40fd 23dcb6c2 f16f634e 0dc84bb1 69bf7b95 ca9182d3 8b21cdc1

20 20aad899 76f1a2bc dcff40fd 23dcb6c2 cc4769f2 0dc84bb1 69bf7b95 ca9182d3

21 d44dc81a 20aad899 76f1a2bc dcff40fd 5bace62d cc4769f2 0dc84bb1 69bf7b95

22 f13ae55b d44dc81a 20aad899 76f1a2bc 966aa287 5bace62d cc4769f2 0dc84bb1

23 a4195b91 f13ae55b d44dc81a 20aad899 eddbd6ed 966aa287 5bace62d cc4769f2

24 4984fa79 a4195b91 f13ae55b d44dc81a a530d939 eddbd6ed 966aa287 5bace62d

25 aa6cb982 4984fa79 a4195b91 f13ae55b 0b5eeea4 a530d939 eddbd6ed 966aa287

26 9450fbbc aa6cb982 4984fa79 a4195b91 09166dda 0b5eeea4 a530d939 eddbd6ed

27 0d936bab 9450fbbc aa6cb982 4984fa79 6e495d4b 09166dda 0b5eeea4 a530d939

28 d958b529 0d936bab 9450fbbc aa6cb982 c2fa99b1 6e495d4b 09166dda 0b5eeea4

29 1cfa5eb0 d958b529 0d936bab 9450fbbc 6c49db9f c2fa99b1 6e495d4b 09166dda

30 02ef3a5f 1cfa5eb0 d958b529 0d936bab 5da10665 6c49db9f c2fa99b1 6e495d4b

31 b0eab1c5 02ef3a5f 1cfa5eb0 d958b529 f6d93952 5da10665 6c49db9f c2fa99b1

32 0bfba73c b0eab1c5 02ef3a5f 1cfa5eb0 8b99e3a9 f6d93952 5da10665 6c49db9f

33 4bd1df96 0bfba73c b0eab1c5 02ef3a5f 905e44ac 8b99e3a9 f6d93952 5da10665

34 9907f1b6 4bd1df96 0bfba73c b0eab1c5 66c3043d 905e44ac 8b99e3a9 f6d93952

35 ecde4e0d 9907f1b6 4bd1df96 0bfba73c 5dc119e6 66c3043d 905e44ac 8b99e3a9

36 2f11c939 ecde4e0d 9907f1b6 4bd1df96 fed4ce1d 5dc119e6 66c3043d 905e44ac

37 d949682b 2f11c939 ecde4e0d 9907f1b6 32d99008 fed4ce1d 5dc119e6 66c3043d

38 adca7a96 d949682b 2f11c939 ecde4e0d c6cce4ff 32d99008 fed4ce1d 5dc119e6

39 221b8a5a adca7a96 d949682b 2f11c939 0b82c5eb c6cce4ff 32d99008 fed4ce1d

40 12d97845 221b8a5a adca7a96 d949682b e4213ca2 0b82c5eb c6cce4ff 32d99008

41 2c794876 12d97845 221b8a5a adca7a96 ff6759ba e4213ca2 0b82c5eb c6cce4ff

42 8300fca2 2c794876 12d97845 221b8a5a e0e3457c ff6759ba e4213ca2 0b82c5eb

43 f2ad6322 8300fca2 2c794876 12d97845 cc48c7f3 e0e3457c ff6759ba e4213ca2

44 0f154e11 f2ad6322 8300fca2 2c794876 6f9517cb cc48c7f3 e0e3457c ff6759ba

45 104a7db4 0f154e11 f2ad6322 8300fca2 5348e8f6 6f9517cb cc48c7f3 e0e3457c

46 0b3303a7 104a7db4 0f154e11 f2ad6322 bbe1c39a 5348e8f6 6f9517cb cc48c7f3

47 d7354d5b 0b3303a7 104a7db4 0f154e11 aad55b6b bbe1c39a 5348e8f6 6f9517cb

48 b736d7a6 d7354d5b 0b3303a7 104a7db4 68f25260 aad55b6b bbe1c39a 5348e8f6

49 2748e5ec b736d7a6 d7354d5b 0b3303a7 d4b58576 68f25260 aad55b6b bbe1c39a

50 d8aabcf9 2748e5ec b736d7a6 d7354d5b 27844711 d4b58576 68f25260 aad55b6b

51 1a6bcf6a d8aabcf9 2748e5ec b736d7a6 ff5e99d0 27844711 d4b58576 68f25260

52 4eca6fa0 1a6bcf6a d8aabcf9 2748e5ec 989ed071 ff5e99d0 27844711 d4b58576

53 ec02560a 4eca6fa0 1a6bcf6a d8aabcf9 7151df8e 989ed071 ff5e99d0 27844711

54 d9f0c115 ec02560a 4eca6fa0 1a6bcf6a 624150c4 7151df8e 989ed071 ff5e99d0

55 92952710 d9f0c115 ec02560a 4eca6fa0 226806d6 624150c4 7151df8e 989ed071

56 20d4d0e4 92952710 d9f0c115 ec02560a 4e515a4d 226806d6 624150c4 7151df8e

57 4348eb1f 20d4d0e4 92952710 d9f0c115 c21eddf9 4e515a4d 226806d6 624150c4

58 286fe5f0 4348eb1f 20d4d0e4 92952710 54076664 c21eddf9 4e515a4d 226806d6

59 1c4cddd9 286fe5f0 4348eb1f 20d4d0e4 f487a853 54076664 c21eddf9 4e515a4d

60 a9f181dd 1c4cddd9 286fe5f0 4348eb1f 27ccb387 f487a853 54076664 c21eddf9

61 b25cef29 a9f181dd 1c4cddd9 286fe5f0 2aa1bb13 27ccb387 f487a853 54076664

62 908c2123 b25cef29 a9f181dd 1c4cddd9 9a392956 2aa1bb13 27ccb387 f487a853

63 9ea7148b 908c2123 b25cef29 a9f181dd 2c5c4ed0 9a392956 2aa1bb13 27ccb387

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

$$X_0 = 85e655d6 \boxplus 9ea7148b = 248d6a61$$

$$X_1 = 417a1795 \boxplus 908c2123 = d20638b8$$

$$X_2 = 3363376a \boxplus b25cef29 = e5c02693$$

$$X_3 = 624cde5c \boxplus a9f181dd = 0c3e6039$$

$$X_4 = 76e09589 \boxplus 2c5c4ed0 = a33ce459$$

$$X_5 = cac5f811 \boxplus 9a392956 = 64ff2167$$

$$X_6 = cc4b32c1 \boxplus 2aa1bb13 = f6ecedd4$$

$$X_7 = \texttt{f20e533a} \boxplus \texttt{27ccb387} = \texttt{19db06c1}$$

مقدار درهم برای این پیام به صورت زیر است

```
248d6a61 d20638b8 e5c02693 0c3e6039 a33ce459 64ff2167 f6ecedd4 19db06c1
```

الف-۴-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای 10^6 بار تکرار می‌شود.

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
cdc76e5c 9914fb92 81a1c7e2 84d73e67 f1809a48 a497200e 046d39cc c7112cd0
```

الف-۴-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سرخط بعد از اولین n)

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
cf5b16a7 78af8380 036ce59e 7b049237 0b249b11 e8f07a51 afac4503 7afee9d1
```

الف۴-۱۱ مثال۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۲۵۶ بیتی ذیل است:

```
b09cbd26 3b043f00 0c5befca a40bc2f5 5a4785e0 24e5deb7 49b56061 eafb65e9
```

الف-۵ تابع درهم‌ساز اختصاصی ۵

الف-۵-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
cf83e1357eefb8bd f1542850d66d8007 d620e4050b5715dc 83f4a921d36ce9ce
47d0d13c5d85f2b0 ff8318d2877eec2f 63b931bd47417a81 a538327af927da3e
```

الف-۵-۲ مثال۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ’a‘ .

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
1f40fc92da241694 750979ee6cf582f2 d5d7d28e18335de0 5abc54d0560e0f53
02860c652bf08d56 0252aa5e74210546 f369fbbbce8c12cf c7957b2652fe9a75
```

الف-۵-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000018 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ آمده است.

```
Init 6a09e667f3bcc908 bb67ae8584caa73b 3c6ef372fe94f82b a54ff53a5f1d36f1
     510e527fade682d1 9b05688c2b3e6c1f 1f83d9abfb41bd6b 5be0cd19137e2179
   0 f6afceb8bcfcddf5 6a09e667f3bcc908 bb67ae8584caa73b 3c6ef372fe94f82b
     58cb02347ab51f91 510e527fade682d1 9b05688c2b3e6c1f 1f83d9abfb41bd6b
   1 1320f8c9fb872cc0 f6afceb8bcfcddf5 6a09e667f3bcc908 bb67ae8584caa73b
     c3d4ebfd48650ffa 58cb02347ab51f91 510e527fade682d1 9b05688c2b3e6c1f
   2 ebcffc07203d91f3 1320f8c9fb872cc0 f6afceb8bcfcddf5 6a09e667f3bcc908
     dfa9b239f2697812 c3d4ebfd48650ffa 58cb02347ab51f91 510e527fade682d1
   3 5a83cb3e80050e82 ebcffc07203d91f3 1320f8c9fb872cc0 f6afceb8bcfcddf5
     0b47b4bb1928990e dfa9b239f2697812 c3d4ebfd48650ffa 58cb02347ab51f91
   4 b680953951604860 5a83cb3e80050e82 ebcffc07203d91f3 1320f8c9fb872cc0
     745aca4a342ed2e2 0b47b4bb1928990e dfa9b239f2697812 c3d4ebfd48650ffa
   5 af573b02403e89cd b680953951604860 5a83cb3e80050e82 ebcffc07203d91f3
     96f60209b6dc35ba 745aca4a342ed2e2 0b47b4bb1928990e dfa9b239f2697812
```

 6 c4875b0c7abc076b af573b02403e89cd b680953951604860 5a83cb3e80050e82
   5a6c781f54dcc00c 96f60209b6dc35ba 745aca4a342ed2e2 0b47b4bb1928990e
 7 8093d195e0054fa3 c4875b0c7abc076b af573b02403e89cd b680953951604860
   86f67263a0f0ec0a 5a6c781f54dcc00c 96f60209b6dc35ba 745aca4a342ed2e2
 8 f1eca5544cb89225 8093d195e0054fa3 c4875b0c7abc076b af573b02403e89cd
   d0403c398fc40002 86f67263a0f0ec0a 5a6c781f54dcc00c 96f60209b6dc35ba
 9 81782d4a5db48f03 f1eca5544cb89225 8093d195e0054fa3 c4875b0c7abc076b
   00091f460be46c52 d0403c398fc40002 86f67263a0f0ec0a 5a6c781f54dcc00c
10 69854c4aa0f25b59 81782d4a5db48f03 f1eca5544cb89225 8093d195e0054fa3
   d375471bde1ba3f4 00091f460be46c52 d0403c398fc40002 86f67263a0f0ec0a
11 db0a9963f80c2eaa 69854c4aa0f25b59 81782d4a5db48f03 f1eca5544cb89225
   475975b91a7a462c d375471bde1ba3f4 00091f460be46c52 d0403c398fc40002
12 5e41214388186c14 db0a9963f80c2eaa 69854c4aa0f25b59 81782d4a5db48f03
   cdf3bff2883fc9d9 475975b91a7a462c d375471bde1ba3f4 00091f460be46c52
13 44249631255d2ca0 5e41214388186c14 db0a9963f80c2eaa 69854c4aa0f25b59
   860acf9effba6f61 cdf3bff2883fc9d9 475975b91a7a462c d375471bde1ba3f4
14 fa967eed85a08028 44249631255d2ca0 5e41214388186c14 db0a9963f80c2eaa
   874bfe5f6aae9f2f 860acf9effba6f61 cdf3bff2883fc9d9 475975b91a7a462c
15 0ae07c86b1181c75 fa967eed85a08028 44249631255d2ca0 5e41214388186c14
   a77b7c035dd4c161 874bfe5f6aae9f2f 860acf9effba6f61 cdf3bff2883fc9d9
16 caf81a425d800537 0ae07c86b1181c75 fa967eed85a08028 44249631255d2ca0
   2deecc6b39d64d78 a77b7c035dd4c161 874bfe5f6aae9f2f 860acf9effba6f61
17 4725be249ad19e6b caf81a425d800537 0ae07c86b1181c75 fa967eed85a08028
   f47e8353f8047455 2deecc6b39d64d78 a77b7c035dd4c161 874bfe5f6aae9f2f
18 3c4b4104168e3edb 4725be249ad19e6b caf81a425d800537 0ae07c86b1181c75
   29695fd88d81dbd0 f47e8353f8047455 2deecc6b39d64d78 a77b7c035dd4c161
19 9a3fb4d38ab6cf06 3c4b4104168e3edb 4725be249ad19e6b caf81a425d800537
   f14998dd5f70767e 29695fd88d81dbd0 f47e8353f8047455 2deecc6b39d64d78
20 8dc5ae65569d3855 9a3fb4d38ab6cf06 3c4b4104168e3edb 4725be249ad19e6b
   4bb9e66d1145bfdc f14998dd5f70767e 29695fd88d81dbd0 f47e8353f8047455

21 da34d6673d452dcf 8dc5ae65569d3855 9a3fb4d38ab6cf06 3c4b4104168e3edb
8e30ff09ad488753 4bb9e66d1145bfdc f14998dd5f70767e 29695fd88d81dbd0
22 3e2644567b709a78 da34d6673d452dcf 8dc5ae65569d3855 9a3fb4d38ab6cf06
0ac2b11da8f571c6 8e30ff09ad488753 4bb9e66d1145bfdc f14998dd5f70767e
23 4f6877b58fe55484 3e2644567b709a78 da34d6673d452dcf 8dc5ae65569d3855
c66005f87db55233 0ac2b11da8f571c6 8e30ff09ad488753 4bb9e66d1145bfdc
24 9aff71163fa3a940 4f6877b58fe55484 3e2644567b709a78 da34d6673d452dcf
d3ecf13769180e6f c66005f87db55233 0ac2b11da8f571c6 8e30ff09ad488753
25 0bc5f791f8e6816b 9aff71163fa3a940 4f6877b58fe55484 3e2644567b709a78
6ddf1fd7edcce336 d3ecf13769180e6f c66005f87db55233 0ac2b11da8f571c6
26 884c3bc27bc4f941 0bc5f791f8e6816b 9aff71163fa3a940 4f6877b58fe55484
e6e48c9a8e948365 6ddf1fd7edcce336 d3ecf13769180e6f c66005f87db55233
27 eab4a9e5771b8d09 884c3bc27bc4f941 0bc5f791f8e6816b 9aff71163fa3a940
09068a4e255a0dac e6e48c9a8e948365 6ddf1fd7edcce336 d3ecf13769180e6f
28 e62349090f47d30a eab4a9e5771b8d09 884c3bc27bc4f941 0bc5f791f8e6816b
0fcdf99710f21584 09068a4e255a0dac e6e48c9a8e948365 6ddf1fd7edcce336
29 74bf40f869094c63 e62349090f47d30a eab4a9e5771b8d09 884c3bc27bc4f941
f0aec2fe1437f085 0fcdf99710f21584 09068a4e255a0dac e6e48c9a8e948365
30 4c4fbbb75f1873a6 74bf40f869094c63 e62349090f47d30a eab4a9e5771b8d09
73e025d91b9efea3 f0aec2fe1437f085 0fcdf99710f21584 09068a4e255a0dac
31 ff4d3f1f0d46a736 4c4fbbb75f1873a6 74bf40f869094c63 e62349090f47d30a
3cd388e119e8162e 73e025d91b9efea3 f0aec2fe1437f085 0fcdf99710f21584
32 a0509015ca08c8d4 ff4d3f1f0d46a736 4c4fbbb75f1873a6 74bf40f869094c63
e1034573654a106f 3cd388e119e8162e 73e025d91b9efea3 f0aec2fe1437f085
33 60d4e6995ed91fe6 a0509015ca08c8d4 ff4d3f1f0d46a736 4c4fbbb75f1873a6
efabbd8bf47c041a e1034573654a106f 3cd388e119e8162e 73e025d91b9efea3
34 2c59ec7743632621 60d4e6995ed91fe6 a0509015ca08c8d4 ff4d3f1f0d46a736
0fbae670fa780fd3 efabbd8bf47c041a e1034573654a106f 3cd388e119e8162e
35 1a081afc59fdbc2c 2c59ec7743632621 60d4e6995ed91fe6 a0509015ca08c8d4
f098082f502b44cd 0fbae670fa780fd3 efabbd8bf47c041a e1034573654a106f

36 88df85b0bbe77514 1a081afc59fdbc2c 2c59ec7743632621 60d4e6995ed91fe6
8fbfd0162bbf4675 f098082f502b44cd 0fbae670fa780fd3 efabbd8bf47c041a
37 002bb8e4cd989567 88df85b0bbe77514 1a081afc59fdbc2c 2c59ec7743632621
66adcfa249ac7bbd 8fbfd0162bbf4675 f098082f502b44cd 0fbae670fa780fd3
38 b3bb8542b3376de5 002bb8e4cd989567 88df85b0bbe77514 1a081afc59fdbc2c
b49596c20feba7de 66adcfa249ac7bbd 8fbfd0162bbf4675 f098082f502b44cd
39 8e01e125b855d225 b3bb8542b3376de5 002bb8e4cd989567 88df85b0bbe77514
0c710a47ba6a567b b49596c20feba7de 66adcfa249ac7bbd 8fbfd0162bbf4675
40 b01521dd6a6be12c 8e01e125b855d225 b3bb8542b3376de5 002bb8e4cd989567
169008b3a4bb170b 0c710a47ba6a567b b49596c20feba7de 66adcfa249ac7bbd
41 e96f89dd48cbd851 b01521dd6a6be12c 8e01e125b855d225 b3bb8542b3376de5
f0996439e7b50cb1 169008b3a4bb170b 0c710a47ba6a567b b49596c20feba7de
42 bc05ba8de5d3c480 e96f89dd48cbd851 b01521dd6a6be12c 8e01e125b855d225
639cb938e14dc190 f0996439e7b50cb1 169008b3a4bb170b 0c710a47ba6a567b
43 35d7e7f41defcbd5 bc05ba8de5d3c480 e96f89dd48cbd851 b01521dd6a6be12c
cc5100997f5710f2 639cb938e14dc190 f0996439e7b50cb1 169008b3a4bb170b
44 c47c9d5c7ea8a234 35d7e7f41defcbd5 bc05ba8de5d3c480 e96f89dd48cbd851
858d832ae0e8911c cc5100997f5710f2 639cb938e14dc190 f0996439e7b50cb1
45 021fbadbabab5ac6 c47c9d5c7ea8a234 35d7e7f41defcbd5 bc05ba8de5d3c480
e95c2a57572d64d9 858d832ae0e8911c cc5100997f5710f2 639cb938e14dc190
46 f61e672694de2d67 021fbadbabab5ac6 c47c9d5c7ea8a234 35d7e7f41defcbd5
c6bc35740d8daa9a e95c2a57572d64d9 858d832ae0e8911c cc5100997f5710f2
47 6b69fc1bb482feac f61e672694de2d67 021fbadbabab5ac6 c47c9d5c7ea8a234
35264334c03ac8ad c6bc35740d8daa9a e95c2a57572d64d9 858d832ae0e8911c
48 571f323d96b3a047 6b69fc1bb482feac f61e672694de2d67 021fbadbabab5ac6
271580ed6c3e5650 35264334c03ac8ad c6bc35740d8daa9a e95c2a57572d64d9
49 ca9bd862c5050918 571f323d96b3a047 6b69fc1bb482feac f61e672694de2d67
dfe091dab182e645 271580ed6c3e5650 35264334c03ac8ad c6bc35740d8daa9a
50 813a43dd2c502043 ca9bd862c5050918 571f323d96b3a047 6b69fc1bb482feac
07a0d8ef821c5e1a dfe091dab182e645 271580ed6c3e5650 35264334c03ac8ad

51 d43f83727325dd77 813a43dd2c502043 ca9bd862c5050918 571f323d96b3a047
483f80a82eaee23e 07a0d8ef821c5e1a dfe091dab182e645 271580ed6c3e5650
52 03df11b32d42e203 d43f83727325dd77 813a43dd2c502043 ca9bd862c5050918
504f94e40591cffa 483f80a82eaee23e 07a0d8ef821c5e1a dfe091dab182e645
53 d63f68037ddf06aa 03df11b32d42e203 d43f83727325dd77 813a43dd2c502043
a6781efe1aa1ce02 504f94e40591cffa 483f80a82eaee23e 07a0d8ef821c5e1a
54 f650857b5babda4d d63f68037ddf06aa 03df11b32d42e203 d43f83727325dd77
9ccfb31a86df0f86 a6781efe1aa1ce02 504f94e40591cffa 483f80a82eaee23e
55 63b460e42748817e f650857b5babda4d d63f68037ddf06aa 03df11b32d42e203
c6b4dd2a9931c509 9ccfb31a86df0f86 a6781efe1aa1ce02 504f94e40591cffa
56 7a52912943d52b05 63b460e42748817e f650857b5babda4d d63f68037ddf06aa
d2e89bbd91e00be0 c6b4dd2a9931c509 9ccfb31a86df0f86 a6781efe1aa1ce02
57 4b81c3aec976ea4b 7a52912943d52b05 63b460e42748817e f650857b5babda4d
70505988124351ac d2e89bbd91e00be0 c6b4dd2a9931c509 9ccfb31a86df0f86
58 581ecb3355dcd9b8 4b81c3aec976ea4b 7a52912943d52b05 63b460e42748817e
6a3c9b0f71c8bf36 70505988124351ac d2e89bbd91e00be0 c6b4dd2a9931c509
59 2c074484ef1eac8c 581ecb3355dcd9b8 4b81c3aec976ea4b 7a52912943d52b05
4797cde4ed370692 6a3c9b0f71c8bf36 70505988124351ac d2e89bbd91e00be0
60 3857dfd2fc37d3ba 2c074484ef1eac8c 581ecb3355dcd9b8 4b81c3aec976ea4b
a6af4e9c9f807e51 4797cde4ed370692 6a3c9b0f71c8bf36 70505988124351ac
61 cfcd928c5424e2b6 3857dfd2fc37d3ba 2c074484ef1eac8c 581ecb3355dcd9b8
09aee5bda1644de5 a6af4e9c9f807e51 4797cde4ed370692 6a3c9b0f71c8bf36
62 a81dedbb9f19e643 cfcd928c5424e2b6 3857dfd2fc37d3ba 2c074484ef1eac8c
84058865d60a05fa 09aee5bda1644de5 a6af4e9c9f807e51 4797cde4ed370692
63 ab44e86276478d85 a81dedbb9f19e643 cfcd928c5424e2b6 3857dfd2fc37d3ba
cd881ee59ca6bc53 84058865d60a05fa 09aee5bda1644de5 a6af4e9c9f807e51
64 5a806d7e9821a501 ab44e86276478d85 a81dedbb9f19e643 cfcd928c5424e2b6
aa84b086688a5c45 cd881ee59ca6bc53 84058865d60a05fa 09aee5bda1644de5
65 eeb9c21bb0102598 5a806d7e9821a501 ab44e86276478d85 a81dedbb9f19e643
3b5fed0d6a1f96e1 aa84b086688a5c45 cd881ee59ca6bc53 84058865d60a05fa

```
66 46c4210ab2cc155d eeb9c21bb0102598 5a806d7e9821a501 ab44e86276478d85
   29fab5a7bff53366 3b5fed0d6a1f96e1 aa84b086688a5c45 cd881ee59ca6bc53
67 54ba35cf56a0340e 46c4210ab2cc155d eeb9c21bb0102598 5a806d7e9821a501
   1c66f46d95690bcf 29fab5a7bff53366 3b5fed0d6a1f96e1 aa84b086688a5c45
68 181839d609c79748 54ba35cf56a0340e 46c4210ab2cc155d eeb9c21bb0102598
   0ada78ba2d446140 1c66f46d95690bcf 29fab5a7bff53366 3b5fed0d6a1f96e1
69 fb6aaae5d0b6a447 181839d609c79748 54ba35cf56a0340e 46c4210ab2cc155d
   e3711cb6564d112d 0ada78ba2d446140 1c66f46d95690bcf 29fab5a7bff53366
70 7652c579cb60f19c fb6aaae5d0b6a447 181839d609c79748 54ba35cf56a0340e
   aff62c9665ff80fa e3711cb6564d112d 0ada78ba2d446140 1c66f46d95690bcf
71 f15e9664b2803575 7652c579cb60f19c fb6aaae5d0b6a447 181839d609c79748
   947c3dfafee570ef aff62c9665ff80fa e3711cb6564d112d 0ada78ba2d446140
72 358406d165aee9ab f15e9664b2803575 7652c579cb60f19c fb6aaae5d0b6a447
   8c7b5fd91a794ca0 947c3dfafee570ef aff62c9665ff80fa e3711cb6564d112d
73 20878dcd29cdfaf5 358406d165aee9ab f15e9664b2803575 7652c579cb60f19c
   054d3536539948d0 8c7b5fd91a794ca0 947c3dfafee570ef aff62c9665ff80fa
74 33d48dabb5521de2 20878dcd29cdfaf5 358406d165aee9ab f15e9664b2803575
   2ba18245b50de4cf 054d3536539948d0 8c7b5fd91a794ca0 947c3dfafee570ef
75 c8960e6be864b916 33d48dabb5521de2 20878dcd29cdfaf5 358406d165aee9ab
   995019a6ff3ba3de 2ba18245b50de4cf 054d3536539948d0 8c7b5fd91a794ca0
76 654ef9abec389ca9 c8960e6be864b916 33d48dabb5521de2 20878dcd29cdfaf5
   ceb9fc3691ce8326 995019a6ff3ba3de 2ba18245b50de4cf 054d3536539948d0
77 d67806db8b148677 654ef9abec389ca9 c8960e6be864b916 33d48dabb5521de2
   25c96a7768fb2aa3 ceb9fc3691ce8326 995019a6ff3ba3de 2ba18245b50de4cf
78 10d9c4c4295599f6 d67806db8b148677 654ef9abec389ca9 c8960e6be864b916
   9bb4d39778c07f9e 25c96a7768fb2aa3 ceb9fc3691ce8326 995019a6ff3ba3de
79 73a54f399fa4b1b2 10d9c4c4295599f6 d67806db8b148677 654ef9abec389ca9
   d08446aa79693ed7 9bb4d39778c07f9e 25c96a7768fb2aa3 ceb9fc3691ce8326
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

X_0 = 6a09e667f3bcc908 ⊎ 73a54f399fa4b1b2 = ddaf35a193617aba

X_1 = bb67ae8584caa73b ⊎ 10d9c4c4295599f6 = cc417349ae204131

X_2 = 3c6ef372fe94f82b ⊎ d67806db8b148677 = 12e6fa4e89a97ea2

X_3 = a54ff53a5f1d36f1 ⊎ 654ef9abec389ca9 = 0a9eeee64b55d39a

X_4 = 510e527fade682d1 ⊎ d08446aa79693ed7 = 2192992a274fc1a8

X_5 = 9b05688c2b3e6c1f ⊎ 9bb4d39778c07f9e = 36ba3c23a3feebbd

X_6 = 1f83d9abfb41bd6b ⊎ 25c96a7768fb2aa3 = 454d4423643ce80e

X_7 = 5be0cd19137e2179 ⊎ ceb9fc3691ce8326 = 2a9ac94fa54ca49f

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
ddaf35a193617aba cc417349ae204131 12e6fa4e89a97ea2 0a9eeee64b55d39a
2192992a274fc1a8 36ba3c23a3feebbd 454d4423643ce80e 2a9ac94fa54ca49f
```

الف-۵-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
107dbf389d9e9f71 a3a95f6c055b9251 bc5268c2be16d6c1 3492ea45b0199f33
09e16455ab1e9611 8e8a905d5597b720 38ddb372a8982604 6de66687bb420e7c
```

الف-۵-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII
´abcdefghijklmnopqrstuvwxyz´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
4dbff86cc2ca1bae 1e16468a05cb9881 c97f1753bce36190 34898faa1aabe429
955a1bf8ec483d74 21fe3c1646613a59 ed5441fb0f321389 f77f48a879c7b1f1
```

الف-۵-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
1e07be23c26a86ea 37ea810c8ec78093 52515a970e9253c2 6f536cfc7a9996c4
```

```
5c8370583e0a78fa 4a90041d71a4ceab 7423f19c71b9d5a3 e01249f0bebd5894
```

الف-۵-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار
‘1234567890’

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
72ec1ef1124a45b0 47e8b7c75a932195 135bb61de24ec0d1 914042246e0aec3a
2354e093d76f3048 b456764346900cb1 30d2a4fd5dd16abb 5e30bcb850dee843
```

الف-۵-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
204a8fc6dda82f0a 0ced7beb8e08a416 57c16ef468b228a8 279be331a703c335
96fd15c13b1b07f9 aa1d3bea57789ca0 31ad85c7a71dd703 54ec631238ca3445
```

الف-۵-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای 10^6 بار تکرار می‌شود.

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
e718483d0ce76964 4e2e42c7bc15b463 8e1f98b13b204428 5632a803afa973eb
de0ff244877ea60a 4cb0432ce577c31b eb009c5c2c49aa2e 4eadb217ad8cc09b
```

الف-۵-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سرخط بعد از اولین n)

بعد از فرایند لایه‌گذاری، دو بلوک ۱۶ کلمه‌ای مشتق شده از رشته-داده همانند ذیل است:

```
61626364 65666768 62636465 66676869 63646566 6768696a 64656667 68696a6b
65666768 696a6b6c 66676869 6a6b6c6d 6768696a 6b6c6d6e 68696a6b 6c6d6e6f
696a6b6c 6d6e6f70 6a6b6c6d 6e6f7071 6b6c6d6e 6f707172 6c6d6e6f 70717273
```

```
6d6e6f70 71727374 6e6f7071 72737475 80000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000380
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ که در طول پردازش اولین بلوک به‌دست آمده، آورده شده است.

```
 init 6a09e667f3bcc908 bb67ae8584caa73b 3c6ef372fe94f82b a54ff53a5f1d36f1
      510e527fade682d1 9b05688c2b3e6c1f 1f83d9abfb41bd6b 5be0cd19137e2179
    0 f6afce9d2263455d 6a09e667f3bcc908 bb67ae8584caa73b 3c6ef372fe94f82b
      58cb0218e01b86f9 510e527fade682d1 9b05688c2b3e6c1f 1f83d9abfb41bd6b
    1 0b7056a534ae5f62 f6afce9d2263455d 6a09e667f3bcc908 bb67ae8584caa73b
      f8c7198fe39e4c8c 58cb0218e01b86f9 510e527fade682d1 9b05688c2b3e6c1f
    2 2ca82233760c9942 0b7056a534ae5f62 f6afce9d2263455d 6a09e667f3bcc908
      303eccccd65953de f8c7198fe39e4c8c 58cb0218e01b86f9 510e527fade682d1
    3 a023f17ce52cda7b 2ca82233760c9942 0b7056a534ae5f62 f6afce9d2263455d
      ffdee5eedcc9ca42 303eccccd65953de f8c7198fe39e4c8c 58cb0218e01b86f9
    4 8f0a67d9d591a1a7 a023f17ce52cda7b 2ca82233760c9942 0b7056a534ae5f62
      cb4cfbb166505f2f ffdee5eedcc9ca42 303eccccd65953de f8c7198fe39e4c8c
    5 b466267371acc493 8f0a67d9d591a1a7 a023f17ce52cda7b 2ca82233760c9942
      73d6c84c54d399ee cb4cfbb166505f2f ffdee5eedcc9ca42 303eccccd65953de
    6 658269f1a312fccd b466267371acc493 8f0a67d9d591a1a7 a023f17ce52cda7b
      cdc40314975fb275 73d6c84c54d399ee cb4cfbb166505f2f ffdee5eedcc9ca42
    7 65e3519c5b88181b 658269f1a312fccd b466267371acc493 8f0a67d9d591a1a7
      a657850ab3970c5a cdc40314975fb275 73d6c84c54d399ee cb4cfbb166505f2f
    8 56604fbb4b6393ec 65e3519c5b88181b 658269f1a312fccd b466267371acc493
      e8b3be22fbe64df7 a657850ab3970c5a cdc40314975fb275 73d6c84c54d399ee
    9 c4562769a37d02c0 56604fbb4b6393ec 65e3519c5b88181b 658269f1a312fccd
      0062e70a1ef705c1 e8b3be22fbe64df7 a657850ab3970c5a cdc40314975fb275
```

10 27c0b4c9186e1736 c4562769a37d02c0 56604fbb4b6393ec 65e3519c5b88181b
bc9740477a18ae2d 0062e70a1ef705c1 e8b3be22fbe64df7 a657850ab3970c5a
11 f17f52fb02f4eb74 27c0b4c9186e1736 c4562769a37d02c0 56604fbb4b6393ec
be58522cb9590ee1 bc9740477a18ae2d 0062e70a1ef705c1 e8b3be22fbe64df7
12 f2c245ac903d4a35 f17f52fb02f4eb74 27c0b4c9186e1736 c4562769a37d02c0
49d5fa3a16dcd502 be58522cb9590ee1 bc9740477a18ae2d 0062e70a1ef705c1
13 9b04175ea8090daa f2c245ac903d4a35 f17f52fb02f4eb74 27c0b4c9186e1736
ec9c5e98ff98760d 49d5fa3a16dcd502 be58522cb9590ee1 bc9740477a18ae2d
14 481b8a6ee5e07031 9b04175ea8090daa f2c245ac903d4a35 f17f52fb02f4eb74
e4d35b613a5ac420 ec9c5e98ff98760d 49d5fa3a16dcd502 be58522cb9590ee1
15 9356ac3ec3e51459 481b8a6ee5e07031 9b04175ea8090daa f2c245ac903d4a35
701f17d27582443b e4d35b613a5ac420 ec9c5e98ff98760d 49d5fa3a16dcd502
16 b889ed34abd7aa37 9356ac3ec3e51459 481b8a6ee5e07031 9b04175ea8090daa
1d05d9ba779a1a78 701f17d27582443b e4d35b613a5ac420 ec9c5e98ff98760d
17 bf537b1f3edc7381 b889ed34abd7aa37 9356ac3ec3e51459 481b8a6ee5e07031
c362ff9cf932951d 1d05d9ba779a1a78 701f17d27582443b e4d35b613a5ac420
18 d4e44d54e8242ad8 bf537b1f3edc7381 b889ed34abd7aa37 9356ac3ec3e51459
459e4e6888919f36 c362ff9cf932951d 1d05d9ba779a1a78 701f17d27582443b
19 05f3fba454e5de3d d4e44d54e8242ad8 bf537b1f3edc7381 b889ed34abd7aa37
caed4b5fa322b984 459e4e6888919f36 c362ff9cf932951d 1d05d9ba779a1a78
20 cdb73772dc0248bf 05f3fba454e5de3d d4e44d54e8242ad8 bf537b1f3edc7381
dc8049afa6acd502 caed4b5fa322b984 459e4e6888919f36 c362ff9cf932951d
21 1d47a3268ff677ed cdb73772dc0248bf 05f3fba454e5de3d d4e44d54e8242ad8
8407818e9b28cc12 dc8049afa6acd502 caed4b5fa322b984 459e4e6888919f36
22 af4e23eb622d0df4 1d47a3268ff677ed cdb73772dc0248bf 05f3fba454e5de3d
64b5ae5424598428 8407818e9b28cc12 dc8049afa6acd502 caed4b5fa322b984
23 be50606778de14a6 af4e23eb622d0df4 1d47a3268ff677ed cdb73772dc0248bf
0a5d727cc92e7adb 64b5ae5424598428 8407818e9b28cc12 dc8049afa6acd502
24 821e44f6678ac478 be50606778de14a6 af4e23eb622d0df4 1d47a3268ff677ed
f367e596d0a038a5 0a5d727cc92e7adb 64b5ae5424598428 8407818e9b28cc12

25 0c852b1359a77c18 821e44f6678ac478 be50606778de14a6 af4e23eb622d0df4
6dec8a3396a80c3f f367e596d0a038a5 0a5d727cc92e7adb 64b5ae5424598428
26 ebb574fad4b7a7e4 0c852b1359a77c18 821e44f6678ac478 be50606778de14a6
a241e7efc1eb6ff9 6dec8a3396a80c3f f367e596d0a038a5 0a5d727cc92e7adb
27 a092821c3cdf08da ebb574fad4b7a7e4 0c852b1359a77c18 821e44f6678ac478
c84e849917a7c08e a241e7efc1eb6ff9 6dec8a3396a80c3f f367e596d0a038a5
28 82ba2e1a2df2a4f1 a092821c3cdf08da ebb574fad4b7a7e4 0c852b1359a77c18
61845f6924789851 c84e849917a7c08e a241e7efc1eb6ff9 6dec8a3396a80c3f
29 1959ad991c63d06a 82ba2e1a2df2a4f1 a092821c3cdf08da ebb574fad4b7a7e4
231faf24910a891a 61845f6924789851 c84e849917a7c08e a241e7efc1eb6ff9
30 9b32d4cacd9a625b 1959ad991c63d06a 82ba2e1a2df2a4f1 a092821c3cdf08da
533066919d608799 231faf24910a891a 61845f6924789851 c84e849917a7c08e
31 dc55339f4d841965 9b32d4cacd9a625b 1959ad991c63d06a 82ba2e1a2df2a4f1
e2517f359998a58d 533066919d608799 231faf24910a891a 61845f6924789851
32 fdebb1283b12514f dc55339f4d841965 9b32d4cacd9a625b 1959ad991c63d06a
b1989170a183c661 e2517f359998a58d 533066919d608799 231faf24910a891a
33 b44c7975a83e3334 fdebb1283b12514f dc55339f4d841965 9b32d4cacd9a625b
009ad175b8d588a4 b1989170a183c661 e2517f359998a58d 533066919d608799
34 0bac61bfc53d18b7 b44c7975a83e3334 fdebb1283b12514f dc55339f4d841965
a7d5416d690557b8 009ad175b8d588a4 b1989170a183c661 e2517f359998a58d
35 392893c22e75856a 0bac61bfc53d18b7 b44c7975a83e3334 fdebb1283b12514f
7a7c9eb7bc813248 a7d5416d690557b8 009ad175b8d588a4 b1989170a183c661
36 824408631432e09b 392893c22e75856a 0bac61bfc53d18b7 b44c7975a83e3334
5e696a9fda56d6bf 7a7c9eb7bc813248 a7d5416d690557b8 009ad175b8d588a4
37 a64162f151a8c1cb 824408631432e09b 392893c22e75856a 0bac61bfc53d18b7
0f57062401dc680b 5e696a9fda56d6bf 7a7c9eb7bc813248 a7d5416d690557b8
38 922537abad1e95a1 a64162f151a8c1cb 824408631432e09b 392893c22e75856a
4f4c193d435ff721 0f57062401dc680b 5e696a9fda56d6bf 7a7c9eb7bc813248
39 b80591f6fbfadcde 922537abad1e95a1 a64162f151a8c1cb 824408631432e09b
00f4407c0f37237e 4f4c193d435ff721 0f57062401dc680b 5e696a9fda56d6bf

40 08f151f4b8d0fa2e b80591f6fbfadcde 922537abad1e95a1 a64162f151a8c1cb
ec8b96fe402094cd 00f4407c0f37237e 4f4c193d435ff721 0f57062401dc680b
41 12b5fcc2b68f65c0 08f151f4b8d0fa2e b80591f6fbfadcde 922537abad1e95a1
d688101dfd24a148 ec8b96fe402094cd 00f4407c0f37237e 4f4c193d435ff721
42 a71bf5bd64289948 12b5fcc2b68f65c0 08f151f4b8d0fa2e b80591f6fbfadcde
e052bfb7a6945939 d688101dfd24a148 ec8b96fe402094cd 00f4407c0f37237e
43 890c2cd670c4aea3 a71bf5bd64289948 12b5fcc2b68f65c0 08f151f4b8d0fa2e
dd13e4edeeff00e7 e052bfb7a6945939 d688101dfd24a148 ec8b96fe402094cd
44 ca61990b43297ffc 890c2cd670c4aea3 a71bf5bd64289948 12b5fcc2b68f65c0
139aa55c51d9ee5f dd13e4edeeff00e7 e052bfb7a6945939 d688101dfd24a148
45 7196e8fa538ba4bf ca61990b43297ffc 890c2cd670c4aea3 a71bf5bd64289948
046735513cdd14d3 139aa55c51d9ee5f dd13e4edeeff00e7 e052bfb7a6945939
46 1f0720944dbeb6a4 7196e8fa538ba4bf ca61990b43297ffc 890c2cd670c4aea3
a41eb7e5a27588e3 046735513cdd14d3 139aa55c51d9ee5f dd13e4edeeff00e7
47 d6d4f8608b8ab199 1f0720944dbeb6a4 7196e8fa538ba4bf ca61990b43297ffc
24b9c216f915da60 a41eb7e5a27588e3 046735513cdd14d3 139aa55c51d9ee5f
48 88761eb67845978e d6d4f8608b8ab199 1f0720944dbeb6a4 7196e8fa538ba4bf
9fe22e39448d50ed 24b9c216f915da60 a41eb7e5a27588e3 046735513cdd14d3
49 7d40e6be47d85702 88761eb67845978e d6d4f8608b8ab199 1f0720944dbeb6a4
d9c900e01968c33e 9fe22e39448d50ed 24b9c216f915da60 a41eb7e5a27588e3
50 7d0d988df5768598 7d40e6be47d85702 88761eb67845978e d6d4f8608b8ab199
2ec2e522a7c7d12c d9c900e01968c33e 9fe22e39448d50ed 24b9c216f915da60
51 48a8b60575b37f31 7d0d988df5768598 7d40e6be47d85702 88761eb67845978e
7059f9bc8c88a373 2ec2e522a7c7d12c d9c900e01968c33e 9fe22e39448d50ed
52 6bc425af294bbf79 48a8b60575b37f31 7d0d988df5768598 7d40e6be47d85702
6a8143b1716ee33d 7059f9bc8c88a373 2ec2e522a7c7d12c d9c900e01968c33e
53 307a456158ee8849 6bc425af294bbf79 48a8b60575b37f31 7d0d988df5768598
4372e85c16ee4440 6a8143b1716ee33d 7059f9bc8c88a373 2ec2e522a7c7d12c
54 af36382c8fd716be 307a456158ee8849 6bc425af294bbf79 48a8b60575b37f31
a8f8b0033187a916 4372e85c16ee4440 6a8143b1716ee33d 7059f9bc8c88a373

55 810ebee951c64ca1 af36382c8fd716be 307a456158ee8849 6bc425af294bbf79
16a64f5997b9cca6 a8f8b0033187a916 4372e85c16ee4440 6a8143b1716ee33d
56 2dd7659f1b4d13cd 810ebee951c64ca1 af36382c8fd716be 307a456158ee8849
5da6793bb7286a4b 16a64f5997b9cca6 a8f8b0033187a916 4372e85c16ee4440
57 5ac712acff4b98be 2dd7659f1b4d13cd 810ebee951c64ca1 af36382c8fd716be
91f6395b301adbfd 5da6793bb7286a4b 16a64f5997b9cca6 a8f8b0033187a916
58 c1af358833cb03c0 5ac712acff4b98be 2dd7659f1b4d13cd 810ebee951c64ca1
d4883c0c21dda190 91f6395b301adbfd 5da6793bb7286a4b 16a64f5997b9cca6
59 88a306074d388c7d c1af358833cb03c0 5ac712acff4b98be 2dd7659f1b4d13cd
9fc52468b897f9c8 d4883c0c21dda190 91f6395b301adbfd 5da6793bb7286a4b
60 f11bfd0cf67d3040 88a306074d388c7d c1af358833cb03c0 5ac712acff4b98be
47efb6407f74d318 9fc52468b897f9c8 d4883c0c21dda190 91f6395b301adbfd
61 1f065e7828ed4e1b f11bfd0cf67d3040 88a306074d388c7d c1af358833cb03c0
7481899904a4ce23 47efb6407f74d318 9fc52468b897f9c8 d4883c0c21dda190
62 aebde39f2bc42ec1 1f065e7828ed4e1b f11bfd0cf67d3040 88a306074d388c7d
62ab526ff177a988 7481899904a4ce23 47efb6407f74d318 9fc52468b897f9c8
63 d35a94706e3e5df2 aebde39f2bc42ec1 1f065e7828ed4e1b f11bfd0cf67d3040
53f92b648d5d815c 62ab526ff177a988 7481899904a4ce23 47efb6407f74d318
64 d72d727c53e09ab9 d35a94706e3e5df2 aebde39f2bc42ec1 1f065e7828ed4e1b
10746426ba9824f4 53f92b648d5d815c 62ab526ff177a988 7481899904a4ce23
65 3a7235e5a4051d94 d72d727c53e09ab9 d35a94706e3e5df2 aebde39f2bc42ec1
afe455daec5c2b00 10746426ba9824f4 53f92b648d5d815c 62ab526ff177a988
66 f7f510fe73ef7e76 3a7235e5a4051d94 d72d727c53e09ab9 d35a94706e3e5df2
f1202c0bb7c4583f afe455daec5c2b00 10746426ba9824f4 53f92b648d5d815c
67 23c2acfb393523e9 f7f510fe73ef7e76 3a7235e5a4051d94 d72d727c53e09ab9
a0bc2a61044ac12e f1202c0bb7c4583f afe455daec5c2b00 10746426ba9824f4
68 0307d241a1ed7121 23c2acfb393523e9 f7f510fe73ef7e76 3a7235e5a4051d94
fad5f38f1e0aea12 a0bc2a61044ac12e f1202c0bb7c4583f afe455daec5c2b00
69 191814d82f0a16fb 0307d241a1ed7121 23c2acfb393523e9 f7f510fe73ef7e76
39d325086e66e200 fad5f38f1e0aea12 a0bc2a61044ac12e f1202c0bb7c4583f

```
70 0a1ed41b6da18c01 191814d82f0a16fb 0307d241a1ed7121 23c2acfb393523e9
   b3d3521e166e5df1 39d325086e66e200 fad5f38f1e0aea12 a0bc2a61044ac12e
71 8a3f07db93f6c827 0a1ed41b6da18c01 191814d82f0a16fb 0307d241a1ed7121
   6b370074be040ed7 b3d3521e166e5df1 39d325086e66e200 fad5f38f1e0aea12
72 002744d87ef80d28 8a3f07db93f6c827 0a1ed41b6da18c01 191814d82f0a16fb
   8c5a245de2d72fe6 6b370074be040ed7 b3d3521e166e5df1 39d325086e66e200
73 778dc7880a4a2aa0 002744d87ef80d28 8a3f07db93f6c827 0a1ed41b6da18c01
   45a375b466e5e342 8c5a245de2d72fe6 6b370074be040ed7 b3d3521e166e5df1
74 a3f11de5ede05b11 778dc7880a4a2aa0 002744d87ef80d28 8a3f07db93f6c827
   f5bbf52f1ab7cc05 45a375b466e5e342 8c5a245de2d72fe6 6b370074be040ed7
75 629c8ae6ecd8af4b a3f11de5ede05b11 778dc7880a4a2aa0 002744d87ef80d28
   5a8fe5919d3cf136 f5bbf52f1ab7cc05 45a375b466e5e342 8c5a245de2d72fe6
76 c9a8c1e2d063ce94 629c8ae6ecd8af4b a3f11de5ede05b11 778dc7880a4a2aa0
   aacd089bfae8faf9 5a8fe5919d3cf136 f5bbf52f1ab7cc05 45a375b466e5e342
77 c517cba6a09bb26a c9a8c1e2d063ce94 629c8ae6ecd8af4b a3f11de5ede05b11
   e1682bd33c8f8e23 aacd089bfae8faf9 5a8fe5919d3cf136 f5bbf52f1ab7cc05
78 11e3570e06e3b74e c517cba6a09bb26a c9a8c1e2d063ce94 629c8ae6ecd8af4b
   075aabbade34fd01 e1682bd33c8f8e23 aacd089bfae8faf9 5a8fe5919d3cf136
79 d90f1b1237b3a561 11e3570e06e3b74e c517cba6a09bb26a c9a8c1e2d063ce94
   867983f69d3a3ad1 075aabbade34fd01 e1682bd33c8f8e23 aacd089bfae8faf9
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را در فرایند اولین بلوک نمایش می‌دهند:

X_0 = 6a09e667f3bcc908 ⊎ d90f1b1237b3a561 = 4319017a2b706e69

X_1 = bb67ae8584caa73b ⊎ 11e3570e06e3b74e = cd4b05938bae5e89

X_2 = 3c6ef372fe94f82b ⊎ c517cba6a09bb26a = 0186bf199f30aa95

X_3 = a54ff53a5f1d36f1 ⊎ c9a8c1e2d063ce94 = 6ef8b71d2f810585

X_4 = 510e527fade682d1 ⊎ 867983f69d3a3ad1 = d787d6764b20bda2

X_5 = 9b05688c2b3e6c1f ⊎ 075aabbade34fd01 = a260144709736920

X_6 = 1f83d9abfb41bd6b ⊎ e1682bd33c8f8e23 = 00ec057f37d14b8e

X_7 = 5be0cd19137e2179 ⊎ aacd089bfae8faf9 = 06add5b50e671c72

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ که در طول پردازش دومین بلوک به‌دست آمده، آورده شده است.

```
init 4319017a2b706e69 cd4b05938bae5e89 0186bf199f30aa95 6ef8b71d2f810585
     d787d6764b20bda2 a260144709736920 00ec057f37d14b8e 06add5b50e671c72
   0 b8fdb92bdfb187e8 4319017a2b706e69 cd4b05938bae5e89 0186bf199f30aa95
     1d5f4d5ad031b8e6 d787d6764b20bda2 a260144709736920 00ec057f37d14b8e
   1 6eb90718369c5cd7 b8fdb92bdfb187e8 4319017a2b706e69 cd4b05938bae5e89
     4b9b4877d987b0fe 1d5f4d5ad031b8e6 d787d6764b20bda2 a260144709736920
   2 c83451f2335d5144 6eb90718369c5cd7 b8fdb92bdfb187e8 4319017a2b706e69
     d6b67350e0781e99 4b9b4877d987b0fe 1d5f4d5ad031b8e6 d787d6764b20bda2
   3 28ec1deb2a9ee6e3 c83451f2335d5144 6eb90718369c5cd7 b8fdb92bdfb187e8
     25e3136be5999b8c d6b67350e0781e99 4b9b4877d987b0fe 1d5f4d5ad031b8e6
   4 806abd86c0479e5b 28ec1deb2a9ee6e3 c83451f2335d5144 6eb90718369c5cd7
     1b8f7670eab1cf89 25e3136be5999b8c d6b67350e0781e99 4b9b4877d987b0fe
   5 234788f8a54aed38 806abd86c0479e5b 28ec1deb2a9ee6e3 c83451f2335d5144
     4fabe51c67d5d156 1b8f7670eab1cf89 25e3136be5999b8c d6b67350e0781e99
   6 01264f18257b5e2c 234788f8a54aed38 806abd86c0479e5b 28ec1deb2a9ee6e3
     1c3506096b99de50 4fabe51c67d5d156 1b8f7670eab1cf89 25e3136be5999b8c
   7 5b14f38104dde991 01264f18257b5e2c 234788f8a54aed38 806abd86c0479e5b
     13f8bfdc4001c362 1c3506096b99de50 4fabe51c67d5d156 1b8f7670eab1cf89
   8 f522574a41b2aac6 5b14f38104dde991 01264f18257b5e2c 234788f8a54aed38
     63a5f09617622ed2 13f8bfdc4001c362 1c3506096b99de50 4fabe51c67d5d156
   9 6ec258b855afae5a f522574a41b2aac6 5b14f38104dde991 01264f18257b5e2c
     211e271d92770b36 63a5f09617622ed2 13f8bfdc4001c362 1c3506096b99de50
  10 9364214ba48b416c 6ec258b855afae5a f522574a41b2aac6 5b14f38104dde991
     d64dcb6ec0fe5bac 211e271d92770b36 63a5f09617622ed2 13f8bfdc4001c362
  11 082ba62147ecbbd5 9364214ba48b416c 6ec258b855afae5a f522574a41b2aac6
```

34fe78473b61266e d64dcb6ec0fe5bac 211e271d92770b36 63a5f09617622ed2
12 5790f6ba82bba809 082ba62147ecbbd5 9364214ba48b416c 6ec258b855afae5a
d491e309141dcaa3 34fe78473b61266e d64dcb6ec0fe5bac 211e271d92770b36
13 a6b8aefd086d33ce 5790f6ba82bba809 082ba62147ecbbd5 9364214ba48b416c
044943c2992cc0f0 d491e309141dcaa3 34fe78473b61266e d64dcb6ec0fe5bac
14 bf2324a9a363abe7 a6b8aefd086d33ce 5790f6ba82bba809 082ba62147ecbbd5
0cf5f4bde5977c54 044943c2992cc0f0 d491e309141dcaa3 34fe78473b61266e
15 00e8e32076a61aff bf2324a9a363abe7 a6b8aefd086d33ce 5790f6ba82bba809
43bf4eb269a2650c 0cf5f4bde5977c54 044943c2992cc0f0 d491e309141dcaa3
16 f0376dff66fff4a7 00e8e32076a61aff bf2324a9a363abe7 a6b8aefd086d33ce
69fa5896969e85b8 43bf4eb269a2650c 0cf5f4bde5977c54 044943c2992cc0f0
17 2fad194272cda857 f0376dff66fff4a7 00e8e32076a61aff bf2324a9a363abe7
ddb519d663b7b6ec 69fa5896969e85b8 43bf4eb269a2650c 0cf5f4bde5977c54
18 9ae56936e95325ac 2fad194272cda857 f0376dff66fff4a7 00e8e32076a61aff
04ceb04676619057 ddb519d663b7b6ec 69fa5896969e85b8 43bf4eb269a2650c
19 d94ccb853f53433b 9ae56936e95325ac 2fad194272cda857 f0376dff66fff4a7
dcdc0f45813fb5a2 04ceb04676619057 ddb519d663b7b6ec 69fa5896969e85b8
20 837f8075d2945995 d94ccb853f53433b 9ae56936e95325ac 2fad194272cda857
272b5f79a91419d8 dcdc0f45813fb5a2 04ceb04676619057 ddb519d663b7b6ec
21 786bde689f7aa62d 837f8075d2945995 d94ccb853f53433b 9ae56936e95325ac
566586e69ad3f487 272b5f79a91419d8 dcdc0f45813fb5a2 04ceb04676619057
22 276457f01812aa6f 786bde689f7aa62d 837f8075d2945995 d94ccb853f53433b
e78fb8b0dfbbc62f 566586e69ad3f487 272b5f79a91419d8 dcdc0f45813fb5a2
23 0de519f5d6c2c298 276457f01812aa6f 786bde689f7aa62d 837f8075d2945995
5ca3e5cd1a30b954 e78fb8b0dfbbc62f 566586e69ad3f487 272b5f79a91419d8
24 54314dff825e2b22 0de519f5d6c2c298 276457f01812aa6f 786bde689f7aa62d
b81a51e0c96ccf77 5ca3e5cd1a30b954 e78fb8b0dfbbc62f 566586e69ad3f487
25 5d3f98dd7b29c363 54314dff825e2b22 0de519f5d6c2c298 276457f01812aa6f
95d49494f5a0d14a b81a51e0c96ccf77 5ca3e5cd1a30b954 e78fb8b0dfbbc62f
26 5e9da426aa7d4a58 5d3f98dd7b29c363 54314dff825e2b22 0de519f5d6c2c298

d22cccad2e391cd4 95d49494f5a0d14a b81a51e0c96ccf77 5ca3e5cd1a30b954
27 3b62dd973298ea43 5e9da426aa7d4a58 5d3f98dd7b29c363 54314dff825e2b22
aceb5d06101e514e d22cccad2e391cd4 95d49494f5a0d14a b81a51e0c96ccf77
28 fd258ff809b2253d 3b62dd973298ea43 5e9da426aa7d4a58 5d3f98dd7b29c363
26c991e85352da6f aceb5d06101e514e d22cccad2e391cd4 95d49494f5a0d14a
29 b462a20846af417d fd258ff809b2253d 3b62dd973298ea43 5e9da426aa7d4a58
291eee54c034c326 26c991e85352da6f aceb5d06101e514e d22cccad2e391cd4
30 d5471e3dc7171224 b462a20846af417d fd258ff809b2253d 3b62dd973298ea43
0aaf99c59e7fadbd 291eee54c034c326 26c991e85352da6f aceb5d06101e514e
31 9ace856ba1290e6e d5471e3dc7171224 b462a20846af417d fd258ff809b2253d
658f0bea63804d05 0aaf99c59e7fadbd 291eee54c034c326 26c991e85352da6f
32 80a0d154506b37c4 9ace856ba1290e6e d5471e3dc7171224 b462a20846af417d
bbe6e3b3bb7fefab 658f0bea63804d05 0aaf99c59e7fadbd 291eee54c034c326
33 fb90a8a76dea1bfe 80a0d154506b37c4 9ace856ba1290e6e d5471e3dc7171224
65234d5b5049e665 bbe6e3b3bb7fefab 658f0bea63804d05 0aaf99c59e7fadbd
34 f517b690d940a294 fb90a8a76dea1bfe 80a0d154506b37c4 9ace856ba1290e6e
e4dd663f44d313bc 65234d5b5049e665 bbe6e3b3bb7fefab 658f0bea63804d05
35 b70883992932880d f517b690d940a294 fb90a8a76dea1bfe 80a0d154506b37c4
dc5dd7c12b1cb6e3 e4dd663f44d313bc 65234d5b5049e665 bbe6e3b3bb7fefab
36 b2a2be77b0fcf3bf b70883992932880d f517b690d940a294 fb90a8a76dea1bfe
50fca57291e19874 dc5dd7c12b1cb6e3 e4dd663f44d313bc 65234d5b5049e665
37 8575839b0f08472b b2a2be77b0fcf3bf b70883992932880d f517b690d940a294
bd7176bd099bb2f2 50fca57291e19874 dc5dd7c12b1cb6e3 e4dd663f44d313bc
38 4405d2765de0adfc 8575839b0f08472b b2a2be77b0fcf3bf b70883992932880d
7ca4916f2cd8db10 bd7176bd099bb2f2 50fca57291e19874 dc5dd7c12b1cb6e3
39 eec6fca5aa657661 4405d2765de0adfc 8575839b0f08472b b2a2be77b0fcf3bf
7be0b7e70bdabe53 7ca4916f2cd8db10 bd7176bd099bb2f2 50fca57291e19874
40 bb3fcd7585b59e32 eec6fca5aa657661 4405d2765de0adfc 8575839b0f08472b
2201c7cbd34e31fe 7be0b7e70bdabe53 7ca4916f2cd8db10 bd7176bd099bb2f2
41 0e109efc47927341 bb3fcd7585b59e32 eec6fca5aa657661 4405d2765de0adfc

d43e5686506fa05d 2201c7cbd34e31fe 7be0b7e70bdabe53 7ca4916f2cd8db10

42 55c0dba83bcdc6e0 0e109efc47927341 bb3fcd7585b59e32 eec6fca5aa657661

5b634502f1671535 d43e5686506fa05d 2201c7cbd34e31fe 7be0b7e70bdabe53

43 f5756f847bfaef67 55c0dba83bcdc6e0 0e109efc47927341 bb3fcd7585b59e32

e2d307fd94f4818a 5b634502f1671535 d43e5686506fa05d 2201c7cbd34e31fe

44 f1438c9cf271c06e f5756f847bfaef67 55c0dba83bcdc6e0 0e109efc47927341

ad8ac1ed966b2dc6 e2d307fd94f4818a 5b634502f1671535 d43e5686506fa05d

45 a7dcaffdbefb9d4a f1438c9cf271c06e f5756f847bfaef67 55c0dba83bcdc6e0

9e46e9f915099c34 ad8ac1ed966b2dc6 e2d307fd94f4818a 5b634502f1671535

46 985ba373680b8e94 a7dcaffdbefb9d4a f1438c9cf271c06e f5756f847bfaef67

7d4c0abc676b1a8b 9e46e9f915099c34 ad8ac1ed966b2dc6 e2d307fd94f4818a

47 807f45784852303f 985ba373680b8e94 a7dcaffdbefb9d4a f1438c9cf271c06e

082ee70d3f352aac 7d4c0abc676b1a8b 9e46e9f915099c34 ad8ac1ed966b2dc6

48 d9c523173b1a1e05 807f45784852303f 985ba373680b8e94 a7dcaffdbefb9d4a

e301dca32c44ca05 082ee70d3f352aac 7d4c0abc676b1a8b 9e46e9f915099c34

49 b6df019ca515cafb d9c523173b1a1e05 807f45784852303f 985ba373680b8e94

754b3a461a665640 e301dca32c44ca05 082ee70d3f352aac 7d4c0abc676b1a8b

50 427a642921b2e645 b6df019ca515cafb d9c523173b1a1e05 807f45784852303f

08a30fefe981f2ec 754b3a461a665640 e301dca32c44ca05 082ee70d3f352aac

51 7aab58dbe1b9df7b 427a642921b2e645 b6df019ca515cafb d9c523173b1a1e05

2749c52d0b3d1225 08a30fefe981f2ec 754b3a461a665640 e301dca32c44ca05

52 974ddd552aec16ce 7aab58dbe1b9df7b 427a642921b2e645 b6df019ca515cafb

a9e6cbfb416a591f 2749c52d0b3d1225 08a30fefe981f2ec 754b3a461a665640

53 55e0b99d4404f6ca 974ddd552aec16ce 7aab58dbe1b9df7b 427a642921b2e645

6c24ad697b41b1b9 a9e6cbfb416a591f 2749c52d0b3d1225 08a30fefe981f2ec

54 901f632579ee1eee 55e0b99d4404f6ca 974ddd552aec16ce 7aab58dbe1b9df7b

4ee99476db1bb7a9 6c24ad697b41b1b9 a9e6cbfb416a591f 2749c52d0b3d1225

55 f90db9f292a60463 901f632579ee1eee 55e0b99d4404f6ca 974ddd552aec16ce

5401644992a1f8b8 4ee99476db1bb7a9 6c24ad697b41b1b9 a9e6cbfb416a591f

56 9b906a7df1007357 f90db9f292a60463 901f632579ee1eee 55e0b99d4404f6ca

f5e402ee21db8915 5401644992a1f8b8 4ee99476db1bb7a9 6c24ad697b41b1b9
57 71a0a998fb48c0fc 9b906a7df1007357 f90db9f292a60463 901f632579ee1eee
96bece755cd203cb f5e402ee21db8915 5401644992a1f8b8 4ee99476db1bb7a9
58 c25e798e50752535 71a0a998fb48c0fc 9b906a7df1007357 f90db9f292a60463
9d548440d8e110f2 96bece755cd203cb f5e402ee21db8915 5401644992a1f8b8
59 1ce4f2591812e6ae c25e798e50752535 71a0a998fb48c0fc 9b906a7df1007357
b27252537a83cf27 9d548440d8e110f2 96bece755cd203cb f5e402ee21db8915
60 c1700e250dc6ffed 1ce4f2591812e6ae c25e798e50752535 71a0a998fb48c0fc
970088839126bda5 b27252537a83cf27 9d548440d8e110f2 96bece755cd203cb
61 f8e6924412fd0c64 c1700e250dc6ffed 1ce4f2591812e6ae c25e798e50752535
d50cf4f73910e3ee 970088839126bda5 b27252537a83cf27 9d548440d8e110f2
62 d53e0a39eee47528 f8e6924412fd0c64 c1700e250dc6ffed 1ce4f2591812e6ae
1b6d7234ace15d7d d50cf4f73910e3ee 970088839126bda5 b27252537a83cf27
63 3960545ab926c0d5 d53e0a39eee47528 f8e6924412fd0c64 c1700e250dc6ffed
9eabb5618b4fcd13 1b6d7234ace15d7d d50cf4f73910e3ee 970088839126bda5
64 b2c164d71abb92fe 3960545ab926c0d5 d53e0a39eee47528 f8e6924412fd0c64
f1736fbbfb6ebe72 9eabb5618b4fcd13 1b6d7234ace15d7d d50cf4f73910e3ee
65 4d979e985b067e75 b2c164d71abb92fe 3960545ab926c0d5 d53e0a39eee47528
d1fb300f35992350 f1736fbbfb6ebe72 9eabb5618b4fcd13 1b6d7234ace15d7d
66 59d0238ce137abd7 4d979e985b067e75 b2c164d71abb92fe 3960545ab926c0d5
5f3c64b7546e2cec d1fb300f35992350 f1736fbbfb6ebe72 9eabb5618b4fcd13
67 bf8d9453b9876b0a 59d0238ce137abd7 4d979e985b067e75 b2c164d71abb92fe
6c27893a31b0e07e 5f3c64b7546e2cec d1fb300f35992350 f1736fbbfb6ebe72
68 c45dd4a2d2fea059 bf8d9453b9876b0a 59d0238ce137abd7 4d979e985b067e75
48253e21b26d8cf9 6c27893a31b0e07e 5f3c64b7546e2cec d1fb300f35992350
69 e08471946c17b0b6 c45dd4a2d2fea059 bf8d9453b9876b0a 59d0238ce137abd7
714e2adf4e23ff24 48253e21b26d8cf9 6c27893a31b0e07e 5f3c64b7546e2cec
70 b4838c1c28fee7bc e08471946c17b0b6 c45dd4a2d2fea059 bf8d9453b9876b0a
371f12f333f7e5b9 714e2adf4e23ff24 48253e21b26d8cf9 6c27893a31b0e07e
71 851cf60a77f6e6d1 b4838c1c28fee7bc e08471946c17b0b6 c45dd4a2d2fea059

a2a475deac0e8b42 371f12f333f7e5b9 714e2adf4e23ff24 48253e21b26d8cf9

72 f53d23c50249af2d 851cf60a77f6e6d1 b4838c1c28fee7bc e08471946c17b0b6

1e99cae9d4cf0409 a2a475deac0e8b42 371f12f333f7e5b9 714e2adf4e23ff24

73 b81e85d427045550 f53d23c50249af2d 851cf60a77f6e6d1 b4838c1c28fee7bc

f5794711faa60f63 1e99cae9d4cf0409 a2a475deac0e8b42 371f12f333f7e5b9

74 ae70c7d11ea84a83 b81e85d427045550 f53d23c50249af2d 851cf60a77f6e6d1

dc0d633411c289b2 f5794711faa60f63 1e99cae9d4cf0409 a2a475deac0e8b42

75 5c54592e13c76135 ae70c7d11ea84a83 b81e85d427045550 f53d23c50249af2d

1620dd5479e94b9b dc0d633411c289b2 f5794711faa60f63 1e99cae9d4cf0409

76 03a0f79087078a93 5c54592e13c76135 ae70c7d11ea84a83 b81e85d427045550

57e90fa678e4cc97 1620dd5479e94b9b dc0d633411c289b2 f5794711faa60f63

77 8df0baad4c6ed50c 03a0f79087078a93 5c54592e13c76135 ae70c7d11ea84a83

c6e7246f7f0bdac6 57e90fa678e4cc97 1620dd5479e94b9b dc0d633411c289b2

78 bfa9f194894db5b6 8df0baad4c6ed50c 03a0f79087078a93 5c54592e13c76135

90bb8597bb41da1a c6e7246f7f0bdac6 57e90fa678e4cc97 1620dd5479e94b9b

79 4b7c99fbaf72a571 bfa9f194894db5b6 8df0baad4c6ed50c 03a0f79087078a93

78955227fde03a42 90bb8597bb41da1a c6e7246f7f0bdac6 57e90fa678e4cc97

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

X_0 = 4319017a2b706e69 ⊎ 4b7c99fbaf72a571 = 8e959b75dae313da

X_1 = cd4b05938bae5e89 ⊎ bfa9f194894db5b6 = 8cf4f72814fc143f

X_2 = 0186bf199f30aa95 ⊎ 8df0baad4c6ed50c = 8f7779c6eb9f7fa1

X_3 = 6ef8b71d2f810585 ⊎ 03a0f79087078a93 = 7299aeadb6889018

X_4 = d787d6764b20bda2 ⊎ 78955227fde03a42 = 501d289e4900f7e4

X_5 = a260144709736920 ⊎ 90bb8597bb41da1a = 331b99dec4b5433a

X_6 = 00ec057f37d14b8e ⊎ c6e7246f7f0bdac6 = c7d329eeb6dd2654

X_7 = 06add5b50e671c72 ⊎ 57e90fa678e4cc97 = 5e96e55b874be909

مقدار درهم برای این پیام به صورت زیر است:

8e959b75dae313da 8cf4f72814fc143f 8f7779c6eb9f7fa1 7299aeadb6889018

```
501d289e4900f7e4 331b99dec4b5433a c7d329eeb6dd2654 5e96e55b874be909
```

الف-۵-۱۱ مثال۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
c50e7a500d4058bf 530ec603b66b032a 989a3e033a340090 dc51086cfd8cb222
09027932ea830f9b 6bc09dafe882f908 38c2c91018245904 828c1232fc0942eb
```

الف-۶ تابع درهم‌ساز اختصاصی ۶

الف-۶-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
38b060a751ac9638 4cd9327eb1b1e36a 21fdb71114be0743 4c0cc7bf63f6e1da
274edebfe76f65fb d51ad2f14898b95b
```

الف-۶-۲ مثال۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ’a‘ .

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
54a59b9f22b0b808 80d8427e548b7c23 abd873486e1f035d ce9cd697e8517503
3caa88e6d57bc35e fae0b5afd3145f31
```

الف-۶-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

بعد از فرایند لایه‌گذاری، بلوک ۱۶ کلمه‌ای تنها مشتق شده از رشته-داده همانند ذیل است:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ آمده است.

```
init cbbb9d5dc1059ed8 629a292a367cd507 9159015a3070dd17 152fecd8f70e5939
     67332667ffc00b31 8eb44a8768581511 db0c2e0d64f98fa7 47b5481dbefa4fa4
   0 470994ad30873f88 cbbb9d5dc1059ed8 629a292a367cd507 9159015a3070dd17
     bd03f724be6075f9 67332667ffc00b31 8eb44a8768581511 db0c2e0d64f98fa7
   1 2e91230306a12ae0 470994ad30873f88 cbbb9d5dc1059ed8 629a292a367cd507
     5e1b4e1695372b9e bd03f724be6075f9 67332667ffc00b31 8eb44a8768581511
   2 eebe5d379be707ad 2e91230306a12ae0 470994ad30873f88 cbbb9d5dc1059ed8
     54074a65aef34336 5e1b4e1695372b9e bd03f724be6075f9 67332667ffc00b31
   3 e308483153e15ad6 eebe5d379be707ad 2e91230306a12ae0 470994ad30873f88
     086c5b2d36a89178 54074a65aef34336 5e1b4e1695372b9e bd03f724be6075f9
   4 3a7a023c593d8479 e308483153e15ad6 eebe5d379be707ad 2e91230306a12ae0
     8aa1144850633794 086c5b2d36a89178 54074a65aef34336 5e1b4e1695372b9e
   5 333199a85f92b052 3a7a023c593d8479 e308483153e15ad6 eebe5d379be707ad
     7a6316f0ef047ce7 8aa1144850633794 086c5b2d36a89178 54074a65aef34336
   6 76f0741213dd2ef6 333199a85f92b052 3a7a023c593d8479 e308483153e15ad6
     74063cba385f0675 7a6316f0ef047ce7 8aa1144850633794 086c5b2d36a89178
   7 02f2a04d3aab1629 76f0741213dd2ef6 333199a85f92b052 3a7a023c593d8479
     1688b9bf14980fc0 74063cba385f0675 7a6316f0ef047ce7 8aa1144850633794
   8 73e5b2a1704a0349 02f2a04d3aab1629 76f0741213dd2ef6 333199a85f92b052
     fd00139f705907d0 1688b9bf14980fc0 74063cba385f0675 7a6316f0ef047ce7
   9 bf3f67ba12882648 73e5b2a1704a0349 02f2a04d3aab1629 76f0741213dd2ef6
     652e311d4f0a4257 fd00139f705907d0 1688b9bf14980fc0 74063cba385f0675
  10 33254508bb2ea48d bf3f67ba12882648 73e5b2a1704a0349 02f2a04d3aab1629
     9e18991c4f39f0ba 652e311d4f0a4257 fd00139f705907d0 1688b9bf14980fc0
  11 c1fdb2a0205ea0e5 33254508bb2ea48d bf3f67ba12882648 73e5b2a1704a0349
     04732e8bc4044582 9e18991c4f39f0ba 652e311d4f0a4257 fd00139f705907d0
  12 185f9ff038a50f39 c1fdb2a0205ea0e5 33254508bb2ea48d bf3f67ba12882648
     8b4acfc4d2b8afe6 04732e8bc4044582 9e18991c4f39f0ba 652e311d4f0a4257
  13 e5f06744c0d7563a 185f9ff038a50f39 c1fdb2a0205ea0e5 33254508bb2ea48d
     2fa93d1ce9523015 8b4acfc4d2b8afe6 04732e8bc4044582 9e18991c4f39f0ba
```

14 7e32dc0e9f414783 e5f06744c0d7563a 185f9ff038a50f39 c1fdb2a0205ea0e5
3a9950aaa5e75884 2fa93d1ce9523015 8b4acfc4d2b8afe6 04732e8bc4044582
15 1eab6159ae87ef6d 7e32dc0e9f414783 e5f06744c0d7563a 185f9ff038a50f39
153b895cfbc436c5 3a9950aaa5e75884 2fa93d1ce9523015 8b4acfc4d2b8afe6
16 33ef2cebbf1739aa 1eab6159ae87ef6d 7e32dc0e9f414783 e5f06744c0d7563a
9d1a64baf1d366aa 153b895cfbc436c5 3a9950aaa5e75884 2fa93d1ce9523015
17 7df1b65f1b87d6ca 33ef2cebbf1739aa 1eab6159ae87ef6d 7e32dc0e9f414783
5b6e369d36e8e181 9d1a64baf1d366aa 153b895cfbc436c5 3a9950aaa5e75884
18 63a24014a34bb0f6 7df1b65f1b87d6ca 33ef2cebbf1739aa 1eab6159ae87ef6d
e13e610eae680d85 5b6e369d36e8e181 9d1a64baf1d366aa 153b895cfbc436c5
19 f1aabd313309509b 63a24014a34bb0f6 7df1b65f1b87d6ca 33ef2cebbf1739aa
674385f0d87db94f e13e610eae680d85 5b6e369d36e8e181 9d1a64baf1d366aa
20 9ba737ae88a72c64 f1aabd313309509b 63a24014a34bb0f6 7df1b65f1b87d6ca
3fc2614c43906c0f 674385f0d87db94f e13e610eae680d85 5b6e369d36e8e181
21 042c2dc9a5bf558a 9ba737ae88a72c64 f1aabd313309509b 63a24014a34bb0f6
19316bebc88e01f2 3fc2614c43906c0f 674385f0d87db94f e13e610eae680d85
22 7799c75acc748c0f 042c2dc9a5bf558a 9ba737ae88a72c64 f1aabd313309509b
a7bbd65bf64f58c8 19316bebc88e01f2 3fc2614c43906c0f 674385f0d87db94f
23 ccf99a80f92bf002 7799c75acc748c0f 042c2dc9a5bf558a 9ba737ae88a72c64
e52a24fae4e8fc9b a7bbd65bf64f58c8 19316bebc88e01f2 3fc2614c43906c0f
24 ae993474363efe68 ccf99a80f92bf002 7799c75acc748c0f 042c2dc9a5bf558a
587f308d58681928 e52a24fae4e8fc9b a7bbd65bf64f58c8 19316bebc88e01f2
25 335063d1a2aec92f ae993474363efe68 ccf99a80f92bf002 7799c75acc748c0f
c2d6d65e38c6ea79 587f308d58681928 e52a24fae4e8fc9b a7bbd65bf64f58c8
26 53a78b0cca01ba37 335063d1a2aec92f ae993474363efe68 ccf99a80f92bf002
3b65a26c3c92c8f3 c2d6d65e38c6ea79 587f308d58681928 e52a24fae4e8fc9b
27 ab7ffa529f622930 53a78b0cca01ba37 335063d1a2aec92f ae993474363efe68
b9d8a2f2762901ea 3b65a26c3c92c8f3 c2d6d65e38c6ea79 587f308d58681928
28 e428bb43afe3d63e ab7ffa529f622930 53a78b0cca01ba37 335063d1a2aec92f
6a8527525f898726 b9d8a2f2762901ea 3b65a26c3c92c8f3 c2d6d65e38c6ea79

29 bbed541a5128088c e428bb43afe3d63e ab7ffa529f622930 53a78b0cca01ba37
7973aadbde294be9 6a8527525f898726 b9d8a2f2762901ea 3b65a26c3c92c8f3
30 4c5c38df7ec8baf4 bbed541a5128088c e428bb43afe3d63e ab7ffa529f622930
422ceea0200e9ee4 7973aadbde294be9 6a8527525f898726 b9d8a2f2762901ea
31 4ba456ec244033ed 4c5c38df7ec8baf4 bbed541a5128088c e428bb43afe3d63e
7cf40857056d86b0 422ceea0200e9ee4 7973aadbde294be9 6a8527525f898726
32 aa4a6ab2ac5f5dd8 4ba456ec244033ed 4c5c38df7ec8baf4 bbed541a5128088c
ad2b1ecfb5bfc556 7cf40857056d86b0 422ceea0200e9ee4 7973aadbde294be9
33 9cb941f2ced774b3 aa4a6ab2ac5f5dd8 4ba456ec244033ed 4c5c38df7ec8baf4
029f66c7b4569bf0 ad2b1ecfb5bfc556 7cf40857056d86b0 422ceea0200e9ee4
34 39265f358594de27 9cb941f2ced774b3 aa4a6ab2ac5f5dd8 4ba456ec244033ed
3f7b1c260c82e54f 029f66c7b4569bf0 ad2b1ecfb5bfc556 7cf40857056d86b0
35 09cca487d39b02a1 39265f358594de27 9cb941f2ced774b3 aa4a6ab2ac5f5dd8
4a22b37b58a5b1b0 3f7b1c260c82e54f 029f66c7b4569bf0 ad2b1ecfb5bfc556
36 d48d97ce438cf4f0 09cca487d39b02a1 39265f358594de27 9cb941f2ced774b3
a239e00b8baa0410 4a22b37b58a5b1b0 3f7b1c260c82e54f 029f66c7b4569bf0
37 d6f41e25a8b634d6 d48d97ce438cf4f0 09cca487d39b02a1 39265f358594de27
25755cb8179dd0b0 a239e00b8baa0410 4a22b37b58a5b1b0 3f7b1c260c82e54f
38 54078334358573b4 d6f41e25a8b634d6 d48d97ce438cf4f0 09cca487d39b02a1
0e419fb0802b0efc 25755cb8179dd0b0 a239e00b8baa0410 4a22b37b58a5b1b0
39 db24f9a03f4fff6b 54078334358573b4 d6f41e25a8b634d6 d48d97ce438cf4f0
d30e99b4b394b090 0e419fb0802b0efc 25755cb8179dd0b0 a239e00b8baa0410
40 3604c53a845efc37 db24f9a03f4fff6b 54078334358573b4 d6f41e25a8b634d6
791b2b4af7338b99 d30e99b4b394b090 0e419fb0802b0efc 25755cb8179dd0b0
41 f41b1c0eee89bdc6 3604c53a845efc37 db24f9a03f4fff6b 54078334358573b4
e319b77d9e4e87f9 791b2b4af7338b99 d30e99b4b394b090 0e419fb0802b0efc
42 36644ae374632e3a f41b1c0eee89bdc6 3604c53a845efc37 db24f9a03f4fff6b
458250878a3972b2 e319b77d9e4e87f9 791b2b4af7338b99 d30e99b4b394b090
43 88806f6ae9fcd65b 36644ae374632e3a f41b1c0eee89bdc6 3604c53a845efc37
cfde2e6ea54fa576 458250878a3972b2 e319b77d9e4e87f9 791b2b4af7338b99

44 51dcaa36995c301d 88806f6ae9fcd65b 36644ae374632e3a f41b1c0eee89bdc6
e37f778353998050 cfde2e6ea54fa576 458250878a3972b2 e319b77d9e4e87f9
45 ef5e3885a2f238df 51dcaa36995c301d 88806f6ae9fcd65b 36644ae374632e3a
740e347f24e18fda e37f778353998050 cfde2e6ea54fa576 458250878a3972b2
46 eb3753f4283f4818 ef5e3885a2f238df 51dcaa36995c301d 88806f6ae9fcd65b
0ae48cf840bb8be9 740e347f24e18fda e37f778353998050 cfde2e6ea54fa576
47 a6998d63a5d09e04 eb3753f4283f4818 ef5e3885a2f238df 51dcaa36995c301d
e21095012ee0b72a 0ae48cf840bb8be9 740e347f24e18fda e37f778353998050
48 d3698fb64df175b0 a6998d63a5d09e04 eb3753f4283f4818 ef5e3885a2f238df
c2f0b90ffce80739 e21095012ee0b72a 0ae48cf840bb8be9 740e347f24e18fda
49 317a3b295b991914 d3698fb64df175b0 a6998d63a5d09e04 eb3753f4283f4818
1cadff2e6cb5aa4d c2f0b90ffce80739 e21095012ee0b72a 0ae48cf840bb8be9
50 0941da08148ba463 317a3b295b991914 d3698fb64df175b0 a6998d63a5d09e04
833eb9a4bb5a073e 1cadff2e6cb5aa4d c2f0b90ffce80739 e21095012ee0b72a
51 494ac238d68c3d0b 0941da08148ba463 317a3b295b991914 d3698fb64df175b0
80c8fc138e645028 833eb9a4bb5a073e 1cadff2e6cb5aa4d c2f0b90ffce80739
52 c87e9168db9e97de 494ac238d68c3d0b 0941da08148ba463 317a3b295b991914
65cf7f6a829aca04 80c8fc138e645028 833eb9a4bb5a073e 1cadff2e6cb5aa4d
53 edb4448879391dbb c87e9168db9e97de 494ac238d68c3d0b 0941da08148ba463
7729c85475dd318f 65cf7f6a829aca04 80c8fc138e645028 833eb9a4bb5a073e
54 073775c2456dc7db edb4448879391dbb c87e9168db9e97de 494ac238d68c3d0b
a9cca0b6266b1d77 7729c85475dd318f 65cf7f6a829aca04 80c8fc138e645028
55 54de8857b24afaf7 073775c2456dc7db edb4448879391dbb c87e9168db9e97de
8de51cff2ae4b068 a9cca0b6266b1d77 7729c85475dd318f 65cf7f6a829aca04
56 8a9cdd80f7f09c05 54de8857b24afaf7 073775c2456dc7db edb4448879391dbb
a60ba5e9ebaeb96a 8de51cff2ae4b068 a9cca0b6266b1d77 7729c85475dd318f
57 3eeb22a7524d8d7f 8a9cdd80f7f09c05 54de8857b24afaf7 073775c2456dc7db
e2e6830b139df58f a60ba5e9ebaeb96a 8de51cff2ae4b068 a9cca0b6266b1d77
58 0ed77c9cde8883d3 3eeb22a7524d8d7f 8a9cdd80f7f09c05 54de8857b24afaf7
38413a2052387a9e e2e6830b139df58f a60ba5e9ebaeb96a 8de51cff2ae4b068

59 e64e4135f9d30dbc 0ed77c9cde8883d3 3eeb22a7524d8d7f 8a9cdd80f7f09c05
45b640454c75c349 38413a2052387a9e e2e6830b139df58f a60ba5e9ebaeb96a
60 1ca93a293d544328 e64e4135f9d30dbc 0ed77c9cde8883d3 3eeb22a7524d8d7f
efbef83a35c0319e 45b640454c75c349 38413a2052387a9e e2e6830b139df58f
61 3dc764f89e54043a 1ca93a293d544328 e64e4135f9d30dbc 0ed77c9cde8883d3
a57784945550cf94 efbef83a35c0319e 45b640454c75c349 38413a2052387a9e
62 56fb5883f1c87a05 3dc764f89e54043a 1ca93a293d544328 e64e4135f9d30dbc
f5198a41eb80e022 a57784945550cf94 efbef83a35c0319e 45b640454c75c349
63 24a1124262a331c7 56fb5883f1c87a05 3dc764f89e54043a 1ca93a293d544328
06edacae6e7b54ad f5198a41eb80e022 a57784945550cf94 efbef83a35c0319e
64 eb85d19201c89694 24a1124262a331c7 56fb5883f1c87a05 3dc764f89e54043a
9ced24983eec8723 06edacae6e7b54ad f5198a41eb80e022 a57784945550cf94
65 cc981ab3a59c1db4 eb85d19201c89694 24a1124262a331c7 56fb5883f1c87a05
eac5516336bc8882 9ced24983eec8723 06edacae6e7b54ad f5198a41eb80e022
66 ceef5d997e148b44 cc981ab3a59c1db4 eb85d19201c89694 24a1124262a331c7
617bbf70bb165212 eac5516336bc8882 9ced24983eec8723 06edacae6e7b54ad
67 689edf608a8e3f14 ceef5d997e148b44 cc981ab3a59c1db4 eb85d19201c89694
3280d88472c100fd 617bbf70bb165212 eac5516336bc8882 9ced24983eec8723
68 1e6e0255ab88079f 689edf608a8e3f14 ceef5d997e148b44 cc981ab3a59c1db4
f2001138439902b1 3280d88472c100fd 617bbf70bb165212 eac5516336bc8882
69 8c5d3b7fdad66e70 1e6e0255ab88079f 689edf608a8e3f14 ceef5d997e148b44
90d18ec8b69f0345 f2001138439902b1 3280d88472c100fd 617bbf70bb165212
70 32e5ed8655871e9b 8c5d3b7fdad66e70 1e6e0255ab88079f 689edf608a8e3f14
51105f6241313777 90d18ec8b69f0345 f2001138439902b1 3280d88472c100fd
71 bcd5061679be7336 32e5ed8655871e9b 8c5d3b7fdad66e70 1e6e0255ab88079f
454b99f654443ad0 51105f6241313777 90d18ec8b69f0345 f2001138439902b1
72 e7d913b6678e78ef bcd5061679be7336 32e5ed8655871e9b 8c5d3b7fdad66e70
1ff613b5aa63776e 454b99f654443ad0 51105f6241313777 90d18ec8b69f0345
73 e6b8cb8dfa3475ab e7d913b6678e78ef bcd5061679be7336 32e5ed8655871e9b
2e75f34303d39bb0 1ff613b5aa63776e 454b99f654443ad0 51105f6241313777

```
74 fdd4a30e168c4ae5 e6b8cb8dfa3475ab e7d913b6678e78ef bcd5061679be7336
   83a35dbe2a64fc26 2e75f34303d39bb0 1ff613b5aa63776e 454b99f654443ad0
75 12aeb6268dfa3e14 fdd4a30e168c4ae5 e6b8cb8dfa3475ab e7d913b6678e78ef
   f660943b276786f7 83a35dbe2a64fc26 2e75f34303d39bb0 1ff613b5aa63776e
76 055b73814cf102b4 12aeb6268dfa3e14 fdd4a30e168c4ae5 e6b8cb8dfa3475ab
   c4b149710f5d6a71 f660943b276786f7 83a35dbe2a64fc26 2e75f34303d39bb0
77 95d33150de6df44c 055b73814cf102b4 12aeb6268dfa3e14 fdd4a30e168c4ae5
   c7f7bff08ebf0d30 c4b149710f5d6a71 f660943b276786f7 83a35dbe2a64fc26
78 5306143f64497b00 95d33150de6df44c 055b73814cf102b4 12aeb6268dfa3e14
   ca06a219cc701096 c7f7bff08ebf0d30 c4b149710f5d6a71 f660943b276786f7
79 ff44d7e1849dbfb3 5306143f64497b00 95d33150de6df44c 055b73814cf102b4
   1952e0c3a227c0f2 ca06a219cc701096 c7f7bff08ebf0d30 c4b149710f5d6a71
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

X_0 = cbbb9d5dc1059ed8 ⊎ ff44d7e1849dbfb3 = cb00753f45a35e8b

X_1 = 629a292a367cd507 ⊎ 5306143f64497b00 = b5a03d699ac65007

X_2 = 9159015a3070dd17 ⊎ 95d33150de6df44c = 272c32ab0eded163

X_3 = 152fecd8f70e5939 ⊎ 055b73814cf102b4 = 1a8b605a43ff5bed

X_4 = 67332667ffc00b31 ⊎ 1952e0c3a227c0f2 = 8086072ba1e7cc23

X_5 = 8eb44a8768581511 ⊎ ca06a219cc701096 = 58baeca134c825a7

X_6 = db0c2e0d64f98fa7 ⊎ c7f7bff08ebf0d30 = a303edfdf3b89cd7

X_7 = 47b5481dbefa4fa4 ⊎ c4b149710f5d6a71 = 0c66918ece57ba15

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
cb00753f45a35e8b b5a03d699ac65007 272c32ab0eded163 1a8b605a43ff5bed
8086072ba1e7cc23 58baeca134c825a7
```

الف-۶-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´
کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
473ed35167ec1f5d 8e550368a3db39be 54639f828868e945 4c239fc8b52e3c61
dbd0d8b4de1390c2 56dcbb5d5fd99cd5
```

الف-۶-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶ بایتی است، معادل کد ASCII
´abcdefghijklmnopqrstuvwxyz´
کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
feb67349df3db6f5 924815d6c3dc133f 091809213731fe5c 7b5f4999e463479f
f2877f5f2936fa63 bb43784b12f3ebb4
```

الف-۶-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
1761336e3f7cbfe5 1deb137f026f89e0 1a448e3b1fafa640 39c1464ee8732f11
a5341a6f41e0c202 294736ed64db1a84
```

الف-۶-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار
‘1234567890’
کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
b12932b0627d1c06 0942f54477641556 55bd4da0c9afa6dd 9b9ef53129af1b8f
b0195996d2de9ca0 df9d821ffee67026
```

الف-۶-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
3391fdddfc8dc739 3707a65b1b470939 7cf8b1d162af05ab fe8f450de5f36bc6
b0455a8520bc4e6f 5fe95b1fe3c8452b
```

الف-۶-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای $۱۰^۶$ بار تکرار می‌شود.

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
9d0e1809716474cb 086e834e310a4a1c ed149e9c00f24852 7972cec5704c2a5b
07b8b3dc38ecc4eb ae97ddd87f3d8985
```

الف-۶-۱۰ مثال ۱۰

در این مثال رشته-داده شامل یک رشته‌ای ۱۱۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn
hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu’

(بدون سرخط بعد از اولین n)

بعد از فرایند لایه‌گذاری، دو بلوک ۱۶ کلمه‌ای مشتق شده از رشته-داده همانند ذیل است:

```
61626364 65666768 62636465 66676869 63646566 6768696a 64656667 68696a6b
65666768 696a6b6c 66676869 6a6b6c6d 6768696a 6b6c6d6e 68696a6b 6c6d6e6f
696a6b6c 6d6e6f70 6a6b6c6d 6e6f7071 6b6c6d6e 6f707172 6c6d6e6f 70717273
6d6e6f70 71727374 6e6f7071 72737475 80000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000380
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ که در طول پردازش اولین بلوک به‌دست آمده، آورده شده است.

```
init cbbb9d5dc1059ed8 629a292a367cd507 9159015a3070dd17 152fecd8f70e5939
     67332667ffc00b31 8eb44a8768581511 db0c2e0d64f98fa7 47b5481dbefa4fa4
   0 4709949195eda6f0 cbbb9d5dc1059ed8 629a292a367cd507 9159015a3070dd17
     bd03f70923c6dd61 67332667ffc00b31 8eb44a8768581511 db0c2e0d64f98fa7
   1 78d3f8bc03a38303 4709949195eda6f0 cbbb9d5dc1059ed8 629a292a367cd507
     ae067f071cd18a36 bd03f70923c6dd61 67332667ffc00b31 8eb44a8768581511
```

2 ed59d30beff95306 78d3f8bc03a38303 4709949195eda6f0 cbbb9d5dc1059ed8
c180c7a74ed5cf1f ae067f071cd18a36 bd03f70923c6dd61 67332667ffc00b31
3 8e7fe2aba3168f2b ed59d30beff95306 78d3f8bc03a38303 4709949195eda6f0
d92d19667920b327 c180c7a74ed5cf1f ae067f071cd18a36 bd03f70923c6dd61
4 1174f9b374a9263a 8e7fe2aba3168f2b ed59d30beff95306 78d3f8bc03a38303
dd371f2d13661c52 d92d19667920b327 c180c7a74ed5cf1f ae067f071cd18a36
5 27aaafb7fbef806b 1174f9b374a9263a 8e7fe2aba3168f2b ed59d30beff95306
21af3c6430a9af9c dd371f2d13661c52 d92d19667920b327 c180c7a74ed5cf1f
6 b352d03a0bd34d65 27aaafb7fbef806b 1174f9b374a9263a 8e7fe2aba3168f2b
69397de9a30e1473 21af3c6430a9af9c dd371f2d13661c52 d92d19667920b327
7 412db7f990563d7c b352d03a0bd34d65 27aaafb7fbef806b 1174f9b374a9263a
5062fd5924e2b62e 69397de9a30e1473 21af3c6430a9af9c dd371f2d13661c52
8 0f79040546e6edf7 412db7f990563d7c b352d03a0bd34d65 27aaafb7fbef806b
6b6c511b25a6bdbc 5062fd5924e2b62e 69397de9a30e1473 21af3c6430a9af9c
9 ebf02410f67b8ee7 0f79040546e6edf7 412db7f990563d7c b352d03a0bd34d65
dac695b91543ae80 6b6c511b25a6bdbc 5062fd5924e2b62e 69397de9a30e1473
10 97aa05d89b8dbe6d ebf02410f67b8ee7 0f79040546e6edf7 412db7f990563d7c
83b8b72646c0b598 dac695b91543ae80 6b6c511b25a6bdbc 5062fd5924e2b62e
11 23d0a36b692118eb 97aa05d89b8dbe6d ebf02410f67b8ee7 0f79040546e6edf7
a5f6c5155e221e8c 83b8b72646c0b598 dac695b91543ae80 6b6c511b25a6bdbc
12 e1041368d2fca1a2 23d0a36b692118eb 97aa05d89b8dbe6d ebf02410f67b8ee7
ae01675bfb003180 a5f6c5155e221e8c 83b8b72646c0b598 dac695b91543ae80
13 45bd6f69efec540d e1041368d2fca1a2 23d0a36b692118eb 97aa05d89b8dbe6d
c35cc50c1cf7ef98 ae01675bfb003180 a5f6c5155e221e8c 83b8b72646c0b598
14 c237fa23abb9bc16 45bd6f69efec540d e1041368d2fca1a2 23d0a36b692118eb
a16c4f134b28923e c35cc50c1cf7ef98 ae01675bfb003180 a5f6c5155e221e8c
15 b4092df1c0f81853 c237fa23abb9bc16 45bd6f69efec540d e1041368d2fca1a2
008178e17fa649f2 a16c4f134b28923e c35cc50c1cf7ef98 ae01675bfb003180
16 21e5c91d11809c13 b4092df1c0f81853 c237fa23abb9bc16 45bd6f69efec540d
a26dfa04ed8c9b63 008178e17fa649f2 a16c4f134b28923e c35cc50c1cf7ef98

17 2c957137cd4304a5 21e5c91d11809c13 b4092df1c0f81853 c237fa23abb9bc16
6be210614b10949b a26dfa04ed8c9b63 008178e17fa649f2 a16c4f134b28923e
18 2180e61afe322bc7 2c957137cd4304a5 21e5c91d11809c13 b4092df1c0f81853
76396996200065f7 6be210614b10949b a26dfa04ed8c9b63 008178e17fa649f2
19 f2911c11c96e5ff5 2180e61afe322bc7 2c957137cd4304a5 21e5c91d11809c13
1bc2160f4f3711dc 76396996200065f7 6be210614b10949b a26dfa04ed8c9b63
20 5eab10b19a5143a8 f2911c11c96e5ff5 2180e61afe322bc7 2c957137cd4304a5
98d2b19d201f2bb6 1bc2160f4f3711dc 76396996200065f7 6be210614b10949b
21 29c5348d87cd5590 5eab10b19a5143a8 f2911c11c96e5ff5 2180e61afe322bc7
4324c8caccf7753c 98d2b19d201f2bb6 1bc2160f4f3711dc 76396996200065f7
22 33c6b4a0166b7c9c 29c5348d87cd5590 5eab10b19a5143a8 f2911c11c96e5ff5
d49cef5bd2dec121 4324c8caccf7753c 98d2b19d201f2bb6 1bc2160f4f3711dc
23 1db4ee606d2a7a96 33c6b4a0166b7c9c 29c5348d87cd5590 5eab10b19a5143a8
b17d15b397521ab3 d49cef5bd2dec121 4324c8caccf7753c 98d2b19d201f2bb6
24 5cef5b2f00142660 1db4ee606d2a7a96 33c6b4a0166b7c9c 29c5348d87cd5590
789e540f22e13932 b17d15b397521ab3 d49cef5bd2dec121 4324c8caccf7753c
25 ff74f4a162435903 5cef5b2f00142660 1db4ee606d2a7a96 33c6b4a0166b7c9c
6c0be33dcc6e7572 789e540f22e13932 b17d15b397521ab3 d49cef5bd2dec121
26 41740b736e9676a9 ff74f4a162435903 5cef5b2f00142660 1db4ee606d2a7a96
d8e401251592da6c 6c0be33dcc6e7572 789e540f22e13932 b17d15b397521ab3
27 931059fe9279ff1d 41740b736e9676a9 ff74f4a162435903 5cef5b2f00142660
7f31116887eea596 d8e401251592da6c 6c0be33dcc6e7572 789e540f22e13932
28 356d08d982e2ead4 931059fe9279ff1d 41740b736e9676a9 ff74f4a162435903
40c28c34b1bbe906 7f31116887eea596 d8e401251592da6c 6c0be33dcc6e7572
29 89dc825e7235c74b 356d08d982e2ead4 931059fe9279ff1d 41740b736e9676a9
7a499ae05da50bf2 40c28c34b1bbe906 7f31116887eea596 d8e401251592da6c
30 97901f333e662fdc 89dc825e7235c74b 356d08d982e2ead4 931059fe9279ff1d
4472b2e331ddfab4 7a499ae05da50bf2 40c28c34b1bbe906 7f31116887eea596
31 69c8f40eb38b6022 97901f333e662fdc 89dc825e7235c74b 356d08d982e2ead4
177589502dd39aa2 4472b2e331ddfab4 7a499ae05da50bf2 40c28c34b1bbe906

32 4920943ffe52b207 69c8f40eb38b6022 97901f333e662fdc 89dc825e7235c74b
6b813a0d0cdf4991 177589502dd39aa2 4472b2e331ddfab4 7a499ae05da50bf2
33 b4cb0df332d108ab 4920943ffe52b207 69c8f40eb38b6022 97901f333e662fdc
8fe3d28097f18618 6b813a0d0cdf4991 177589502dd39aa2 4472b2e331ddfab4
34 e7748fbf744a5240 b4cb0df332d108ab 4920943ffe52b207 69c8f40eb38b6022
0d7ab03208f1d7a5 8fe3d28097f18618 6b813a0d0cdf4991 177589502dd39aa2
35 7416ca18d9e265e0 e7748fbf744a5240 b4cb0df332d108ab 4920943ffe52b207
11200c2d47c082f8 0d7ab03208f1d7a5 8fe3d28097f18618 6b813a0d0cdf4991
36 75476f5456e82f9c 7416ca18d9e265e0 e7748fbf744a5240 b4cb0df332d108ab
3024702447f76224 11200c2d47c082f8 0d7ab03208f1d7a5 8fe3d28097f18618
37 f638a568b53a2f8f 75476f5456e82f9c 7416ca18d9e265e0 e7748fbf744a5240
6217c1c02153302c 3024702447f76224 11200c2d47c082f8 0d7ab03208f1d7a5
38 c418f6f90602c79a f638a568b53a2f8f 75476f5456e82f9c 7416ca18d9e265e0
87f0901c227adbb3 6217c1c02153302c 3024702447f76224 11200c2d47c082f8
39 4f1f4f21df3dcf43 c418f6f90602c79a f638a568b53a2f8f 75476f5456e82f9c
fb7c63fcddf4a1c2 87f0901c227adbb3 6217c1c02153302c 3024702447f76224
40 13eb82e4b98d0e67 4f1f4f21df3dcf43 c418f6f90602c79a f638a568b53a2f8f
fb6c0e54d48d4f2d fb7c63fcddf4a1c2 87f0901c227adbb3 6217c1c02153302c
41 820e75046567bace 13eb82e4b98d0e67 4f1f4f21df3dcf43 c418f6f90602c79a
b16a9397472f0123 fb6c0e54d48d4f2d fb7c63fcddf4a1c2 87f0901c227adbb3
42 741fa5dc290dd02c 820e75046567bace 13eb82e4b98d0e67 4f1f4f21df3dcf43
ed40c88214823792 b16a9397472f0123 fb6c0e54d48d4f2d fb7c63fcddf4a1c2
43 a4809bf6da6aa8bd 741fa5dc290dd02c 820e75046567bace 13eb82e4b98d0e67
bec3d7e88c855194 ed40c88214823792 b16a9397472f0123 fb6c0e54d48d4f2d
44 d70b1aa4c800979c a4809bf6da6aa8bd 741fa5dc290dd02c 820e75046567bace
4962f310bdbd54b0 bec3d7e88c855194 ed40c88214823792 b16a9397472f0123
45 9a195492cfdb4745 d70b1aa4c800979c a4809bf6da6aa8bd 741fa5dc290dd02c
2c82d09cf05cf687 4962f310bdbd54b0 bec3d7e88c855194 ed40c88214823792
46 b7e68364f07f017e 9a195492cfdb4745 d70b1aa4c800979c a4809bf6da6aa8bd
2a1ffb84031b1b6c 2c82d09cf05cf687 4962f310bdbd54b0 bec3d7e88c855194

47 0e574b8e0b35e452 b7e68364f07f017e 9a195492cfdb4745 d70b1aa4c800979c
29bdab29ee472a23 2a1ffb84031b1b6c 2c82d09cf05cf687 4962f310bdbd54b0
48 c176009cf82fa842 0e574b8e0b35e452 b7e68364f07f017e 9a195492cfdb4745
cca47fbe31b335f4 29bdab29ee472a23 2a1ffb84031b1b6c 2c82d09cf05cf687
49 5d4f78c7a9bdbed2 c176009cf82fa842 0e574b8e0b35e452 b7e68364f07f017e
eaf198615e99ffdc cca47fbe31b335f4 29bdab29ee472a23 2a1ffb84031b1b6c
50 51ab3be828d8d13c 5d4f78c7a9bdbed2 c176009cf82fa842 0e574b8e0b35e452
bd527cd188fb59ae eaf198615e99ffdc cca47fbe31b335f4 29bdab29ee472a23
51 4d639ef80d0f6d3e 51ab3be828d8d13c 5d4f78c7a9bdbed2 c176009cf82fa842
b2611b90f90d732f bd527cd188fb59ae eaf198615e99ffdc cca47fbe31b335f4
52 bba9c9efe0fbc6c8 4d639ef80d0f6d3e 51ab3be828d8d13c 5d4f78c7a9bdbed2
fc0579337591a2c9 b2611b90f90d732f bd527cd188fb59ae eaf198615e99ffdc
53 3405d7cad2e8a689 bba9c9efe0fbc6c8 4d639ef80d0f6d3e 51ab3be828d8d13c
0f6649f64ec8e109 fc0579337591a2c9 b2611b90f90d732f bd527cd188fb59ae
54 ea54d908505798b3 3405d7cad2e8a689 bba9c9efe0fbc6c8 4d639ef80d0f6d3e
ef48a48999108077 0f6649f64ec8e109 fc0579337591a2c9 b2611b90f90d732f
55 be31d1c0ccc143bc ea54d908505798b3 3405d7cad2e8a689 bba9c9efe0fbc6c8
4fc2d4cad0c91afc ef48a48999108077 0f6649f64ec8e109 fc0579337591a2c9
56 285a76d23f6a0073 be31d1c0ccc143bc ea54d908505798b3 3405d7cad2e8a689
a730855599b738a3 4fc2d4cad0c91afc ef48a48999108077 0f6649f64ec8e109
57 a714ceff14bebc24 285a76d23f6a0073 be31d1c0ccc143bc ea54d908505798b3
53c581dae1831d80 a730855599b738a3 4fc2d4cad0c91afc ef48a48999108077
58 697ca14913a50a26 a714ceff14bebc24 285a76d23f6a0073 be31d1c0ccc143bc
34d39344354aacd2 53c581dae1831d80 a730855599b738a3 4fc2d4cad0c91afc
59 3a38fa3775d7007c 697ca14913a50a26 a714ceff14bebc24 285a76d23f6a0073
e26f3a21e9a27691 34d39344354aacd2 53c581dae1831d80 a730855599b738a3
60 44ea14d8e450c844 3a38fa3775d7007c 697ca14913a50a26 a714ceff14bebc24
5319374fb88dd485 e26f3a21e9a27691 34d39344354aacd2 53c581dae1831d80
61 0928b75c925f91e2 44ea14d8e450c844 3a38fa3775d7007c 697ca14913a50a26
79f4be3c5a372911 5319374fb88dd485 e26f3a21e9a27691 34d39344354aacd2

62 6db5469fa19c0e27 0928b75c925f91e2 44ea14d8e450c844 3a38fa3775d7007c
16beec0fec168e79 79f4be3c5a372911 5319374fb88dd485 e26f3a21e9a27691
63 384e3159898a7362 6db5469fa19c0e27 0928b75c925f91e2 44ea14d8e450c844
55fa3ad1102298a8 16beec0fec168e79 79f4be3c5a372911 5319374fb88dd485
64 483c64d3fdebf828 384e3159898a7362 6db5469fa19c0e27 0928b75c925f91e2
1a238431921ea75e 55fa3ad1102298a8 16beec0fec168e79 79f4be3c5a372911
65 c9464988a1939bcf 483c64d3fdebf828 384e3159898a7362 6db5469fa19c0e27
e3f3f08ac90f86cd 1a238431921ea75e 55fa3ad1102298a8 16beec0fec168e79
66 98bc93bca795059c c9464988a1939bcf 483c64d3fdebf828 384e3159898a7362
9e04fb49a5fd91de e3f3f08ac90f86cd 1a238431921ea75e 55fa3ad1102298a8
67 b6fc101ad1d74e20 98bc93bca795059c c9464988a1939bcf 483c64d3fdebf828
fd13cd3620f6c1f4 9e04fb49a5fd91de e3f3f08ac90f86cd 1a238431921ea75e
68 fac26e6e4da4705d b6fc101ad1d74e20 98bc93bca795059c c9464988a1939bcf
0d60228aa6e55b6e fd13cd3620f6c1f4 9e04fb49a5fd91de e3f3f08ac90f86cd
69 2a630c58cc27fcaa fac26e6e4da4705d b6fc101ad1d74e20 98bc93bca795059c
a2f7f27a3ec25aba 0d60228aa6e55b6e fd13cd3620f6c1f4 9e04fb49a5fd91de
70 159a02d4faee11b4 2a630c58cc27fcaa fac26e6e4da4705d b6fc101ad1d74e20
b2860fc55bdedaa6 a2f7f27a3ec25aba 0d60228aa6e55b6e fd13cd3620f6c1f4
71 9d38bdb9df22b557 159a02d4faee11b4 2a630c58cc27fcaa fac26e6e4da4705d
dfc37c68af65f8bc b2860fc55bdedaa6 a2f7f27a3ec25aba 0d60228aa6e55b6e
72 d42c3a57cfa78513 9d38bdb9df22b557 159a02d4faee11b4 2a630c58cc27fcaa
bb56dea6a325ba32 dfc37c68af65f8bc b2860fc55bdedaa6 a2f7f27a3ec25aba
73 abab4b0ca75a17c7 d42c3a57cfa78513 9d38bdb9df22b557 159a02d4faee11b4
9ac71d1c037a8bbd bb56dea6a325ba32 dfc37c68af65f8bc b2860fc55bdedaa6
74 500f7b61186f6c2e abab4b0ca75a17c7 d42c3a57cfa78513 9d38bdb9df22b557
8347f5736531b3ec 9ac71d1c037a8bbd bb56dea6a325ba32 dfc37c68af65f8bc
75 4abe0af6a67db2fe 500f7b61186f6c2e abab4b0ca75a17c7 d42c3a57cfa78513
14e986342ddced0f 8347f5736531b3ec 9ac71d1c037a8bbd bb56dea6a325ba32
76 e1053fc85f9e56be 4abe0af6a67db2fe 500f7b61186f6c2e abab4b0ca75a17c7
4779767cc2ec5321 14e986342ddced0f 8347f5736531b3ec 9ac71d1c037a8bbd

```
77 7001201948fb3d71 e1053fc85f9e56be 4abe0af6a67db2fe 500f7b61186f6c2e
   5cdf6c58fc052572 4779767cc2ec5321 14e986342ddced0f 8347f5736531b3ec
78 88146da76ff6f23a 7001201948fb3d71 e1053fc85f9e56be 4abe0af6a67db2fe
   8901cffe7a74db98 5cdf6c58fc052572 4779767cc2ec5321 14e986342ddced0f
79 5ec3802b9ecfef33 88146da76ff6f23a 7001201948fb3d71 e1053fc85f9e56be
   5f2eead69efb4233 8901cffe7a74db98 5cdf6c58fc052572 4779767cc2ec5321
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را در فرایند اولین بلوک نمایش می‌دهند:

X_0 = cbbb9d5dc1059ed8 ⊎ 5ec3802b9ecfef33 = 2a7f1d895fd58e0b

X_1 = 629a292a367cd507 ⊎ 88146da76ff6f23a = eaae96d1a673c741

X_2 = 9159015a3070dd17 ⊎ 7001201948fb3d71 = 015a2173796c1a88

X_3 = 152fecd8f70e5939 ⊎ e1053fc85f9e56be = f6352ca156acaff7

X_4 = 67332667ffc00b31 ⊎ 5f2eead69efb4233 = c662113e9ebb4d64

X_5 = 8eb44a8768581511 ⊎ 8901cffe7a74db98 = 17b61a85e2ccf0a9

X_6 = db0c2e0d64f98fa7 ⊎ 5cdf6c58fc052572 = 37eb9a6660feb519

X_7 = 47b5481dbefa4fa4 ⊎ 4779767cc2ec5321 = 8f2ebe9a81e6a2c5

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ که در طول پردازش دومین بلوک به‌دست آمده، آورده شده است.

```
init 2a7f1d895fd58e0b eaae96d1a673c741 015a2173796c1a88 f6352ca156acaff7
     c662113e9ebb4d64 17b61a85e2ccf0a9 37eb9a6660feb519 8f2ebe9a81e6a2c5
   0 657a3c2ca9639d40 2a7f1d895fd58e0b eaae96d1a673c741 015a2173796c1a88
     791f2ad0055fdd62 c662113e9ebb4d64 17b61a85e2ccf0a9 37eb9a6660feb519
   1 2a4ad5d9b9fd6d86 657a3c2ca9639d40 2a7f1d895fd58e0b eaae96d1a673c741
     dbf2e656b5be3f14 791f2ad0055fdd62 c662113e9ebb4d64 17b61a85e2ccf0a9
   2 f0aa6758653d1664 2a4ad5d9b9fd6d86 657a3c2ca9639d40 2a7f1d895fd58e0b
     6e0466c82f4fd35d dbf2e656b5be3f14 791f2ad0055fdd62 c662113e9ebb4d64
```

 3 43a76f011a73d317 f0aa6758653d1664 2a4ad5d9b9fd6d86 657a3c2ca9639d40
   1367bd36d15e8b40 6e0466c82f4fd35d dbf2e656b5be3f14 791f2ad0055fdd62
 4 d802c2dfd7cc48f6 43a76f011a73d317 f0aa6758653d1664 2a4ad5d9b9fd6d86
   f73d759b839a2a21 1367bd36d15e8b40 6e0466c82f4fd35d dbf2e656b5be3f14
 5 481208e5e8314602 d802c2dfd7cc48f6 43a76f011a73d317 f0aa6758653d1664
   6b2271a46f14c843 f73d759b839a2a21 1367bd36d15e8b40 6e0466c82f4fd35d
 6 af9f8112df35cf33 481208e5e8314602 d802c2dfd7cc48f6 43a76f011a73d317
   257f4a7d524d7b0b 6b2271a46f14c843 f73d759b839a2a21 1367bd36d15e8b40
 7 6730781342d1131b af9f8112df35cf33 481208e5e8314602 d802c2dfd7cc48f6
   81957ad408cec995 257f4a7d524d7b0b 6b2271a46f14c843 f73d759b839a2a21
 8 82e64c677356a82e 6730781342d1131b af9f8112df35cf33 481208e5e8314602
   10b62fdce4ebaa51 81957ad408cec995 257f4a7d524d7b0b 6b2271a46f14c843
 9 203578820a8f27d0 82e64c677356a82e 6730781342d1131b af9f8112df35cf33
   9937b3a0cb9248a1 10b62fdce4ebaa51 81957ad408cec995 257f4a7d524d7b0b
10 0bac2a84c29a1e2b 203578820a8f27d0 82e64c677356a82e 6730781342d1131b
   6ad288dab3de0d53 9937b3a0cb9248a1 10b62fdce4ebaa51 81957ad408cec995
11 dd3ff8a140485c25 0bac2a84c29a1e2b 203578820a8f27d0 82e64c677356a82e
   3149b728123c465e 6ad288dab3de0d53 9937b3a0cb9248a1 10b62fdce4ebaa51
12 e826239f830c5346 dd3ff8a140485c25 0bac2a84c29a1e2b 203578820a8f27d0
   4bb7b199c4ced186 3149b728123c465e 6ad288dab3de0d53 9937b3a0cb9248a1
13 32215ce49aae40f8 e826239f830c5346 dd3ff8a140485c25 0bac2a84c29a1e2b
   9a2872c72d790d49 4bb7b199c4ced186 3149b728123c465e 6ad288dab3de0d53
14 859533bac457f94e 32215ce49aae40f8 e826239f830c5346 dd3ff8a140485c25
   539f225d25ebeb4c 9a2872c72d790d49 4bb7b199c4ced186 3149b728123c465e
15 a88704d9962849f3 859533bac457f94e 32215ce49aae40f8 e826239f830c5346
   63bf0472ef24f7a5 539f225d25ebeb4c 9a2872c72d790d49 4bb7b199c4ced186
16 3aa5c566a6cfad1c a88704d9962849f3 859533bac457f94e 32215ce49aae40f8
   ce23f6380ead33c2 63bf0472ef24f7a5 539f225d25ebeb4c 9a2872c72d790d49
17 2e9c483a7c08c9c1 3aa5c566a6cfad1c a88704d9962849f3 859533bac457f94e
   b033f945f3e6b4a2 ce23f6380ead33c2 63bf0472ef24f7a5 539f225d25ebeb4c

18 5a68585ae0835231 2e9c483a7c08c9c1 3aa5c566a6cfad1c a88704d9962849f3
8a0187a9ce93d875 b033f945f3e6b4a2 ce23f6380ead33c2 63bf0472ef24f7a5
19 cf9cd481e6407ced 5a68585ae0835231 2e9c483a7c08c9c1 3aa5c566a6cfad1c
37a29fa30531bac7 8a0187a9ce93d875 b033f945f3e6b4a2 ce23f6380ead33c2
20 3f463f864f6474d9 cf9cd481e6407ced 5a68585ae0835231 2e9c483a7c08c9c1
0cf45bb3c07e847d 37a29fa30531bac7 8a0187a9ce93d875 b033f945f3e6b4a2
21 cea26288dff931a5 3f463f864f6474d9 cf9cd481e6407ced 5a68585ae0835231
34f1b5f46bf48a73 0cf45bb3c07e847d 37a29fa30531bac7 8a0187a9ce93d875
22 89634cd0f4f6c08a cea26288dff931a5 3f463f864f6474d9 cf9cd481e6407ced
3a728a543405a8e4 34f1b5f46bf48a73 0cf45bb3c07e847d 37a29fa30531bac7
23 625fa38464e5c880 89634cd0f4f6c08a cea26288dff931a5 3f463f864f6474d9
cee1b47a49b2fc42 3a728a543405a8e4 34f1b5f46bf48a73 0cf45bb3c07e847d
24 7dd21453a15a3b92 625fa38464e5c880 89634cd0f4f6c08a cea26288dff931a5
9308bfa1be1f800b cee1b47a49b2fc42 3a728a543405a8e4 34f1b5f46bf48a73
25 3d76277bc8cb0601 7dd21453a15a3b92 625fa38464e5c880 89634cd0f4f6c08a
480e017f5d1f0b1e 9308bfa1be1f800b cee1b47a49b2fc42 3a728a543405a8e4
26 c8d904196f5a1f54 3d76277bc8cb0601 7dd21453a15a3b92 625fa38464e5c880
4bd2f1f6e940c332 480e017f5d1f0b1e 9308bfa1be1f800b cee1b47a49b2fc42
27 b033139b58b6e423 c8d904196f5a1f54 3d76277bc8cb0601 7dd21453a15a3b92
f816ec1cbe0adafb 4bd2f1f6e940c332 480e017f5d1f0b1e 9308bfa1be1f800b
28 097768182cb65f57 b033139b58b6e423 c8d904196f5a1f54 3d76277bc8cb0601
62e3de54dcd8f974 f816ec1cbe0adafb 4bd2f1f6e940c332 480e017f5d1f0b1e
29 3196649ab5f5cc39 097768182cb65f57 b033139b58b6e423 c8d904196f5a1f54
f6887de116d0bd8f 62e3de54dcd8f974 f816ec1cbe0adafb 4bd2f1f6e940c332
30 f78d3d221d16965f 3196649ab5f5cc39 097768182cb65f57 b033139b58b6e423
c7e4859c2858ed3c f6887de116d0bd8f 62e3de54dcd8f974 f816ec1cbe0adafb
31 f58e9876b4984b51 f78d3d221d16965f 3196649ab5f5cc39 097768182cb65f57
621352b394b8ca02 c7e4859c2858ed3c f6887de116d0bd8f 62e3de54dcd8f974
32 38fbf0e726e04f78 f58e9876b4984b51 f78d3d221d16965f 3196649ab5f5cc39
4319856f17a0a430 621352b394b8ca02 c7e4859c2858ed3c f6887de116d0bd8f

33 f4be0b32a57597a2 38fbf0e726e04f78 f58e9876b4984b51 f78d3d221d16965f
   c6d392a3b4eb0ed8 4319856f17a0a430 621352b394b8ca02 c7e4859c2858ed3c
34 f8a6b3fe2e4f0634 f4be0b32a57597a2 38fbf0e726e04f78 f58e9876b4984b51
   602663c0f34eff33 c6d392a3b4eb0ed8 4319856f17a0a430 621352b394b8ca02
35 9bc3871be8046113 f8a6b3fe2e4f0634 f4be0b32a57597a2 38fbf0e726e04f78
   05542ecd9883c6ba 602663c0f34eff33 c6d392a3b4eb0ed8 4319856f17a0a430
36 f1bd2d46be619585 9bc3871be8046113 f8a6b3fe2e4f0634 f4be0b32a57597a2
   e47b9933bafdc655 05542ecd9883c6ba 602663c0f34eff33 c6d392a3b4eb0ed8
37 24c84b58d119affe f1bd2d46be619585 9bc3871be8046113 f8a6b3fe2e4f0634
   5ae0b1175beb5d2b e47b9933bafdc655 05542ecd9883c6ba 602663c0f34eff33
38 ec6d3abc2b291fd3 24c84b58d119affe f1bd2d46be619585 9bc3871be8046113
   9ecc381d277748a3 5ae0b1175beb5d2b e47b9933bafdc655 05542ecd9883c6ba
39 e266c1f77d5ee90e ec6d3abc2b291fd3 24c84b58d119affe f1bd2d46be619585
   d92f34c110296b32 9ecc381d277748a3 5ae0b1175beb5d2b e47b9933bafdc655
40 5adbaa463642b570 e266c1f77d5ee90e ec6d3abc2b291fd3 24c84b58d119affe
   83e8f410f859388e d92f34c110296b32 9ecc381d277748a3 5ae0b1175beb5d2b
41 50fdb7bb2e499a34 5adbaa463642b570 e266c1f77d5ee90e ec6d3abc2b291fd3
   257ed8ea645e933a 83e8f410f859388e d92f34c110296b32 9ecc381d277748a3
42 06514212bb7fa152 50fdb7bb2e499a34 5adbaa463642b570 e266c1f77d5ee90e
   466781db35181abe 257ed8ea645e933a 83e8f410f859388e d92f34c110296b32
43 673ed5a55ff2b07d 06514212bb7fa152 50fdb7bb2e499a34 5adbaa463642b570
   ba78f3545e7914f0 466781db35181abe 257ed8ea645e933a 83e8f410f859388e
44 125e2e5118393e2b 673ed5a55ff2b07d 06514212bb7fa152 50fdb7bb2e499a34
   4453b23a3e13b090 ba78f3545e7914f0 466781db35181abe 257ed8ea645e933a
45 07ee813df5910cec 125e2e5118393e2b 673ed5a55ff2b07d 06514212bb7fa152
   eae013a0510d23cc 4453b23a3e13b090 ba78f3545e7914f0 466781db35181abe
46 0a0508f0a1d719c3 07ee813df5910cec 125e2e5118393e2b 673ed5a55ff2b07d
   a93815eb58891016 eae013a0510d23cc 4453b23a3e13b090 ba78f3545e7914f0
47 0fc8f3b3efcb1b96 0a0508f0a1d719c3 07ee813df5910cec 125e2e5118393e2b
   a071cc73b966e801 a93815eb58891016 eae013a0510d23cc 4453b23a3e13b090

48 02aa5b28199f304a 0fc8f3b3efcb1b96 0a0508f0a1d719c3 07ee813df5910cec
a49f1e14f8a2be7a a071cc73b966e801 a93815eb58891016 eae013a0510d23cc
49 9223e1b34382f104 02aa5b28199f304a 0fc8f3b3efcb1b96 0a0508f0a1d719c3
bfe2106e512a7331 a49f1e14f8a2be7a a071cc73b966e801 a93815eb58891016
50 e01a1e47ee8d5656 9223e1b34382f104 02aa5b28199f304a 0fc8f3b3efcb1b96
592b899b35469a78 bfe2106e512a7331 a49f1e14f8a2be7a a071cc73b966e801
51 fa7b17aad857c2f4 e01a1e47ee8d5656 9223e1b34382f104 02aa5b28199f304a
eb6e85e4682c1671 592b899b35469a78 bfe2106e512a7331 a49f1e14f8a2be7a
52 0c523b7a3c84ab77 fa7b17aad857c2f4 e01a1e47ee8d5656 9223e1b34382f104
b5e80e871ac0c005 eb6e85e4682c1671 592b899b35469a78 bfe2106e512a7331
53 c773d8b69da1fde2 0c523b7a3c84ab77 fa7b17aad857c2f4 e01a1e47ee8d5656
be2b0602fc6f8f65 b5e80e871ac0c005 eb6e85e4682c1671 592b899b35469a78
54 c6b1bc79a4f23679 c773d8b69da1fde2 0c523b7a3c84ab77 fa7b17aad857c2f4
c80bdc57f38a05e4 be2b0602fc6f8f65 b5e80e871ac0c005 eb6e85e4682c1671
55 bef9bb0fe467fd60 c6b1bc79a4f23679 c773d8b69da1fde2 0c523b7a3c84ab77
1dab0bd116e434e5 c80bdc57f38a05e4 be2b0602fc6f8f65 b5e80e871ac0c005
56 8e3db3e380ec7f22 bef9bb0fe467fd60 c6b1bc79a4f23679 c773d8b69da1fde2
32ef50751734ffee 1dab0bd116e434e5 c80bdc57f38a05e4 be2b0602fc6f8f65
57 1003ec42412c7b7d 8e3db3e380ec7f22 bef9bb0fe467fd60 c6b1bc79a4f23679
1ec0d46f349fd058 32ef50751734ffee 1dab0bd116e434e5 c80bdc57f38a05e4
58 375facc76291f85e 1003ec42412c7b7d 8e3db3e380ec7f22 bef9bb0fe467fd60
59c8bc0488f9768b 1ec0d46f349fd058 32ef50751734ffee 1dab0bd116e434e5
59 bd113d92e0354fb9 375facc76291f85e 1003ec42412c7b7d 8e3db3e380ec7f22
e66c73db3fad397d 59c8bc0488f9768b 1ec0d46f349fd058 32ef50751734ffee
60 2f61d4fd8e36d9d4 bd113d92e0354fb9 375facc76291f85e 1003ec42412c7b7d
e9f21933e1c02948 e66c73db3fad397d 59c8bc0488f9768b 1ec0d46f349fd058
61 1b1ad88b92701ae2 2f61d4fd8e36d9d4 bd113d92e0354fb9 375facc76291f85e
6fd0c1719bcac335 e9f21933e1c02948 e66c73db3fad397d 59c8bc0488f9768b
62 93d09fc06a19c5da 1b1ad88b92701ae2 2f61d4fd8e36d9d4 bd113d92e0354fb9
b765273f571a571e 6fd0c1719bcac335 e9f21933e1c02948 e66c73db3fad397d

63 04bea2ce99cc3bf6 93d09fc06a19c5da 1b1ad88b92701ae2 2f61d4fd8e36d9d4
   6ab0e443c2f63714 b765273f571a571e 6fd0c1719bcac335 e9f21933e1c02948
64 02ebfc0a13492f52 04bea2ce99cc3bf6 93d09fc06a19c5da 1b1ad88b92701ae2
   77300c52e05af415 6ab0e443c2f63714 b765273f571a571e 6fd0c1719bcac335
65 1bf525abce8d6f04 02ebfc0a13492f52 04bea2ce99cc3bf6 93d09fc06a19c5da
   8faf12c33bb371b9 77300c52e05af415 6ab0e443c2f63714 b765273f571a571e
66 b6a36a3431547328 1bf525abce8d6f04 02ebfc0a13492f52 04bea2ce99cc3bf6
   fa8bb40b4e08100f 8faf12c33bb371b9 77300c52e05af415 6ab0e443c2f63714
67 ffdaf83202af0d72 b6a36a3431547328 1bf525abce8d6f04 02ebfc0a13492f52
   8045a82f723a9b4e fa8bb40b4e08100f 8faf12c33bb371b9 77300c52e05af415
68 12737373d2985232 ffdaf83202af0d72 b6a36a3431547328 1bf525abce8d6f04
   870dbce23bad8988 8045a82f723a9b4e fa8bb40b4e08100f 8faf12c33bb371b9
69 6189f68162b256b5 12737373d2985232 ffdaf83202af0d72 b6a36a3431547328
   8c059af157146580 870dbce23bad8988 8045a82f723a9b4e fa8bb40b4e08100f
70 20b0a9a1d21c482d 6189f68162b256b5 12737373d2985232 ffdaf83202af0d72
   f22b874c96785ec8 8c059af157146580 870dbce23bad8988 8045a82f723a9b4e
71 ef6d863c2127b394 20b0a9a1d21c482d 6189f68162b256b5 12737373d2985232
   b7aee28337d69dab f22b874c96785ec8 8c059af157146580 870dbce23bad8988
72 d3efe8b442689074 ef6d863c2127b394 20b0a9a1d21c482d 6189f68162b256b5
   22491ab9cdecb6b0 b7aee28337d69dab f22b874c96785ec8 8c059af157146580
73 4694354944a9f487 d3efe8b442689074 ef6d863c2127b394 20b0a9a1d21c482d
   659890a5818d0c50 22491ab9cdecb6b0 b7aee28337d69dab f22b874c96785ec8
74 b93c2403773dd08c 4694354944a9f487 d3efe8b442689074 ef6d863c2127b394
   88c2c2ac52c4f679 659890a5818d0c50 22491ab9cdecb6b0 b7aee28337d69dab
75 025848e3ab6b69d3 b93c2403773dd08c 4694354944a9f487 d3efe8b442689074
   750da3d4e16a1b64 88c2c2ac52c4f679 659890a5818d0c50 22491ab9cdecb6b0
76 396b53e58d04471b 025848e3ab6b69d3 b93c2403773dd08c 4694354944a9f487
   700486bf252cba75 750da3d4e16a1b64 88c2c2ac52c4f679 659890a5818d0c50
77 51b6f9a3c1ceeb4a 396b53e58d04471b 025848e3ab6b69d3 b93c2403773dd08c
   e6b3850de8ae6230 700486bf252cba75 750da3d4e16a1b64 88c2c2ac52c4f679

```
78 526a98f5dc595406 51b6f9a3c1ceeb4a 396b53e58d04471b 025848e3ab6b69d3
   4f0dcf74aea76f90 e6b3850de8ae6230 700486bf252cba75 750da3d4e16a1b64
79 deb3eeaa973bb9dd 526a98f5dc595406 51b6f9a3c1ceeb4a 396b53e58d04471b
   3665b5dbb6c2e055 4f0dcf74aea76f90 e6b3850de8ae6230 700486bf252cba75
```

هشت کلمه‌ی $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ خروجی تکرار آخر تابع گردساز را نمایش می‌دهند:

X_0 = 2a7f1d895fd58e0b ⊎ deb3eeaa973bb9dd = 09330c33f71147e8

X_1 = eaae96d1a673c741 ⊎ 526a98f5dc595406 = 3d192fc782cd1b47

X_2 = 015a2173796c1a88 ⊎ 51b6f9a3c1ceeb4a = 53111b173b3b05d2

X_3 = f6352ca156acaff7 ⊎ 396b53e58d04471b = 2fa08086e3b0f712

X_4 = c662113e9ebb4d64 ⊎ 3665b5dbb6c2e055 = fcc7c71a557e2db9

X_5 = 17b61a85e2ccf0a9 ⊎ 4f0dcf74aea76f90 = 66c3e9fa91746039

X_6 = 37eb9a6660feb519 ⊎ e6b3850de8ae6230 = 1e9f1f7449ad1749

X_7 = 8f2ebe9a81e6a2c5 ⊎ 700486bf252cba75 = ff334559a7135d3a

مقدار درهم برای این پیام به صورت زیر است:

```
09330c33f71147e8 3d192fc782cd1b47 53111b173b3b05d2 2fa08086e3b0f712
                 fcc7c71a557e2db9 66c3e9fa91746039
```

الف-۶-۱۱ مثال۱۱

در این مثال رشته-داده شامل یک رشته‌ای ۳۲ بایتی است، یعنی نسخه‌ی ASCII-کدی

‘abcdbcdecdefdefgefghfghighiihjk’

کددرهم رشته‌ی ۳۸۴ بیتی ذیل است:

```
d4cc646a83a55044 df94814db93b6062 e656623db0b9e2da b8819174589bf0c9
                 d7192b9799e30169 8b97adaa3d82e20c
```

الف-۷ تابع درهم‌ساز اختصاصی ۷

الف-۷-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.
کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
19FA61D75522A466 9B44E39C1D2E1726 C530232130D407F8 9AFEE0964997F7A7
3E83BE698B288FEB CF88E3E03C4F0757 EA8964E59B63D937 08B138CC42A66EB3
```

الف-۷-۲ مثال۲

در این مثال رشته-داده شامل یک بایت تنهاست، معادل کد ASCII ’a‘ .
کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
8ACA2602792AEC6F 11A67206531FB7D7 F0DFF59413145E69 73C45001D0087B42
D11BC645413AEFF6 3A42391A39145A59 1A92200D560195E5 3B478584FDAE231A
```

الف-۷-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.
بعد از فرایند لایه‌گذاری، ماتریس ۸ در ۸، Z' ،مشتق شده از رشته-داده، همانند ذیل است:

```
61 62 63 80 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 18
```

ماتریس K_0 (از مقدار اولیه‌ی IV) و ماتریس X'' به صورت زیر هستند.

```
00 00 00 00 00 00 00 00          61 62 63 80 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
```

```
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 18
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای K_i، برای i=1 تا ۱۰ و W' آورده شده است.

```
i = 1:
30 0B EE C0 AF 90 29 67          0F 34 9A FF 3F F3 2F E0
28 28 28 28 28 28 28 28          EB CD CD 13 CD 26 DE 87
28 28 28 28 28 28 28 28          2D 2C 98 98 5A 98 B4 C2
28 28 28 28 28 28 28 28          89 03 83 8F 8F 06 8F 0C
28 28 28 28 28 28 28 28          00 00 00 00 00 00 00 00
28 28 28 28 28 28 28 28          00 00 00 00 00 00 00 00
28 28 28 28 28 28 28 28          05 14 05 28 11 0A 2D 05
28 28 28 28 28 28 28 28          00 00 00 00 00 00 00 00
i = 2:
3B AB 89 F8 EA D1 AE 24          1D 0D 4C DA 43 F6 B0 98
44 45 45 66 45 E9 CB AF          E4 5E 3F B8 7B C7 AA 10
70 FE A4 A4 C5 A4 B2 89          C3 31 D1 56 FD E7 7B 8F
C5 FA A9 E1 E1 CC E1 A0          68 2F 47 A1 BE 4A 53 39
48 AC C0 5C FC FC B8 FC          B2 A2 B8 2F 20 72 F0 6C
8F F7 0E 26 90 8F 8F 69          03 D9 F4 6C 67 B1 79 72
96 79 14 07 D7 85 79 79          2C 67 87 6E FD 5C 25 F8
F8 A8 F8 68 B8 C8 78 F8          44 E6 4C 70 50 7C D8 26
i = 3:
D3 19 BF DB 30 46 70 58          EF ED 35 67 80 8E 8D 63
29 5B 23 D1 AF CF 37 DB          2F 03 49 91 5B 18 5C 24
01 2C 8A C2 8B 95 AC 98          77 96 F6 03 BF AA F8 E3
81 63 9E B1 C0 B2 06 A7          0A DC 04 7B 58 5A A5 A1
44 5E 60 7A B0 B2 09 DB          47 96 DA 7F 56 E4 CC 29
```

73 5B 2C CF BC 8C BC 71 20 70 D5 D8 50 01 C8 98

DC 67 09 24 EF ED DD D3 A7 4C 23 FA F6 81 49 A1

7B 8D 3B F0 D7 3B 7D 19 4A CE 46 7D 7D B0 73 A9

i = 4:

38 BE AA C1 DE 11 65 86 95 BD DE 1E CA 0F CA 19

68 7C F3 D0 4A 87 33 7F D3 C1 CF 6C A0 2E 41 E8

F3 37 FA DB 98 AD F0 57 74 C3 5C 63 15 C5 B9 8A

C5 E2 42 58 EE 35 8D BC 36 F0 4E 42 FE 2D D0 5E

11 09 F0 E8 99 6E 24 7E 0A 3C 50 76 A1 91 F8 EC

01 C5 D6 ED 10 B0 34 01 48 6B C7 3E 61 D2 A4 DC

FB C9 52 F1 7B 28 EC D3 ED B8 F0 C5 2C F0 5C 72

32 56 DC 0C C7 F1 27 40 FA 3D 00 D4 FB 9A 66 FF

i = 5:

AF 25 A5 20 94 9B CF 14 06 A6 BA 18 05 54 8D 33

C1 36 26 A9 E3 C4 53 4D 84 55 FE C4 1F B2 0B 1C

E6 0F 7D 86 77 40 F9 E1 6E A2 93 49 3F 17 89 B7

91 5D E6 BB E2 6A 06 29 7D 02 C9 A0 52 85 BB EF

96 5A 54 CC 4C FE 5E 8D AC 55 D7 A9 44 48 89 A9

BE E9 31 CB 62 32 3A A6 CB DE BE 43 AA 4D B5 A0

B1 7B 59 18 96 84 6A 47 60 A6 BA C0 25 D9 4F 8C

D4 F0 C9 36 27 59 AF 31 D7 E4 62 E5 D4 A8 CC C0

i = 6:

E2 F9 B5 C0 25 37 0B B0 DB 1D A8 4A 33 38 4D B3

39 2B CB A2 16 84 94 A5 97 4C 8E 1A 3E 51 F3 48

60 8A F8 CE FA 34 8C 14 47 66 64 C2 33 F5 F2 A9

7A A5 37 64 41 8C 92 19 85 FD AA B1 D5 CB C3 6E

B3 F3 46 A1 FA 83 3F 89 5D 89 59 F2 E1 F8 71 D4

97 49 3F 48 78 02 CF 7C 8C 1F B9 78 8C 16 DD 05

DC AD E8 BA 1E 00 8F 23 62 AF 63 5F 6D EE D5 F4

92 77 4F 49 ED B0 32 3D D8 5B 74 35 5F 8A 98 47

i = 7:

75 41 63 82 77 4D FF 2F	59 3D 86 BD A8 CE 25 E5
FF FA 38 D0 55 03 46 00	BB 33 95 78 26 63 7D 82
BF 7D 02 49 3E 98 F3 61	EF 46 1D AE DC AD 0C 3C
F4 A8 60 C2 9A E5 CE 0B	AF A0 E2 86 5E 8B A3 F9
C8 DF 5A 44 EE 5D 9D 27	C8 8C 0B 43 27 84 31 F4
23 F4 5A 55 04 75 00 A4	41 5F 51 64 4E 55 78 C2
B0 16 10 12 02 F9 E2 8C	F4 C7 C3 B5 EE A4 C5 86
AC 30 CD 29 68 33 33 1D	49 F8 AB 68 4A 4C 96 B7

i = 8:

03 6B F1 82 68 84 AD 89	9C 0D 38 97 73 B2 E4 35
99 40 C6 62 D8 46 71 63	4D 44 89 58 D4 59 27 E8
4C 43 3E 17 4B 19 C2 10	AD 59 2E B0 4C A3 63 32
E2 9C CF D3 4C FF 86 C5	E0 D4 70 F3 83 5A 15 59
21 FF 11 A0 42 DF 26 53	9A 92 69 8C 76 40 A1 51
1B 8E 00 CB 6C E4 4B 13	57 2E 81 EA CB A4 3C 36
A6 12 3B F7 A3 47 B7 CE	5D 63 2F A7 36 BE 4B 61
D9 18 90 0E 3B 28 33 CA	40 0F DA CB 8B 9D E3 8A

i = 9:

D0 1C 67 7A 0A 9A 2C F9	4B F0 5E 9B 46 14 16 D0
2A 94 2F 53 4A 63 B6 B2	72 A8 C1 34 47 13 17 2D
88 42 22 46 FE AC A8 B4	17 33 2A 69 FB 34 98 98
47 4A 5C C7 3D 58 35 59	83 B1 EE 37 93 47 EC A0
74 A6 92 5D A5 5C 6F A1	3B 39 67 11 23 35 B5 78
77 17 E6 8C C4 73 5C 39	FC 78 3D 1F 9D 2F B6 AE
08 2A 3B 0B 53 EC 1A C6	3C F9 38 64 96 9B DE 6C
2A F6 58 EB 81 4D E7 62	42 5A D1 47 6C 0C 49 AE

i = 10:

48 95 48 B6 01 EE BC 3A	2F 46 2B 24 C6 F4 86 BB
A5 0D 6B C6 6B ED 8E 81	16 B6 56 2C 73 B4 02 0B

```
E0 CE 3D CF 88 26 5A 75          F3 04 3E 3A 73 1B CE 72
C2 8C 4A DB C0 F6 9C E9          1A E1 B3 03 D9 7E 6D 4C
54 B7 9C D5 7F 71 85 13          71 81 EE BD B6 C5 7E 27
43 41 4B 8A 97 7D 0B 7B          7D 0E 34 95 71 14 CB D6
63 19 35 BB DB F6 15 7A          C7 97 FC 9D 95 D8 B5 82
6A 7A 4E F6 37 01 82 27          D2 25 29 20 76 D4 EE ED
```

مقدار Y' خروجی از تابع گردساز به صورت زیر است.

```
4E 24 48 A4 C6 F4 86 BB
16 B6 56 2C 73 B4 02 0B
F3 04 3E 3A 73 1B CE 72
1A E1 B3 03 D9 7E 6D 4C
71 81 EE BD B6 C5 7E 27
7D 0E 34 95 71 14 CB D6
C7 97 FC 9D 95 D8 B5 82
D2 25 29 20 76 D4 EE F5
```

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
4E2448A4C6F486BB 16B6562C73B4020B F3043E3A731BCE72 1AE1B303D97E6D4C
7181EEBDB6C57E27 7D0E34957114CBD6 C797FC9D95D8B582 D225292076D4EEF5
```

الف-۷-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII

´message digest´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
378C84A4126E2DC6 E56DCC7458377AAC 838D00032230F53C E1F5700C0FFB4D3B
8421557659EF55C1 06B4B52AC5A4AAA6 92ED920052838F33 62E86DBD37A8903E
```

الف-۷-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۲۶بایتی است، معادل کد ASCII

´abcdefghijklmnopqrstuvwxyz´

کددرهم رشته‌ی۵۱۲ بیتی ذیل است:

```
F1D754662636FFE9 2C82EBB9212A484A 8D38631EAD4238F5 442EE13B8054E41B
08BF2A9251C30B6A 0B8AAE86177AB4A6 F68F673E7207865D 5D9819A3DBA4EB3B
```

الف-۷-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´

کددرهم رشته‌ی۵۱۲ بیتی ذیل است:

```
DC37E008CF9EE69B F11F00ED9ABA2690 1DD7C28CDEC066CC 6AF42E40F82F3A1E
08EBA26629129D8F B7CB57211B9281A6 5517CC879D7B9621 42C65F5A7AF01467
```

الف-۷-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار

’1234567890‘

کددرهم رشته‌ی۵۱۲ بیتی ذیل است:

```
466EF18BABB0154D 25B9D38A6414F5C0 8784372BCCB204D6 549C4AFADB601429
4D5BD8DF2A6C44E5 38CD047B2681A51A 2C60481E88C5A20B 2C2A80CF3A9A083B
```

الف-۷-۸ مثال ۸

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
61 62 63 64 62 63 64 65        00 00 00 00 00 00 00 00
63 64 65 66 64 65 66 67        00 00 00 00 00 00 00 00
65 66 67 68 66 67 68 69        00 00 00 00 00 00 00 00
67 68 69 6A 68 69 6A 6B        00 00 00 00 00 00 00 00
80 00 00 00 00 00 00 00        00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00        00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00        00 00 00 00 00 00 00 00
```

```
00 00 00 00 00 00 00 00          00 00 00 00 00 00 01 00
```

اولین ماتریس Z' به‌صورت زیر است:

```
61 62 63 64 62 63 64 65
63 64 65 66 64 65 66 67
65 66 67 68 66 67 68 69
67 68 69 6A 68 69 6A 6B
80 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
```

برای اولین ماتریس Z' ، ماتریس K_0 (از مقدار اولیه‌ی IV) و ماتریس X'' به‌صورت زیر هستند.

```
61 62 63 64 62 63 64 65          00 00 00 00 00 00 00 00
63 64 65 66 64 65 66 67          00 00 00 00 00 00 00 00
65 66 67 68 66 67 68 69          00 00 00 00 00 00 00 00
67 68 69 6A 68 69 6A 6B          00 00 00 00 00 00 00 00
80 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00          00 00 00 00 00 00 00 00
```

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای K_i، برای $i=1$ تا ۱۰ و W' آورده شده است.

```
i = 1:
30 0B EE C0 AF 90 29 67          86 B9 56 DD B4 BD 40 C2
28 28 28 28 28 28 28 28          0B 48 C1 2E 83 9C 2E 41
28 28 28 28 28 28 28 28          40 5E 0A ED 5C E9 42 E7
28 28 28 28 28 28 28 28          B2 1E 5B 93 43 07 7C 4D
28 28 28 28 28 28 28 28          19 04 67 A3 57 CF DA ED
28 28 28 28 28 28 28 28          59 36 7D 57 F8 E7 EA 60
28 28 28 28 28 28 28 28          98 D1 1B 6A C6 1C 4B CD
28 28 28 28 28 28 28 28          5E B9 76 56 F3 51 F4 43
```

i = 2:

3B AB 89 F8 EA D1 AE 24	10 54 A2 C2 9E 00 80 4F
44 45 45 66 45 E9 CB AF	6B C6 9F 0A 98 41 BA 45
70 FE A4 A4 C5 A4 B2 89	6B 0B DE 38 1B F6 5A 3F
C5 FA A9 E1 E1 CC E1 A0	34 F5 52 E4 38 30 DA 32
48 AC C0 5C FC FC B8 FC	A7 4E 3B C9 F2 58 65 5B
8F F7 0E 26 90 8F 8F 69	2C 84 5C F8 DE BA 57 52
96 79 14 07 D7 85 79 79	0B 0B CB 4F 5F 5F 13 10
F8 A8 F8 68 B8 C8 78 F8	B4 43 90 D6 92 4F 65 12

i = 3:

D3 19 BF DB 30 46 70 58	8F 55 E3 10 51 E9 E7 43
29 5B 23 D1 AF CF 37 DB	F3 AE 56 A1 2E 86 11 01
01 2C 8A C2 8B 95 AC 98	01 78 57 78 4C 25 EE 95
81 63 9E B1 C0 B2 06 A7	8B 13 D5 66 9A EA A5 53
44 5E 60 7A B0 B2 09 DB	55 E0 9A 46 78 79 57 56
73 5B 2C CF BC 8C BC 71	E2 3E F3 AF D4 5F 66 62
DC 67 09 24 EF ED DD D3	05 E9 CA 43 59 FC 08 53
7B 8D 3B F0 D7 3B 7D 19	6A 11 68 9A 3D 24 86 2C

i = 4:

38 BE AA C1 DE 11 65 86	BD A3 5F AC C8 4B 7B 24
68 7C F3 D0 4A 87 33 7F	D4 D5 53 36 8A FA 90 C8
F3 37 FA DB 98 AD F0 57	7D 9A 3C 52 B5 B9 28 0B
C5 E2 42 58 EE 35 8D BC	FE CD D7 48 5D 98 AC 21
11 09 F0 E8 99 6E 24 7E	F6 D3 E3 F5 A1 C0 68 F0
01 C5 D6 ED 10 B0 34 01	D9 77 56 2D F1 C4 3C B6
FB C9 52 F1 7B 28 EC D3	C2 85 71 D3 B2 94 91 69
32 56 DC 0C C7 F1 27 40	E2 B9 81 C5 7C 60 42 23

i = 5:

AF 25 A5 20 94 9B CF 14	15 03 B3 53 CF 70 04 4D
C1 36 26 A9 E3 C4 53 4D	D0 74 26 9B 60 EC 9B 92

E6 0F 7D 86 77 40 F9 E1 BE 22 90 B3 34 54 C2 84

91 5D E6 BB E2 6A 06 29 20 F3 7D 53 7D D1 C1 BA

96 5A 54 CC 4C FE 5E 8D 87 0E 9B F5 41 7C 2D 29

BE E9 31 CB 62 32 3A A6 A8 52 51 52 21 71 D5 9D

B1 7B 59 18 96 84 6A 47 96 9C 26 6D 4A B9 C6 AB

D4 F0 C9 36 27 59 AF 31 5A 2B DD 3C D9 8A D1 04

i = 6:

E2 F9 B5 C0 25 37 0B B0 B1 44 C5 6B 09 97 59 91

39 2B CB A2 16 84 94 A5 CF 0D 2C 26 C0 C7 93 54

60 8A F8 CE FA 34 8C 14 18 D0 BE 9C 7A 35 09 8A

7A A5 37 64 41 8C 92 19 32 8B E8 B4 2C B0 10 2A

B3 F3 46 A1 FA 83 3F 89 02 01 B5 CC 2C 68 E9 9C

97 49 3F 48 78 02 CF 7C 12 BF E0 28 EB 7D 3F F1

DC AD E8 BA 1E 00 8F 23 49 BD 0B 4E 55 81 21 AA

92 77 4F 49 ED B0 32 3D 35 F4 59 17 F1 5C 49 DF

i = 7:

75 41 63 82 77 4D FF 2F DD D3 6C 6C F0 7A C1 16

FF FA 38 D0 55 03 46 00 03 42 87 2D A6 3A 4C F4

BF 7D 02 49 3E 98 F3 61 5D C0 C5 7D 6B BC 49 81

F4 A8 60 C2 9A E5 CE 0B 7C 12 58 40 F0 CD DA 1E

C8 DF 5A 44 EE 5D 9D 27 46 AD D5 C4 F9 77 40 C7

23 F4 5A 55 04 75 00 A4 FF 2E 7D 33 E9 7D 27 BA

B0 16 10 12 02 F9 E2 8C 2C CC DF EF 3A 86 58 08

AC 30 CD 29 68 33 33 1D FB AC B4 52 D2 63 9C 25

i = 8:

03 6B F1 82 68 84 AD 89 7B 3B 3C 7B 2D 73 FF 3C

99 40 C6 62 D8 46 71 63 32 7A 01 65 DD 7C 8C 7A

4C 43 3E 17 4B 19 C2 10 0F 70 81 E9 7B A3 B6 80

E2 9C CF D3 4C FF 86 C5 25 DF D5 33 66 08 A2 55

21 FF 11 A0 42 DF 26 53 AB 95 54 FC ED D2 51 92

```
1B 8E 00 CB 6C E4 4B 13          10 3A 15 9C FE CA CF 6E
A6 12 3B F7 A3 47 B7 CE          38 DA 67 14 8A 69 EB B3
D9 18 90 0E 3B 28 33 CA          92 2A 69 0B 03 4B 46 69
```

i = 9:

```
D0 1C 67 7A 0A 9A 2C F9          56 21 86 2A 9C 0B D3 95
2A 94 2F 53 4A 63 B6 B2          D4 5A B8 28 42 F2 59 DC
88 42 22 46 FE AC A8 B4          B2 55 11 33 27 2D E8 43
47 4A 5C C7 3D 58 35 59          B7 2C 18 04 84 19 B2 C7
74 A6 92 5D A5 5C 6F A1          0A DD FF 03 52 91 16 83
77 17 E6 8C C4 73 5C 39          3E A7 8D 11 02 CF E8 C8
08 2A 3B 0B 53 EC 1A C6          A1 22 69 ED AD B3 2A B4
2A F6 58 EB 81 4D E7 62          BE 53 E9 F0 7C B0 79 E7
```

i = 10:

```
48 95 48 B6 01 EE BC 3A          16 5A 82 D1 23 C3 52 8F
A5 0D 6B C6 6B ED 8E 81          26 E9 35 9E 6B C5 7A 23
E0 CE 3D CF 88 26 5A 75          17 EE A9 FF B7 C7 B4 99
C2 8C 4A DB C0 F6 9C E9          71 FD 96 BC 8F 74 63 4E
54 B7 9C D5 7F 71 85 13          B3 BE 30 9F 01 2A 59 09
43 41 4B 8A 97 7D 0B 7B          72 91 14 59 5F 08 6E 76
63 19 35 BB DB F6 15 7A          07 18 AF E3 65 BC 09 DE
6A 7A 4E F6 37 01 82 27          B6 AF A1 80 BC EC 2A 98
```

مقدار Y' خروجی از تابع گردساز برای اولین ماتریس Z' به‌صورت زیر است.

```
77 38 E1 B5 41 A0 36 EA
45 8D 50 F8 0F A0 1C 44
72 88 CE 97 D1 A0 DC F0
16 95 FF D6 E7 1D 09 25
33 BE 30 9F 01 2A 59 09
72 91 14 59 5F 08 6E 76
07 18 AF E3 65 BC 09 DE
```

B6 AF A1 80 BC EC 2A 98

دومین ماتریس Z' به‌صورت زیر است.

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00

00 00 00 00 00 00 01 00

برای دومین ماتریس Z' ، ماتریس K_0 (از مقدار اولیه‌ی IV) و ماتریس X'' به‌صورت زیر هستند.

77 38 E1 B5 41 A0 36 EA	77 38 E1 B5 41 A0 36 EA
45 8D 50 F8 0F A0 1C 44	45 8D 50 F8 0F A0 1C 44
72 88 CE 97 D1 A0 DC F0	72 88 CE 97 D1 A0 DC F0
16 95 FF D6 E7 1D 09 25	16 95 FF D6 E7 1D 09 25
33 BE 30 9F 01 2A 59 09	33 BE 30 9F 01 2A 59 09
72 91 14 59 5F 08 6E 76	72 91 14 59 5F 08 6E 76
07 18 AF E3 65 BC 09 DE	07 18 AF E3 65 BC 09 DE
B6 AF A1 80 BC EC 2B 98	B6 AF A1 80 BC EC 2A 98

در زیر (نمایش مبنای شانزده) مقادیر متوالی متغیرهای K_i، برای $i=1$ تا ۱۰ و W' آورده شده است.

i = 1:

1A 78 4D 7D BD 4C 17 E6	18 23 C6 E8 87 B8 01 4F
27 31 10 AA 63 C5 9E 25	00 00 00 00 00 00 00 00
7A 2E B7 48 C4 5D E0 23	00 00 00 00 00 00 00 00
6D 0D 61 9F 6C 1D 80 AE	00 00 00 00 00 00 00 00
01 A2 D5 6E DB 41 D9 A0	00 00 00 00 00 00 00 00
E9 06 4C D1 27 95 FA 86	8C 23 05 AF 46 26 23 23
77 62 31 BC B4 4E C6 01	00 00 00 00 00 00 00 00

6F CD BC 98 10 78 6F EC 00 00 00 00 00 00 00 00

i = 2:

EB 0F 86 07 40 38 54 4F DF 8A 74 7E 14 4C 22 D0

87 EF DC C8 FE 45 3D 83 2B 04 B7 AE 74 89 5A 13

99 0E F5 4E 73 1F C0 EA 2F FD BC A4 26 03 AD 74

EF E0 05 7F D2 C2 41 39 99 67 EA 50 34 08 BD B9

65 8F 5D 92 3E 9A AF 47 A8 7B 8E 1A 3B 56 CD 91

A9 1D 1C 13 BD 15 73 41 77 59 60 2D DD A2 4A 70

81 AD 80 BD 88 B3 B3 C3 03 43 90 91 2B DE 8E 37

16 26 63 99 AC 18 5D D0 48 6B C0 54 B9 C6 72 C9

i = 3:

7A A3 A3 3A 99 FD F6 5E 6B 92 48 05 C3 F4 1A 6D

E0 78 67 CD 3E 60 BF A7 45 20 59 41 0D 59 73 6D

BC 06 8D 5D 98 70 34 84 AF 72 CF 6A 4B B6 11 F4

80 E8 69 7D 44 CF 6B E6 A2 6D AD C1 12 CC 43 6C

7E 35 09 07 AF 76 70 C3 95 8F C4 AE 60 94 74 74

3B 7E 15 0D CA 5E A9 0A 4B AB 72 C2 3E 2C BC 6D

8D 10 98 19 22 3B FC 57 ED BF 23 B0 D6 82 B0 E8

AB DE A9 DD D3 B6 68 14 C0 4B 32 6B B5 14 B7 BB

i = 4:

3D 21 15 88 E4 48 75 78 78 5A 13 A3 25 81 79 C9

47 BF 56 CC 8E D4 63 CA DC 69 90 E0 14 F2 39 AC

AE F0 D0 31 74 25 3C 4E 89 5A 8F 66 7F F9 FC E3

08 F7 59 13 4F 6D DD 37 3B 5C C5 02 8C 4D 96 0A

C9 70 32 87 D8 F2 C1 E8 00 28 03 E7 DA 63 5E F5

90 E9 2D 7C AB A0 8E A7 DA 35 A5 BF B6 AB C7 EA

BF 22 A6 93 C1 6E 34 74 0D 5B 90 B8 88 56 C7 9F

58 40 F3 10 BF 03 3C 14 65 09 D2 D8 ED DA C6 B1

i = 5:

AE 58 59 43 80 F4 F6 14 6B 77 9A 58 6E 21 06 C1

14 5C 2E E0 5F B0 8E FD A7 2D B3 6D 1D AD 9E 3C

CF B7 1F C1 9A AC 6B 6A 32 CF E9 10 D3 AD CD EB

92 5C 25 E7 6C 28 7B 6B EE 4B 44 77 56 BC BC 63

57 B5 8E 30 FB E4 61 9B 41 05 39 5E 0B A3 8A 46

38 5D B4 49 F9 44 F8 C9 07 B9 8B 76 67 41 AC BD

A5 EE 29 38 0C 2D A8 70 E9 86 74 54 82 35 6F D9

45 8B FE 5E 05 C3 A6 89 27 FB C9 68 EE 1E C7 57

i = 6:

B1 F3 E2 33 93 63 14 AC 53 41 C7 63 02 40 D8 3F

DD 80 87 12 BF E5 70 0E 7F D8 0D FB 5D 97 CF 7A

A5 F5 16 A8 2A 82 CC 76 52 47 5A 93 4A BC D9 84

8A F5 DD F3 5F B1 11 57 95 47 26 76 78 E9 10 42

62 34 D3 BC 57 72 C7 DC E5 BA FB 23 2C 32 7B 6D

8D E2 8A 61 DC 88 CB 1A 62 CA FA 6D 35 F6 AA 13

53 35 F7 4C 99 ED 19 26 43 BF 3B F2 1B 0D B4 46

95 01 75 82 F7 A6 F7 2D BC 1C 9F 38 97 77 17 5B

i = 7:

7E 42 E3 38 39 72 B7 82 61 E7 C2 37 B0 E6 F6 2B

79 B2 EA 12 B3 68 75 B0 46 FE 01 CA 0E 34 5A 26

D8 8D 5F 05 2F AA 73 D2 80 2E F8 49 0D 5F 17 60

90 FC 91 61 30 BB 7B 5C 89 D5 48 F0 59 6D 73 E8

5A 1B F6 C2 20 10 61 23 72 D8 71 5E 44 80 9B E3

E5 31 C5 68 BC 4F 85 F8 8C 90 07 54 63 6B 77 0D

60 72 0A BA A7 90 27 03 63 1B 4E CF D7 C6 5D B5

A7 FD 03 BB E3 E9 CA 19 91 92 11 87 0F FE EA AB

i = 8:

12 EF 8A A7 F3 B5 7E F6 42 C9 DC 71 10 DA FA 7C

E9 59 60 9F 18 84 D3 ED 02 5B 59 54 A2 45 83 20

93 3E 12 E9 EA 51 D7 C1 53 B6 C4 85 4D C3 52 A5

```
EF DA 8A 82 CB 14 13 93          3B 65 C0 24 87 E8 20 BD
4C F0 7B 81 0D 03 9C F3          C5 3C E3 C4 9C DE 93 9F
2F 40 9C A8 76 D4 7D A3          CD 47 4A B3 CB C3 69 1B
32 72 85 CE 7A BD 39 58          24 5E FB 0E 45 E6 7A 96
06 1A CE 00 E7 5F EC B5          2B 36 CC A8 8A 64 C1 40
```

i = 9:

```
7C D9 89 12 FC AB 39 B2          AC C3 BD D6 26 A6 41 F0
20 E1 E9 E6 79 8D 5E 4F          E7 D8 5F 60 03 D2 7B F8
99 70 2C 2A CA E1 07 48          3F 48 9A 48 16 88 0E 1D
A4 85 C1 1F 74 6C 23 DC          D9 C7 62 1D 42 6F 86 A4
CF C8 1D F4 64 41 C6 1B          AD A6 9F 9A 29 CC 8C 6D
7B 0D 6B 84 2A 58 16 40          14 63 22 F6 04 B0 94 F4
4F 0A 55 C3 38 6A 0C 2D          E9 1D 7D 05 0C A8 44 F4
E6 31 16 BA AE C9 AC EC          A7 B1 5B F5 48 C5 2E F7
```

i = 10:

```
B4 74 E1 56 96 31 B9 6C          5D A0 9F 11 4E 31 46 8B
21 A1 B6 33 CC 89 68 1A          B0 5B A0 58 EB C4 53 0C
B1 97 25 86 7B 2B 3F 09          F8 F2 94 C5 0F 4E B9 92
4C 73 C7 62 93 A8 15 CF          11 50 9D 2F 6F F4 55 4C
55 15 C0 C0 9A 05 05 16          25 03 F8 9C 1A EF E7 12
23 44 8D 8D D3 5F B3 6E          09 05 62 60 A1 0D 65 20
7E 6C 2D 37 12 D0 F3 3E          94 83 05 43 C8 43 93 38
CE B8 04 F2 8D 9F C9 99          C2 F4 DA 98 A0 D7 C8 65
```

مقدار Y' خروجی از تابع گردساز برای اولین ماتریس Z' به‌صورت زیر است.

```
2A 98 7E A4 0F 91 70 61
F5 D6 F0 A0 E4 64 4F 48
8A 7A 5A 52 DE EE 65 62
07 C5 62 F9 88 E9 5C 69
16 BD C8 03 1B C5 BE 1B
```

```
7B 94 76 39 FE 05 0B 56
93 9B AA A0 AD FF 9A E6
74 5B 7B 18 1C 3B E3 FD
```

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
2A987EA40F917061 F5D6F0A0E4644F48 8A7A5A52DEEE6562 07C562F988E95C69
16BDC8031BC5BE1B 7B947639FE050B56 939BAAA0ADFF9AE6 745B7B181C3BE3FD
```

الف-۷-۹ مثال ۹

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای $۱۰^۶$ بار تکرار می‌شود.

کددرهم رشته‌ی ۵۱۲ بیتی ذیل است:

```
0C99005BEB57EFF5 0A7CF005560DDF5D 29057FD86B20BFD6 2DECA0F1CCEA4AF5
1FC15490EDDC47AF 32BB2B66C34FF9AD 8C6008AD677F7712 6953B226E4ED8B01
```

الف-۸ تابع درهم‌ساز اختصاصی ۸

الف-۸-۱ مثال ۱

در این مثال رشته-داده رشته‌ای تهی است، به عبارت دیگر رشته‌ای به طول صفر.

کد درهم‌ساز، رشته‌ی ۲۲۴ بیتی زیر است.

```
d14a028c 2a3a2bc9 476102bb 288234c4 15a2b01f 828ea62a c5b3e42f
```

الف-۸-۲ مثال ۲

در این مثال رشته-داده شامل تنها یک بایت است؛ معادل کد ASCII حرف ’a‘ .

کد درهم‌ساز، رشته‌ی ۲۲۴ بیتی زیر است.

```
abd37534 c7d9a2ef b9465de9 31cd7055 ffdb8879 563ae980 78d6d6d5
```

الف-۸-۳ مثال ۳

در این مثال رشته-داده رشته‌ای سه بایتی شامل معادل کد ASCII ’abc‘ است. این معادل رشته بیت ’01100001 01100010 01100011‘ است.

پس از فرایند لایه گذاری، بلوک ۱۶ کلمه‌ای منفردی که از رشته داده بدست می‌آید به‌صورت زیر است.

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018
```

در ادامه (نمایش در مبنای شانزده) مقادیر متوالی متغیرهای $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ آمده است.

```
init: c1059ed8 367cd507 3070dd17 f70e5939 ffc00b31 68581511 64f98fa7 befa4fa4
   0 0e96b2da c1059ed8 367cd507 3070dd17 0434225e ffc00b31 68581511 64f98fa7
   1 c20dab6b 0e96b2da c1059ed8 367cd507 9cab416f 0434225e ffc00b31 68581511
   2 ab113b7a c20dab6b 0e96b2da c1059ed8 82177fe8 9cab416f 0434225e ffc00b31
   3 8253cc1a ab113b7a c20dab6b 0e96b2da 8346b27d 82177fe8 9cab416f 0434225e
   4 08a0dc0c 8253cc1a ab113b7a c20dab6b 05b557db 8346b27d 82177fe8 9cab416f
   5 b2ca3a91 08a0dc0c 8253cc1a ab113b7a 898dc7bb 05b557db 8346b27d 82177fe8
   6 0b6b9023 b2ca3a91 08a0dc0c 8253cc1a a2e49147 898dc7bb 05b557db 8346b27d
   7 f09d116d 0b6b9023 b2ca3a91 08a0dc0c 7a84120d a2e49147 898dc7bb 05b557db
   8 ed6fa633 f09d116d 0b6b9023 b2ca3a91 c037faad 7a84120d a2e49147 898dc7bb
   9 55e6a367 ed6fa633 f09d116d 0b6b9023 aae50091 c037faad 7a84120d a2e49147
  10 0817e82b 55e6a367 ed6fa633 f09d116d c8c53a2c aae50091 c037faad 7a84120d
  11 17142334 0817e82b 55e6a367 ed6fa633 dd4c7be9 c8c53a2c aae50091 c037faad
  12 fc4f023e 17142334 0817e82b 55e6a367 87bea51a dd4c7be9 c8c53a2c aae50091
  13 be316902 fc4f023e 17142334 0817e82b 65141125 87bea51a dd4c7be9 c8c53a2c
  14 1d80d178 be316902 fc4f023e 17142334 4545f53a 65141125 87bea51a dd4c7be9
  15 9f341a45 1d80d178 be316902 fc4f023e 6a61c411 4545f53a 65141125 87bea51a
  16 0f324db9 9f341a45 1d80d178 be316902 06c80d6a 6a61c411 4545f53a 65141125
  17 ffe7012b 0f324db9 9f341a45 1d80d178 b7b601f4 06c80d6a 6a61c411 4545f53a
  18 62932ab8 ffe7012b 0f324db9 9f341a45 763b627a b7b601f4 06c80d6a 6a61c411
  19 5207d867 62932ab8 ffe7012b 0f324db9 7fbba936 763b627a b7b601f4 06c80d6a
  20 07d55ccb 5207d867 62932ab8 ffe7012b 9ba5a6ea 7fbba936 763b627a b7b601f4
  21 dece98a4 07d55ccb 5207d867 62932ab8 293ffb5d 9ba5a6ea 7fbba936 763b627a
  22 e62a812e dece98a4 07d55ccb 5207d867 28fe0fd9 293ffb5d 9ba5a6ea 7fbba936
  23 57206fb8 e62a812e dece98a4 07d55ccb c76084ea 28fe0fd9 293ffb5d 9ba5a6ea
  24 6a6abcf0 57206fb8 e62a812e dece98a4 b2614c5e c76084ea 28fe0fd9 293ffb5d
  25 937514f0 6a6abcf0 57206fb8 e62a812e b42ec21c b2614c5e c76084ea 28fe0fd9
  26 82af3ffb 937514f0 6a6abcf0 57206fb8 be6f6760 b42ec21c b2614c5e c76084ea
  27 eca3bcd5 82af3ffb 937514f0 6a6abcf0 1dccbb10 be6f6760 b42ec21c b2614c5e
  28 2d1576c4 eca3bcd5 82af3ffb 937514f0 01641929 1dccbb10 be6f6760 b42ec21c
```

29 fe3c8658 2d1576c4 eca3bcd5 82af3ffb fc4b36c5 01641929 1dccbb10 be6f6760

30 0d7cce07 fe3c8658 2d1576c4 eca3bcd5 a4a4a3a4 fc4b36c5 01641929 1dccbb10

31 cce1951d 0d7cce07 fe3c8658 2d1576c4 4be9475c a4a4a3a4 fc4b36c5 01641929

32 09b76257 cce1951d 0d7cce07 fe3c8658 0ccddd86 4be9475c a4a4a3a4 fc4b36c5

33 f827767e 09b76257 cce1951d 0d7cce07 db116db7 0ccddd86 4be9475c a4a4a3a4

34 e4a0bb48 f827767e 09b76257 cce1951d 994e2bac db116db7 0ccddd86 4be9475c

35 d8bb1041 e4a0bb48 f827767e 09b76257 5b730abb 994e2bac db116db7 0ccddd86

36 2a2e32f4 d8bb1041 e4a0bb48 f827767e 22e15c59 5b730abb 994e2bac db116db7

37 0d275ca8 2a2e32f4 d8bb1041 e4a0bb48 f6c39382 22e15c59 5b730abb 994e2bac

38 7902369c 0d275ca8 2a2e32f4 d8bb1041 d9f8c2e0 f6c39382 22e15c59 5b730abb

39 f3c80288 7902369c 0d275ca8 2a2e32f4 00e3a7bb d9f8c2e0 f6c39382 22e15c59

40 483bba4d f3c80288 7902369c 0d275ca8 f0a8198c 00e3a7bb d9f8c2e0 f6c39382

41 d75d4d26 483bba4d f3c80288 7902369c fcecdcd4 f0a8198c 00e3a7bb d9f8c2e0

42 0744b618 d75d4d26 483bba4d f3c80288 03186faa fcecdcd4 f0a8198c 00e3a7bb

43 9cce9f01 0744b618 d75d4d26 483bba4d a56f6bbf 03186faa fcecdcd4 f0a8198c

44 a3701bd9 9cce9f01 0744b618 d75d4d26 af1bef5f a56f6bbf 03186faa fcecdcd4

45 131d4c09 a3701bd9 9cce9f01 0744b618 ecb77e1b af1bef5f a56f6bbf 03186faa

46 fb3777d9 131d4c09 a3701bd9 9cce9f01 1d601f44 ecb77e1b af1bef5f a56f6bbf

47 847ea00e fb3777d9 131d4c09 a3701bd9 503a7b95 1d601f44 ecb77e1b af1bef5f

48 aaa69347 847ea00e fb3777d9 131d4c09 5eeb9930 503a7b95 1d601f44 ecb77e1b

49 505caf28 aaa69347 847ea00e fb3777d9 ce695893 5eeb9930 503a7b95 1d601f44

50 675e0b02 505caf28 aaa69347 847ea00e c22dd75f ce695893 5eeb9930 503a7b95

51 abd26099 675e0b02 505caf28 aaa69347 1409c3f8 c22dd75f ce695893 5eeb9930

52 0df9857a abd26099 675e0b02 505caf28 2d864d9f 1409c3f8 c22dd75f ce695893

53 308b8799 0df9857a abd26099 675e0b02 02524f02 2d864d9f 1409c3f8 c22dd75f

54 909cc059 308b8799 0df9857a abd26099 6f2a444a 02524f02 2d864d9f 1409c3f8

55 8d25bd94 909cc059 308b8799 0df9857a 1273c622 6f2a444a 02524f02 2d864d9f

56 f32141da 8d25bd94 909cc059 308b8799 1771ed3f 1273c622 6f2a444a 02524f02

57 8ce24395 f32141da 8d25bd94 909cc059 f52f66a6 1771ed3f 1273c622 6f2a444a

58 07bcd846 8ce24395 f32141da 8d25bd94 149db547 f52f66a6 1771ed3f 1273c622

59 622d5e5b 07bcd846 8ce24395 f32141da b6f4c630 149db547 f52f66a6 1771ed3f

60 c693fc7a 622d5e5b 07bcd846 8ce24395 13dfb889 b6f4c630 149db547 f52f66a6

```
61 55d1c760 c693fc7a 622d5e5b 07bcd846 7e730e00 13dfb889 b6f4c630 149db547
62 fd89031b 55d1c760 c693fc7a 622d5e5b 55489ee6 7e730e00 13dfb889 b6f4c630
63 6203de4a fd89031b 55d1c760 c693fc7a 2aedb1b3 55489ee6 7e730e00 13dfb889
```

هشت کلمه‌ی $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ که در زیر آمده خروجی تکرار نهایی تابع گردساز است.

$$Y_0 = \text{c1059ed8} \boxplus \text{6203de4a} = \text{23097d22}$$

$$Y_1 = \text{367cd507} \boxplus \text{fd89031b} = \text{3405d822}$$

$$Y_2 = \text{3070dd17} \boxplus \text{55d1c760} = \text{8642a477}$$

$$Y_3 = \text{f70e5939} \boxplus \text{c693fc7a} = \text{bda255b3}$$

$$Y_4 = \text{ffc00b31} \boxplus \text{2aedb1b3} = \text{2aadbce4}$$

$$Y_5 = \text{68581511} \boxplus \text{55489ee6} = \text{bda0b3f7}$$

$$Y_6 = \text{64f98fa7} \boxplus \text{7e730e00} = \text{e36c9da7}$$

$$Y_7 = \text{befa4fa4} \boxplus \text{13dfb889} = \text{ad25f72d}$$

مقدار درهم، رشته‌ی ۲۲۴ بیتی زیر است.

```
23097d22 3405d822 8642a477 bda255b3 2aadbce4 bda0b3f7 e36c9da7
```

الف-۸-۴ مثال ۴

در این مثال رشته-داده شامل یک رشته‌ای ۱۴ بایتی است، معادل کد ASCII
´message digest´
کد درهم‌ساز، رشته‌ی ۲۲۴ بیتی زیر است.

```
2cb21c83 ae2f004d e7e81c3c 7019cbcb 65b71ab6 56b22d6d 0c39b8eb
```

الف-۸-۵ مثال ۵

در این مثال رشته-داده شامل یک رشته‌ای ۶۲ بایتی است، معادل کد ASCII

´ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789´
کد درهم‌ساز، رشته‌ی ۲۲۴ بیتی زیر است.

```
bff72b4f cb7d75e5 632900ac 5f90d219 e05e97a7 bde72e74 0db393d9
```

الف-۸-۶ مثال ۶

در این مثال رشته-داده شامل یک رشته‌ای ۸۰ بایتی است، معادل کد ASCII از هشت تکرار
‘1234567890’

رشته‌ی ۲۲۴ بیتی زیر، کد درهم‌ساز را نشان می دهد.

```
b50aecbe 4e9bb0b5 7bc5f3ae 760a8e01 db24f203 fb3cdcd1 3148046e
```

الف-۸-۷ مثال ۷

در این مثال رشته-داده شامل یک رشته‌ای ۵۶ بایتی است، معادل کد ASCII

´abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq´

پس از فرآیند لایه گذاری، دو بلوک ۱۶ کلمه‌ای که از رشته‌ی داده به‌دست می‌آید به‌صورت زیر است.

```
61626364 62636465 63646566 64656667 65666768 66676869 6768696a 68696a6b
696a6b6c 6a6b6c6d 6b6c6d6e 6c6d6e6f 6d6e6f70 6e6f7071 80000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 000001c0
```

در ادامه (نمایش در مبنای شانزده) مقادیر متوالی متغیرهای $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ در اولین بلوک فرآیند آمده است.

```
init: c1059ed8 367cd507 3070dd17 f70e5939 ffc00b31 68581511 64f98fa7 befa4fa4
   0 0e96b2be c1059ed8 367cd507 3070dd17 04342242 ffc00b31 68581511 64f98fa7
   1 51d17d7b 0e96b2be c1059ed8 367cd507 2f8ea3d4 04342242 ffc00b31 68581511
   2 ff1cbd7f 51d17d7b 0e96b2be c1059ed8 79a896fa 2f8ea3d4 04342242 ffc00b31
   3 24bcc047 ff1cbd7f 51d17d7b 0e96b2be 1f60795a 79a896fa 2f8ea3d4 04342242
   4 7d56a6ac 24bcc047 ff1cbd7f 51d17d7b de395286 1f60795a 79a896fa 2f8ea3d4
   5 745beb11 7d56a6ac 24bcc047 ff1cbd7f d863d132 de395286 1f60795a 79a896fa
   6 0dd41573 745beb11 7d56a6ac 24bcc047 2e60d323 d863d132 de395286 1f60795a
   7 9a2541fd 0dd41573 745beb11 7d56a6ac 08d2b348 2e60d323 d863d132 de395286
   8 3140e909 9a2541fd 0dd41573 745beb11 95dfd707 08d2b348 2e60d323 d863d132
   9 b2954925 3140e909 9a2541fd 0dd41573 05ef5e3d 95dfd707 08d2b348 2e60d323
  10 b2a874fb b2954925 3140e909 9a2541fd 9dcaf118 05ef5e3d 95dfd707 08d2b348
  11 116ce44d b2a874fb b2954925 3140e909 0e6d566a 9dcaf118 05ef5e3d 95dfd707
  12 5ff9349a 116ce44d b2a874fb b2954925 08eb3305 0e6d566a 9dcaf118 05ef5e3d
  13 7fa9d65d 5ff9349a 116ce44d b2a874fb 4657cf17 08eb3305 0e6d566a 9dcaf118
  14 006b1b16 7fa9d65d 5ff9349a 116ce44d 08d09e8d 4657cf17 08eb3305 0e6d566a
  15 b301c98a 006b1b16 7fa9d65d 5ff9349a 6fbefa1d 08d09e8d 4657cf17 08eb3305
  16 e623ecc0 b301c98a 006b1b16 7fa9d65d 2b3f859c 6fbefa1d 08d09e8d 4657cf17
```

17 d9244a78 e623ecc0 b301c98a 006b1b16 e66d8d9c 2b3f859c 6fbefa1d 08d09e8d

18 99c72726 d9244a78 e623ecc0 b301c98a b26a409c e66d8d9c 2b3f859c 6fbefa1d

19 ab0cbed2 99c72726 d9244a78 e623ecc0 010d7c65 b26a409c e66d8d9c 2b3f859c

20 78062878 ab0cbed2 99c72726 d9244a78 5678a949 010d7c65 b26a409c e66d8d9c

21 d7c5c5d5 78062878 ab0cbed2 99c72726 b280360c 5678a949 010d7c65 b26a409c

22 bad2ee72 d7c5c5d5 78062878 ab0cbed2 0d4cd0c4 b280360c 5678a949 010d7c65

23 bcf47346 bad2ee72 d7c5c5d5 78062878 d6a19dc8 0d4cd0c4 b280360c 5678a949

24 5ecc417b bcf47346 bad2ee72 d7c5c5d5 3337a11c d6a19dc8 0d4cd0c4 b280360c

25 e15bfa57 5ecc417b bcf47346 bad2ee72 0ce15173 3337a11c d6a19dc8 0d4cd0c4

26 fae6167b e15bfa57 5ecc417b bcf47346 73dbe5c7 0ce15173 3337a11c d6a19dc8

27 991c3f99 fae6167b e15bfa57 5ecc417b 8602a31f 73dbe5c7 0ce15173 3337a11c

28 7055843b 991c3f99 fae6167b e15bfa57 eb4de5f8 8602a31f 73dbe5c7 0ce15173

29 08dcfb6d 7055843b 991c3f99 fae6167b 4606d126 eb4de5f8 8602a31f 73dbe5c7

30 2964b340 08dcfb6d 7055843b 991c3f99 213b3e63 4606d126 eb4de5f8 8602a31f

31 5b3677d0 2964b340 08dcfb6d 7055843b c9689cb0 213b3e63 4606d126 eb4de5f8

32 1ee0fe7d 5b3677d0 2964b340 08dcfb6d 14318a4d c9689cb0 213b3e63 4606d126

33 6b918d6e 1ee0fe7d 5b3677d0 2964b340 216054a8 14318a4d c9689cb0 213b3e63

34 a6710d0d 6b918d6e 1ee0fe7d 5b3677d0 bc823a58 216054a8 14318a4d c9689cb0

35 5e198fed a6710d0d 6b918d6e 1ee0fe7d c49933fe bc823a58 216054a8 14318a4d

36 136c320a 5e198fed a6710d0d 6b918d6e 75687ccb c49933fe bc823a58 216054a8

37 40ee0c43 136c320a 5e198fed a6710d0d f1c2caf6 75687ccb c49933fe bc823a58

38 aa96d78c 40ee0c43 136c320a 5e198fed f48b4ceb f1c2caf6 75687ccb c49933fe

39 27c97b86 aa96d78c 40ee0c43 136c320a b556216a f48b4ceb f1c2caf6 75687ccb

40 b07bd327 27c97b86 aa96d78c 40ee0c43 30ec2d76 b556216a f48b4ceb f1c2caf6

41 d88d56bd b07bd327 27c97b86 aa96d78c dc2fa5a4 30ec2d76 b556216a f48b4ceb

42 5c775077 d88d56bd b07bd327 27c97b86 5fad6db5 dc2fa5a4 30ec2d76 b556216a

43 1526cca3 5c775077 d88d56bd b07bd327 da8a0b1c 5fad6db5 dc2fa5a4 30ec2d76

44 c09dda14 1526cca3 5c775077 d88d56bd d98ec23a da8a0b1c 5fad6db5 dc2fa5a4

45 f885e124 c09dda14 1526cca3 5c775077 e4f23e41 d98ec23a da8a0b1c 5fad6db5

46 5447f0ad f885e124 c09dda14 1526cca3 bfb7497c e4f23e41 d98ec23a da8a0b1c

47 e6227061 5447f0ad f885e124 c09dda14 5b09619b bfb7497c e4f23e41 d98ec23a

48 009cebea e6227061 5447f0ad f885e124 59ecab46 5b09619b bfb7497c e4f23e41

```
49 92b0d169 009cebea e6227061 5447f0ad 9a572b85 59ecab46 5b09619b bfb7497c
50 8d224e54 92b0d169 009cebea e6227061 32144602 9a572b85 59ecab46 5b09619b
51 c1fcac71 8d224e54 92b0d169 009cebea 4e98a8b7 32144602 9a572b85 59ecab46
52 8e6ce843 c1fcac71 8d224e54 92b0d169 2c1823be 4e98a8b7 32144602 9a572b85
53 000f54de 8e6ce843 c1fcac71 8d224e54 f32cf2a8 2c1823be 4e98a8b7 32144602
54 2fe2af3a 000f54de 8e6ce843 c1fcac71 20f763ee f32cf2a8 2c1823be 4e98a8b7
55 1fd539af 2fe2af3a 000f54de 8e6ce843 5acdbd62 20f763ee f32cf2a8 2c1823be
56 7f86644e 1fd539af 2fe2af3a 000f54de 9fc10216 5acdbd62 20f763ee f32cf2a8
57 0e08dc77 7f86644e 1fd539af 2fe2af3a 2a4ea749 9fc10216 5acdbd62 20f763ee
58 0b9f4851 0e08dc77 7f86644e 1fd539af 18b1dfb9 2a4ea749 9fc10216 5acdbd62
59 dbce97c3 0b9f4851 0e08dc77 7f86644e 6ec6ba5b 18b1dfb9 2a4ea749 9fc10216
60 3cd78fe1 dbce97c3 0b9f4851 0e08dc77 3e1ca2f1 6ec6ba5b 18b1dfb9 2a4ea749
61 35f4bf1c 3cd78fe1 dbce97c3 0b9f4851 ba1a8a1b 3e1ca2f1 6ec6ba5b 18b1dfb9
62 86795a7d 35f4bf1c 3cd78fe1 dbce97c3 2ce11258 ba1a8a1b 3e1ca2f1 6ec6ba5b
63 c14b4785 86795a7d 35f4bf1c 3cd78fe1 1108ac7f 2ce11258 ba1a8a1b 3e1ca2f1
```

هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ که در زیر آمده خروجی تابع گردساز در اولین بلوک فرآیند را نشان می‌دهد.

$$Y_0 = c1059ed8 \boxplus c14b4785 = 8250e65d$$

$$Y_1 = 367cd507 \boxplus 86795a7d = bcf62f84$$

$$Y_2 = 3070dd17 \boxplus 35f4bf1c = 66659c33$$

$$Y_3 = f70e5939 \boxplus 3cd78fe1 = 33e5e91a$$

$$Y_4 = ffc00b31 \boxplus 1108ac7f = 10c8b7b0$$

$$Y_5 = 68581511 \boxplus 2ce11258 = 95392769$$

$$Y_6 = 64f98fa7 \boxplus ba1a8a1b = 1f1419c2$$

$$Y_7 = befa4fa4 \boxplus 3e1ca2f1 = fd16f295$$

در ادامه (نمایش مبنای شانزده از) مقادیر متوالی متغیرهای $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ پس از دومین بلوک فرآیند آمده است.

```
init: 8250e65d bcf62f84 66659c33 33e5e91a 10c8b7b0 95392769 1f1419c2 fd16f295
   0 692e407d 8250e65d bcf62f84 66659c33 e4be1e69 10c8b7b0 95392769 1f1419c2
   1 608d83e1 692e407d 8250e65d bcf62f84 3ddb8cee e4be1e69 10c8b7b0 95392769
   2 09bfa89f 608d83e1 692e407d 8250e65d f5813490 3ddb8cee e4be1e69 10c8b7b0
   3 2375fbc5 09bfa89f 608d83e1 692e407d c3e18529 f5813490 3ddb8cee e4be1e69
```

 4 717e79e7 2375fbc5 09bfa89f 608d83e1 77d39ccc c3e18529 f5813490 3ddb8cee

 5 a9319748 717e79e7 2375fbc5 09bfa89f fdbb9913 77d39ccc c3e18529 f5813490

 6 27a42f04 a9319748 717e79e7 2375fbc5 b999cce4 fdbb9913 77d39ccc c3e18529

 7 3419081e 27a42f04 a9319748 717e79e7 54e69e21 b999cce4 fdbb9913 77d39ccc

 8 0ab393c2 3419081e 27a42f04 a9319748 ad29647e 54e69e21 b999cce4 fdbb9913

 9 006784eb 0ab393c2 3419081e 27a42f04 aff457e7 ad29647e 54e69e21 b999cce4

10 ecd5c9db 006784eb 0ab393c2 3419081e 9af42a0e aff457e7 ad29647e 54e69e21

11 4762e8f0 ecd5c9db 006784eb 0ab393c2 8fb6f3d8 9af42a0e aff457e7 ad29647e

12 af93b2a8 4762e8f0 ecd5c9db 006784eb 97e63d39 8fb6f3d8 9af42a0e aff457e7

13 533c517c af93b2a8 4762e8f0 ecd5c9db 7364bae6 97e63d39 8fb6f3d8 9af42a0e

14 03c0a51b 533c517c af93b2a8 4762e8f0 3afb010d 7364bae6 97e63d39 8fb6f3d8

15 5fd065bd 03c0a51b 533c517c af93b2a8 b8e64229 3afb010d 7364bae6 97e63d39

16 18b268b5 5fd065bd 03c0a51b 533c517c 38eda38d b8e64229 3afb010d 7364bae6

17 b87d63b4 18b268b5 5fd065bd 03c0a51b 25c2c397 38eda38d b8e64229 3afb010d

18 b1d846e0 b87d63b4 18b268b5 5fd065bd d674405f 25c2c397 38eda38d b8e64229

19 8ba0aed6 b1d846e0 b87d63b4 18b268b5 b8109422 d674405f 25c2c397 38eda38d

20 1485f843 8ba0aed6 b1d846e0 b87d63b4 1c58cd66 b8109422 d674405f 25c2c397

21 238f4cda 1485f843 8ba0aed6 b1d846e0 39b2eb5f 1c58cd66 b8109422 d674405f

22 7031b061 238f4cda 1485f843 8ba0aed6 4b8262ad 39b2eb5f 1c58cd66 b8109422

23 d4e7ec62 7031b061 238f4cda 1485f843 163c3aa0 4b8262ad 39b2eb5f 1c58cd66

24 66582df3 d4e7ec62 7031b061 238f4cda c0976260 163c3aa0 4b8262ad 39b2eb5f

25 dedb8199 66582df3 d4e7ec62 7031b061 b73e2dec c0976260 163c3aa0 4b8262ad

26 f8536917 dedb8199 66582df3 d4e7ec62 7c2af9c4 b73e2dec c0976260 163c3aa0

27 d7333b8a f8536917 dedb8199 66582df3 b2b0b71a 7c2af9c4 b73e2dec c0976260

28 760847c1 d7333b8a f8536917 dedb8199 5898eff2 b2b0b71a 7c2af9c4 b73e2dec

29 7eabc6d7 760847c1 d7333b8a f8536917 24dd3883 5898eff2 b2b0b71a 7c2af9c4

30 90c49624 7eabc6d7 760847c1 d7333b8a cce25e67 24dd3883 5898eff2 b2b0b71a

31 0b876264 90c49624 7eabc6d7 760847c1 e4e4a53b cce25e67 24dd3883 5898eff2

32 04cb36c0 0b876264 90c49624 7eabc6d7 5403a391 e4e4a53b cce25e67 24dd3883

33 d58cc34a 04cb36c0 0b876264 90c49624 b78767c3 5403a391 e4e4a53b cce25e67

34 0ed14dd7 d58cc34a 04cb36c0 0b876264 fdcdc9d9 b78767c3 5403a391 e4e4a53b

35 5a89a942 0ed14dd7 d58cc34a 04cb36c0 790c4a20 fdcdc9d9 b78767c3 5403a391

```
36 4d30424c 5a89a942 0ed14dd7 d58cc34a f95bf853 790c4a20 fdcdc9d9 b78767c3
37 47f58c5c 4d30424c 5a89a942 0ed14dd7 0ec9be3b f95bf853 790c4a20 fdcdc9d9
38 b5ad85d7 47f58c5c 4d30424c 5a89a942 cf9f1dbe 0ec9be3b f95bf853 790c4a20
39 762fecbc b5ad85d7 47f58c5c 4d30424c 15427ed3 cf9f1dbe 0ec9be3b f95bf853
40 32abe746 762fecbc b5ad85d7 47f58c5c 4053e12e 15427ed3 cf9f1dbe 0ec9be3b
41 84adb2a0 32abe746 762fecbc b5ad85d7 7cece4e2 4053e12e 15427ed3 cf9f1dbe
42 c6e1c5af 84adb2a0 32abe746 762fecbc 42f9990b 7cece4e2 4053e12e 15427ed3
43 35e14bfa c6e1c5af 84adb2a0 32abe746 c9965792 42f9990b 7cece4e2 4053e12e
44 7410bfd8 35e14bfa c6e1c5af 84adb2a0 ca54ce51 c9965792 42f9990b 7cece4e2
45 3fe9e763 7410bfd8 35e14bfa c6e1c5af ae7cdb66 ca54ce51 c9965792 42f9990b
46 853c3a00 3fe9e763 7410bfd8 35e14bfa c2be054d ae7cdb66 ca54ce51 c9965792
47 f7d035e7 853c3a00 3fe9e763 7410bfd8 f6d59d2c c2be054d ae7cdb66 ca54ce51
48 20bae2b8 f7d035e7 853c3a00 3fe9e763 cab73f06 f6d59d2c c2be054d ae7cdb66
49 ae6bf667 20bae2b8 f7d035e7 853c3a00 52384d2f cab73f06 f6d59d2c c2be054d
50 12e504e5 ae6bf667 20bae2b8 f7d035e7 f9a8377f 52384d2f cab73f06 f6d59d2c
51 f3497054 12e504e5 ae6bf667 20bae2b8 d0ab7cfc f9a8377f 52384d2f cab73f06
52 9f166cdb f3497054 12e504e5 ae6bf667 71b3459b d0ab7cfc f9a8377f 52384d2f
53 ccd8fa44 9f166cdb f3497054 12e504e5 0f557ddd 71b3459b d0ab7cfc f9a8377f
54 f5e664bd ccd8fa44 9f166cdb f3497054 a679a5e9 0f557ddd 71b3459b d0ab7cfc
55 d4ea8c7e f5e664bd ccd8fa44 9f166cdb 2958ce2a a679a5e9 0f557ddd 71b3459b
56 e8c8fec7 d4ea8c7e f5e664bd ccd8fa44 35f6800e 2958ce2a a679a5e9 0f557ddd
57 882ed69e e8c8fec7 d4ea8c7e f5e664bd 30267d8e 35f6800e 2958ce2a a679a5e9
58 4ec725f6 882ed69e e8c8fec7 d4ea8c7e ce1d1ce4 30267d8e 35f6800e 2958ce2a
59 5c9cfc69 4ec725f6 882ed69e e8c8fec7 c8242b92 ce1d1ce4 30267d8e 35f6800e
60 c9a31836 5c9cfc69 4ec725f6 882ed69e 9e40a370 c8242b92 ce1d1ce4 30267d8e
61 f754c16e c9a31836 5c9cfc69 4ec725f6 333e0b63 9e40a370 c8242b92 ce1d1ce4
62 94314748 f754c16e c9a31836 5c9cfc69 1fbc63b0 333e0b63 9e40a370 c8242b92
63 f2e7a4b9 94314748 f754c16e c9a31836 9ffd8dac 1fbc63b0 333e0b63 9e40a370
```

هشت کلمه $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ که در زیر آمده خروجی تکرار کننده نهایی تابع گردساز را نشان می‌دهد.

$$Y_0 = 8250e65d \boxplus f2e7a4b9 = 75388b16$$

Y_1 = bcf62f84 ⊎ 94314748 = 512776cc

Y_2 = 66659c33 ⊎ f754c16e = 5dba5da1

Y_3 = 33e5e91a ⊎ c9a31836 = fd890150

Y_4 = 10c8b7b0 ⊎ 9ffd8dac = b0c6455c

Y_5 = 95392769 ⊎ 1fbc63b0 = b4f58b19

Y_6 = 1f1419c2 ⊎ 333e0b63 = 52522525

Y_7 = fd16f295 ⊎ 9e40a370 = 635651e5

مقدار درهم‌ساز، رشته‌ی ۲۲۴ بیتی زیر است.

```
75388b16 512776cc 5dba5da1 fd890150 b0c6455c b4f58b19 52522525
```

الف-۸-۸ مثال ۸

در این مثال رشته-داده شامل رشته‌ای ۱۰۰۰۰۰۰ بایتی است، معادل کد ASCII حرف ´a´ که برای $۱۰^۶$ بار تکرار می‌شود.

کد درهم‌ساز رشته‌ی ۲۲۴ بیتی زیر است.

```
20794655 980c91d8 bbb4c1ea 97618a4b f03f4258 1948b2ee 4ee7ad67
```

الف-۸-۹ مثال ۹

در این مثال رشته‌ی داده شامل یک تک‌بیت ´0´ است.

کد درهم‌ساز رشته‌ی ۲۲۴ بیتی زیر است.

```
d3fe57cb 76cdd24e 9eb23e7e 15684e03 9c75459b eaae100f 89712e9d
```

الف-۸-۱۰ مثال ۱۰

در این مثال رشته‌ی داده شامل یک تک‌بیت ´1´ است.

کد درهم‌ساز رشته‌ی ۲۲۴ بیتی زیر است.

```
0d05096b ca2a4a77 a2b47a05 a59618d0 1174b378 92376135 c1b6e957
```

الف-۸-۱۱ مثال ۱۱

در این مثال رشته‌ی داده شامل ۱۰۱ بیت به‌صورت 01..-1010101 است.

کد درهم‌ساز رشته‌ی ۲۲۴ بیتی زیر است.

```
2b1d4a34 155c04d7 a51065d6 a4476203 9a38dffd 73e76b17 b043555c
```

الف-۸-۱۲ مثال ۱۲

در این مثال رشته‌ی داده شامل ۲۵۶ هشت‌تایی به‌صورت FE FF … 03 02 01 00 است.
کد درهم‌ساز رشته‌ی ۲۲۴ بیتی زیر است.

```
88702e63 237824c4 eb0d0fcf e41469a4 62493e8b eb2a75bb e5981734
```

الف-۸-۱۳ مثال ۱۳

در این مثال رشته‌ی داده H_0 است که شامل ۲۲۴ بیت می شود. برای۱ =i تا ۱۰۰ =i، H_i را برابر با کد درهم‌ساز H_{i-1} قرار می‌دهیم.
کد درهم‌ساز H_{100} رشته‌ی ۲۲۴ بیتی زیر است.

```
a0884cc1 a335042b fe452bf4 6777ed20 217a3472 81dc389e 7b1fbfee
```

الف-۹ بردارهای آزمون کامل برای توابع درهم‌ساز اختصاصی ۴، ۵، ۶ و ۸

توابع درهم‌ساز اختصاصی SHA-256، SHA-384، SHA-512 و SHA-224 به عنوان بخشی از این استاندارد شرح داده شد. پیوست‌های الف-۴، الف-۵، الف-۶ و الف-۸ به ترتیب بردارهای آزمون نمونه را برای این چهار تابع درهم‌ساز ارائه می‌دهد. یک ضعف مهم این مثال‌ها این است که مقادیر ورودی، ترکیب انحصاری از کاراکترهای حروفی عددی کد اسکی هستند. این ضمیمه مجموعه بردارهای آزمون کاملتری برای این توابع درهم‌ساز در بر دارد.

انتخاب بردار آزمون بر پایه ملاحظات زیر است:

۱- ورودی‌های با طول ۱ تا ۲۵۶ (برای SHA-224 و SHA-256) یا ۱۰۲۴ (برای SHA-384 و SHA-512) به منظور آزمایش برنامه لایه‌گذاری تعریف شده اند. (مثال‌ها در پیوست الف-۴، الف-۶ و الف-۸ فقط پیام‌هایی که طول آنها مضرب ۸ است را در بر دارند.) تعداد کمی از بردارهای با طول بیشتر در بر گرفته شده است.
۲- اطمینان حاصل شده که همه‌ی کلمات۳۲ بیتی (SHA-224 و SHA-256) یا همه ۶۴ بیتی (SHA-384 و SHA-512) با وزن همینگ ۱ حداقل یک‌بار به‌عنوان بخشی از ورودی اتفاق افتاده است. این کار به منظور آزمودن توابع بسط پیام انجام شده است.
۳- راه حل اطمینان از سرریز رقم نقلی از یک بایت به بایت دیگر این است که یکی از موارد زیر حداقل یک‌بار اتفاق افتد:

الف- برای SHA-224 و SHA-256

i. 0xFFFFFFFF + 0x00000001

ii. 0xFFFF0000 + 0x00010000

iii. 0x0000FFFF + 0x00000001

iv. 0xFF00FF00 + 0x01000100

v. 0x00FF00FF + 0x00010001

ب- برای SHA-384 و SHA-512

i. 0xFFFFFFFFFFFFFFFF + 0x0000000000000001

ii. 0xFFFFFFFF00000000 + 0x0000000100000000

iii. 0x00000000FFFFFFFF + 0x0000000000000001

iv. 0xFFFF0000FFFF0000 + 0x0001000000010000

v. 0x0000FFFF0000FFFF + 0x0000000100000001

vi. 0xFF00FF00FF00FF00 + 0x0100010001000100

vii. 0x00FF00FF00FF00FF + 0x0001000100010001

فهرست کامل بردارهای آزمون در نشانی وب زیر قابل دسترسی است:

http://www.iaik.tu-graz.ac.at/research/sha2_testvectors.zip.

یادآوری- کلیه‌ی بردارهای آزمون برای توابع درهم‌ساز باقیمانده‌ای که در این استاندارد آمده است در نشانی وب فوق قابل دسترسی نیست. تجزیه و تحلیل‌ها همچنان برای تعیین اینکه آیا بردارهای آزمون بیشتری برای توابع درهم‌ساز باقیمانده مورد نیاز خواهد بود یا نه، و اگر هست چه ملاحظاتی لازم است، ادامه دارد. اگر بردارهای آزمون بیشتری مورد نیاز باشد، ممکن است یک ضمیمه‌ی دوم برای این استاندارد در نظر گرفته شود.

پیوست ب

(اطلاعاتی)

مشخصات رسمی

ب-۰ مقدمه

در این قسمت مشخصات کامل توابع درهم‌ساز اختصاصی ۲،۱ و ۳ با زبان مشخصات[1] Z قرار دارد. علایم لازم برای Z به صورت توصیف شده در [۱] است.

مشخصات Z بسیاری از نام‌گذاری‌ها، ساختارها و سایر مواردی را که در بدنه‌ی اصلی این استاندارد استفاده شده در خود نگه داشته است.

مشخصات Z، از جمله در مشخصات، فقط Z نوشته می‌شود. توضیحات به بخش‌های اصلی متن این استاندارد که Z از آن مشتق شده، اشاره دارد.

مشخصات Z یک پیام را به‌صورت دنباله‌ای اعداد طبیعی ۰ و ۱ (رشته) مدل می‌کند.

ب-۱ مشخصات تابع درهم‌ساز اختصاصی ۱

#۳ اصطلاحات و تعاریف

تابع گردساز

$Bit == \{0, 1\}$
$String == seq\ Bit$

$L_1: \mathbb{N}$
$L_2: \mathbb{N}_1$

$String\_L_1 == \{s: String \mid \#\ s = L_1\}$
$String\_L_2 == \{s: String \mid \#\ s = L_2\}$
$\mid \Phi: String\_L_1 \times String\_L_2 \rightarrow String\_L_2$

کلمه

$Word == \{w: String \mid \#\ w = 32\}$
$Word\_capacity == 2 \uparrow 32$
$Word\_capacity\_m\_1 = = (2 \uparrow 32) - 1$
$IWord == 0\ ..\ Word\_capacity\_m\_1$

#۴ نمادها (و اصطلاحات کوته‌نوشت)

$S^n()$ تنها لازم است S را به صورت S^n (تکرار رابطه[2])، همانگونه که در Z بیان شده، بیان نماید.

$S: Word \rightarrow Word$

[1] - Specification Language

[2] -Relation iteration

```
∀ A : Word •
    (let I == W _to _ I(A) •
     (let shift _I == (I*2 + ( I div (2 ↑ 31 ))) mod (2 ↑ 32 ) •
          S(A) = I _ to _ W (shift _I)))
```

۸ ۷ ⊕ این علایم برای کلمات تعریف شده‌اند و تمام آنچه مورد نیاز است، هستند.

```
BO == Bit × Bit → Bit
```

```
LO : Word × Word × BO→ Word
∀ p,q : Word;bo:BO •
   LO(p,q,bo) = { n:1-.# p  • n  bo(p(n),q(n)))}
```

```
_xor _, _ or _, _ and _ : BO
        0 xor 1 = 1
        0 xor 0 = 0
        1 xor 0 = 1
        1 xor 1 = 0

        0 or 1 = 1
        0 or 0 = 0
        1 or 0 = 1
        1 or 1= 1

        0 and 1= 0
        0 and 0 = 1
        1 and 0 = 0
        1 and 1 = 1
```

```
_ XOR _, _ OR _, _ AND _ : Word × Word → Word
   ∀ A, B : Word •
        A XOR B = LO(A, B, (_xor_)) ∧
        A OR B = LO(A, B, (_or_)) ∧
        A AND B = LO(A, B, (_and_))
```

¬

```
NOT : Word → Word
∀ A : Word •
    NOT A = A XOR { n:1-.# A • n  1}
```

⊞

$$\_ \text{AND} \_ : \text{Word} \times \text{Word} \rightarrow \text{Word}$$

$$\forall A, B : \text{Word} \bullet A \ \ B = \text{I\_to\_W}((\text{W\_to\_I}(A) + \text{W\_to\_I}(B)) \bmod \text{Word\_capacity})$$

 # ۵ الزامات

 # ۶ مدلی برای تابع درهم‌ساز اختصاصی

$$L_H : \mathbb{N}_1$$

$$L_H \leq L_2$$

$$\text{Byte} == \{b: \text{String} \mid \#b = 8\}$$

$$\text{IByte} == 0 \mathinner{..} 255$$

$$\text{B\_to\_I} : \text{Byte} \rightarrow \text{IByte}$$

$$\forall x : \text{Byte} \bullet \text{B\_to\_I}(x) = x(1) * 2 \uparrow 7 + x(2) * 2 \uparrow 6 + x(3) * 2 \uparrow 5 + x(4) * 2 \uparrow 4 + x(5) * 2 \uparrow 3 + x(6) * 2 \uparrow 2 + x(7) * 2 + x(8)$$

عملیات درهم‌سازی

$$IV : \text{String\_}L_2$$

$$\text{Maximum\_Length\_of\_String} : \mathbb{N}$$

$$\text{Hash} : \text{String} \nrightarrow \text{String\_}L_H$$

$$\forall D : \text{String} \mid \#D \leq \text{Maximum\_Length\_of\_String} \bullet \text{Hash}(D) = (\textbf{let } PD == \text{pad}(D) \bullet (\textbf{let } SD == \text{split}(PD) \bullet (\textbf{let } H_q == \text{iterate}(SD, IV) \bullet \text{truncate}(H_q))))$$

مرحله‌ی ۱ (لایه‌گذاری)

$$\text{StringMultiple\_}L_1 == \{s : \text{String} \mid \#s \bmod L_1 = 0\}$$

$$\text{pad} : \text{String} \rightarrow \text{StringMultiple\_}L_1$$

مرحله‌ی ۲ (تقسیم)

$$\text{StringBlocks} == \text{seq String\_}L_1$$

Split : StringMultiple_L_1 → StringBlocks

Split = { sml1 : StringMultiple_L_1 ; sb : StringBlocks | sml1 = □/ sb • sml1 ↦ sb}

مرحله‌ی ۳ (تکرار)

iterate : StringBlocks × String_L_2 ⇸ String_L_2

∀ sb : StringBlocks; H_{i-1} : String_L_2 | # sb ≥ 1 •
Iterate(sb,H_{i-1}) =
(**let** D_i == sb(1) •
(**let** H_i == Φ(D_i,H_{i-1}) •
if # sb = 1
then H_i
else iterate(tail sb,H_i)))

مرحله‌ی ۴ (کوتاه سازی)

String_L_H == {s : String | #s = L_H}

truncate : String_L_2 → String_L_H

∀ sy : String_L_2 •
truncate(sy) = (1 …L_H) ⅂ sy

#۷ تابع درهم‌ساز اختصاصی ۱

Maximum_Length_of_String = (2 ↑ 64)-1

 # ۷-۱ پارامتر، توابع و ثابت‌ها

 # ۷-۱-۱ پارامترها

L_1 = 512
L_2 = 160
L_3 = 160

 # ۷-۱-۲ قاعده‌ی مرتب سازی بایت

W_to_I : Word → IWord

∀ w : Word •
W_to_I(w) =
(**let** B_0 == B_to_I((1 … 8) ⅂ w) •
(**let** B_1 == B_to_I((9 … 16) ⅂ w) •
(**let** B_2 == B_to_I((17 … 24) ⅂ w) •
(**let** B_3 == B_to_I((25 … 32) ⅂ w) •
B_3 * 2 ↑ 24 + B_2 * 2 ↑ 16 + B_1 * 2 ↑ 8 + B_0))))

$$I\_to\_W : IWord \rightarrow Word$$
$$I\_to\_W = W\_to\_I^{\sim}$$

۷-۱-۳ توابع

$$Indexed\_g == \{g : seq (Word \times Word \times Word \rightarrow Word) \mid \# g = 80\}$$

$$g : Indexed\_g$$
$$\forall X_0, X_1, X_2 : Word \bullet$$
$$(\forall i : 1 .. 16 \bullet g(i)(X_0, X_1, X_2) = X_0 \text{ XOR } X_1 \text{ XOR } X_2) \square$$
$$(\forall i : 17 .. 32 \bullet g(i)(X_0, X_1, X_2) = (X_0 \text{ AND } X_1) \text{ OR } (\text{NOT } X_0 \text{ AND } X_2)) \square$$
$$(\forall i : 33 .. 48 \bullet g(i)(X_0, X_1, X_2) = (X_0 \text{ OR NOT } X\_1) \text{ XOR } X\_2) \square$$
$$(\forall i : 49 .. 64 \bullet g(i)(X_0, X_1, X_2) = (X_0 \text{ AND } X_2) \text{ OR } (X_1 \text{ AND NOT } X_2)) \square$$
$$(\forall i : 65 .. 80 \bullet g(i)(X_0, X_1, X_2) = X_0 \text{ XOR } (X_1 \text{ OR NOT } X_2))$$

۷-۱-۴ ثابت‌ها

```
x00000000 == 0

x5A827999 == 1518500249

x6ED9EBA1 == 1859775393

x8F1BBCDC == 2400959708

xA953FD4E == 2840853838

x50A28BE6 == 1352829926

x5C4DD124 == 1548603684

x6D703EF3 == 1836072691

x7A6D76E9 == 2053994217
```

$$Constants == \{c : StringWord \mid \# c = 80\}$$

$$C, C\square : Constants$$

```
(∀ i : 1 .. 16 •
      C(i) = I_to_W(x00000000)) □
(∀ i : 17 .. 32 •
      C(i) = I_to_W(x5A827999)) □
(∀ i : 33 .. 48 •
      C(i) = I_to_W(x6ED9EBA1)) □
(∀ i : 49 .. 64 •
      C(i) = I_to_W(x8F1BBCDC)) □
(∀ i : 65 .. 80 •
      C(i) = I_to_W(XA953FD4E)) □

(∀ i : 1 .. 16 •
      C□ (i) = I_to_W(x50A28BE6)) □
(∀ i : 17 .. 32 •
      C□ (i) = I_to_W(x5C4DD124)) □
(∀ i : 33 .. 48 •
      C□(i) = I_to_W(x6D703EF3)) □
(∀ i : 49 .. 64 •
      C□ (i) = I_to_W(x7A6D76E9)) □
(∀ i : 65 .. 80 •
      C□ (i) = I_to_W(x00000000))
```

t ==

⟨11, 14, 15, 12, 5, 8, 7, 9, 11, 13, 14, 15, 6, 7, 9, 8,
7, 6, 8, 13, 11, 9, 7, 15, 7, 12, 15, 9, 11, 7, 13, 12,
11, 13, 6, 7, 14, 9, 13, 15, 14, 8, 13, 6, 5, 12, 7, 5,
11, 12, 14, 15, 14, 15, 9, 8, 9, 14, 5, 6, 8, 6, 5, 12,
9, 15, 5, 11, 6, 8, 13, 12, 5, 12, 13, 14, 11, 8, 5, 6 ⟩

t′ ==

⟨8, 9, 9, 11, 13, 15, 15, 5, 7, 7, 8, 11, 14, 14, 12, 6,
9, 13, 15, 7, 12, 8, 9, 11, 7, 7, 12, 7, 6, 15, 13, 11
9, 7, 15, 11, 8, 6, 6, 14, 12, 13, 5, 14, 13, 13, 7, 5
15, 5, 8, 11, 14, 14, 6, 14, 6, 9, 12, 9, 12, 5, 15, 8,
8, 5, 12, 9, 12, 5, 14, 6, 8, 13, 6, 5, 15, 13, 11, 11⟩

توجه کنید که مقادیر a و a′ یک واحد بزرگتر از متن الزامی هستند زیرا دنباله‌ها در Z از ۱ آغاز می‌شوند.

a ==

⟨1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
8, 5, 14, 2, 11, 7, 16, 4, 13, 1, 10, 6, 3, 15, 12, 9,
4, 11, 15, 5, 10, 16, 9, 2, 3, 8, 1, 7, 14, 12, 6, 13,
2, 10, 12, 11, 1, 9, 13, 5, 14, 4, 8, 16, 15, 6, 7, 3,
5, 1, 6, 10, 8, 13, 3, 11, 15, 2, 4, 9, 12, 7, 16, 14 ⟩

a' ==

⟨6, 15, 8, 1, 10, 3, 12, 5, 14, 7, 16, 9, 2, 11, 4, 13,
7, 12, 4, 8, 1, 14, 6, 11, 15, 16, 9, 13, 5, 10, 2, 3,
16, 6, 2, 4, 8, 15, 7, 10, 12, 9, 13, 3, 11, 1, 5, 14,
9, 7, 5, 2, 4, 12, 16, 1, 6, 13, 3, 14, 10, 8, 11, 15,
13, 16, 11, 5, 2, 6, 9, 8, 7, 3, 14, 15, 1, 4, 10, 12 ⟩

 # ۷-۱-۵ مقدار اولیه

x67452301 == 1732584193
Y_0 == I_to_W (x67452301)
xEFCDAB89 == 4023233417
Y_1 == I_to_W (xEFCDAB89)
x98BADCFE == 2562383102
Y_2 == I_to_W (x98BADCFE)
x10325476 == 271733878
Y_3 == I_to_W (x10325476)
xC3D2E1F0 == 3285377520
Y_4 == I_to_W (xC3D2E1F0)

IV = Y_0□ Y_1□ Y_2□ Y_3□ Y_4

 # ۷-۲ روش لایه‌گذاری

∀ D : String •
Pad(D) =
(**let** L_D == # D •
(**let** Zeros == {n : 1 … ((447 – L_D) mod 512) • n ↦ 0} •
(**let** Length_D_LSH == I_to_W (L_D mod (2 ↑ 32)) •
(**let** Length_D_MSH == I_to_W (L_D div (2 ↑ 32)) •
D □ ⟨1⟩ □ Zeros □ Length_D_LSH □ Length_D_MSH))))

 # ۷-۳ توصیف تابع گردساز

StringWord == seq Word

Split_String_to_StringWord : String → StringWord
Split_String_to_StringWord =
{s : String; sw : StringWord | s = □/sw • s ↦ sw}

L80:StringWord × StringWord ×
Indexed_g × seq ℕ × seq ℕ ×
Constants × 1 .. 80 → StringWord

```
∀ Z,D : StringWord; g : Indexed_g; t, a : seq ℕ
    C : Constants; i : 1 … 80 | Z = 16 ⍰ # X = 5 •
     L80(Z, X, g, t, a, C, i ) =
     (let X0 == X(1); X1 == X(2); X2 == X(3); X3 == X(4); X4 == X(5) •
     (let W == S^t(i) (X0 ⊞ g(i)(X1, X2, X3) ⊞ Z(a(i)) ⊞ C(i)) ⊞ X4 •
     (let Y == ⟨X4, W, X1, S^10 (X2), X3 ⟩ •
            if (i=80)
            then Y
            else L80(Z,Y, g, t , a, C, i+1))))
```

```
∀ sx : String_L1; sy : String_L2 •
      Φ(sx, sy) =
            (let Z == Split_String_to_StringWord(sx) •
            (let Y == Split_String_to_StringWord(sy) •
            (let X == L80(Z, Y, g, t, a, C, 1) •
            (let X□ == L80(Z,Y, rev g, t□ , a□ , C□ , 1) •
            (let Y0 == Y(2) ⊞ X(3) ⊞ X'(4) •
            (let Y1 == Y(3) ⊞ X(4) ⊞ X'(5) •
            (let Y2 == Y(4) ⊞ X(5) ⊞ X'(1) •
            (let Y3 == Y(5) ⊞ X(1) ⊞ X'(2) •
            (let Y4 == Y(1) ⊞ X(2) ⊞ X'(3) •

                  Y0□ Y1□ Y2□ Y3□ Y4 )))))))))
```

ب-۱-۱ توابع کمکی

```
_ ↑ _ : ℕ × ℕ → ℕ
∀ p : ℕ •
      P ↑ 0 = 1 □
(∀ n: ℕ1 • p ↑ n = p * (p ↑ (n-1)))
```

ب-۲ توصیف تابع درهم‌ساز اختصاصی ۲

بخش‌های #۳، #۴، #۵، #۶ و ب-۱-۱ از پیوست ب-۱ باید در این این بخش از پیوست تکرار شوند.

#۸ تابع درهم‌ساز اختصاصی ۲

```
Maximum_Length_of_String = (2 ↑ 64)-1
```

#۸-۱ پارامتر، توابع و ثابت‌ها

#۸-۱-۱ پارامترها

$L_1 = 512$
$L_2 = 128$

$L_H = 128$

 # ۸-۱-۲ قاعده‌ی مرتب سازی بایت

بخش #۷-۱-۲ از پیوست ب-۱ باید در اینجا تکرار شود.

 # ۸-۱-۳ توابع

بخش #۷-۱-۳ از پیوست ب-۱ باید در اینجا تکرار شود.

g2 == (1…64) 1 g

 # ۸-۱-۴ ثابت‌ها

```
x00000000 == 0

x5A827999 == 1518500249

x6ED9EBA1 == 1859775393

x8F1BBCDC == 2400959708

x50A28BE6 == 1352829926

x5C4DD124 == 1548603684

x6D703EF3 == 1836072691
```

Constants = = {c : StringWord | # c = 64}

C, C□ : Constants

(∀ i : 1 .. 16 •
  C(i) = I_to_W(x00000000)) □
(∀ i : 17 .. 32 •
  C(i) = I_to_W(x5A827999)) □
(∀ i : 33 .. 48 •
  C(i) = I_to_W(x6ED9EBA1)) □
(∀ i : 49 .. 64 •
  C(i) = I_to_W(x8F1BBCDC)) □

(∀ i : 1 .. 16 •
  C□ (i) = I_to_W(x50A28BE6)) □
(∀ i : 17 .. 32 •
  C□ (i) = I_to_W(x5C4DD124)) □
(∀ i : 33 .. 48 •
  C□(i) = I_to_W(x6D703EF3)) □
(∀ i : 49 .. 64 •
  C□ (i) = I_to_W(x00000000))

بخش #۷-۱-۳ از پیوست ب-۱ باید در اینجا تنها برای t، t′، a، a′ تکرار شود.

t2 = = (1 … 64) 1 t
t2′ = = (1 … 64) 1 t′
a2 = = (1 … 64) 1 a
a2′ = = (1 … 64) 1 a′

 # ۸-۱-۵ مقدار اولیه

x67452301 == 1732584193
Y_0 == I_to_W(x67452301)
xEFCDAB89 == 4023233417
Y_1 == I_to_W(xEFCDAB89)
x98BADCFE == 2562383102
Y_2 == I_to_W(x98BADCFE)
x10325476 == 271733878
Y_3 == I_to_W(x10325476)

IV = Y_0□ Y_1□ Y_2□ Y_3□ Y_4

 # ۸-۲ روش لایه‌گذاری

بخش #۷-۲ از پیوست ب-۱ باید در اینجا تکرار شود.

 # ۸-۳ توصیف تابع گردساز

بخش #۷-۳ از پیوست ب-۱ باید در اینجا برای تعاریف StringWord و Split_String_to_StringWord تکرار شود.

L64:StringWord × StringWord ×
Indexed_g × seq ℕ × seq ℕ ×
Constants × 1 .. 64 → StringWord

∀ Z, X : StringWord; g : Indexed_g; t, a : seq ℕ
C : Constants; i : 1 … 64 | Z = 16 ⍰ # X = 5 •
L64(Z, X, g, t, a, C, i) =
(**let** X0 == X(1); X1 == X(2); X2 == X(3); X3 == X(4) •
(**let** W == $S^{t(i)}$ (X0 ⊞ g(i)(X1, X2, X3) ⊞ Z(a(i)) ⊞ C(i))
(**let** Y == ⟨X3, W, X1, X2 ⟩ •
if (i=64)
then Y
else L64(Z,Y, g, t , a, C, i+1))))

∀ sx : $String\_L_1$; sy : $String\_L_2$ •
Φ(sx, sy) =
(**let** Z == Split_String_to_StringWord(sx) •
(**let** Y == Split_String_to_StringWord(sy) •
(**let** X == L64(Z, Y,rev g2, t2⍰, a2⍰, C, 1) •
(**let** X⍰ == L64(Z,Y, rev g2, t2⍰, a2⍰, C⍰, 1) •

(**let** Y0 == Y(2) ⊞ X(3) ⊞ X'(4) •
(**let** Y1 == Y(3) ⊞ X(4) ⊞ X'(1) •
(**let** Y2 == Y(4) ⊞ X(1) ⊞ X'(2) •
(**let** Y3 == Y(1) ⊞ X(2) ⊞ X'(3) •
Y_0 Y_1 Y_2 Y_3))))))))

ب-۳ توصیف تابع درهم‌ساز اختصاصی ۳

بخش‌های #۳، #۴، #۵، #۶، و ب-۱-۱ از پیوست ب-۱ باید در این قسمت از پیوست تکرار شوند.

#۹ تابع درهم‌ساز اختصاصی ۱

Maximum_Length_of_String = (2 ↑ 64)-1

 # ۹-۱ پارامتر، توابع و ثابت‌ها

 # ۹-۱-۱ پارامترها

$L_1 = 512$
$L_2 = 160$
$L_H = 160$

 # ۹-۱-۲ قاعده‌ی مرتب سازی بایت

W_to_I : Word → IWord

∀ w : Word • W_to_I(w)
W_to_I(w) =
(**let** B_0 == B_to_I((1 … 8) ⌈ w) •
(**let** B_1 == B_to_I((9 … 16) ⌈ w) •
(**let** B_2 == B_to_I((17 … 24) ⌈ w) •
(**let** B_3 == B_to_I((25 … 32) ⌈ w) •
$B_3 + B_2 * 2 ↑ 8 + B_1 * 2 ↑ 16 + B_0 * 2 ↑ 24$))))

I_to_W : IWord → Word

I_to_W = W_to_I˜

 # ۹-۱-۳ توابع

Indexed_f == {f : seq (Word × Word × Word → Word) | # f = 80}

f : Indexed_f

$\forall X_0, X_1, X_2 :$ Word •
(∀ i : 1 .. 20 •
f(i)(X_0, X_1, X_2) = (X_0 AND X_1) OR (NOT X_0 AND X_2)) □
(∀ i : 21 .. 40 •
f(i)(X_0, X_1, X_2) = X_0 XOR X_1 XOR X_2) □
(∀ i : 41 .. 60 •
f(i)(X_0, X_1, X_2) = (X_0 AND X_1) OR (X_0 AND X_2) OR(X_1 AND X_2)) □
(∀ i : 61 .. 80 •
f(i)(X_0, X_1, X_2) = X_0 XOR X_1 XOR X_2)

 # ۹-۱-۴ ثابت‌ها

*x*5*A*827999 == 1518500249

*x*6*ED*9*EBA*1 == 1859775393

*x*8*F*1*BBCDC* == 2400959708

*xCA*62*C*1*D*6 == 3395469782

Constants == {c : StringWord | # c = 80}

C: Constants

(∀ i : 1 .. 20 •
C(i) = I_to_W(x5A827999)) □
(∀ i : 21 .. 40 •
C(i) = I_to_W(x6ED9EBA1)) □
(∀ i : 41 .. 60 •
C(i) = I_to_W(x8F1BBCDC)) □
(∀ i : 61 .. 80 •
C(i) = I_to_W(xCA62C1D6)) □

 # ۹-۱-۵ مقدار اولیه

x67452301 == 1732584193
Y_0 == I_to_W(x67452301)
xEFCDAB89 == 4023233417
Y_1 == I_to_W(xEFCDAB89)
x98BADCFE == 2562383102
Y_2 == I_to_W(x98BADCFE)
x10325476 == 271733878
Y_3 == I_to_W(x10325476)
xC3D2E1F0 == 3285377520
Y_4 == I_to_W(xC3D2E1F0)

IV = Y_0□ Y_1□ Y_2□ Y_3□ Y_4

۹-۲ روش لایه‌گذاری

```
∀ D : String •
    Pad(D) =
        (let L_D == # D •
        (let Zeros == {n : 1 … ((447 – L_D) mod 512) • n ↦ 0} •
        (let Length_D_MSH == I_to_W (L_D div (2 ↑ 32)) •
        (let Length_D_LSH == I_to_W (L_D mod (2 ↑ 32)) •
            D ⌢ ⟨1⟩ ⌢ Zeros ⌢ Length_D_MSH ⌢ Length_D_LSH))))
```

۹-۳ توصیف تابع گردساز

```
StringWord == seq Word
```

```
Split_String_to_StringWord : String → StringWord
Split_String_to_StringWord =
    {s : String; sw : StringWord | s = ⌢/sw • s ↦ sw}
```

```
L80:StringWord × StringWord × 1 .. 80 ⇸ StringWord
∀ X, Z : StringWord; i : 1 … 80 | # X = 5 ⌢ Z = 80 •
    L80(X, Z, i ) =
    (let W = = S^5 (X(1)) ⊞ f(i) ⊞ (X(2), X(3), X(4)) ⊞ X(5) ⊞ Z(i) ⊞ C(i) •
    (let X0 == W •
    (let X1 == X(1) •
    (let X2 == S^30 (X(2)) •
    (let X3 == X(3) •
    (let X4 == X(4) •
     (let Y == ⟨X0, X1, X2, X3, X4⟩ •
        if (i=80)
        then Y
        else L80(Y, Z, i+1))))))))
```

```
XOR_Z : StringWord ⇸ StringWord
∀ Z1_16 : StringWord | # Z1_16 = 16 •
    (∀ i : 1 .. 16 •
        XOR_Z(Z_16)(i) = Z1_16(i)) ⌢
     (∀ i : 17 .. 80 •
        XOR_Z(Z_16)(i) = S^1 (XOR_Z1(Z1_16)(i-3) XOR
                              (XOR_Z1(Z1_16)(i-8) XOR
                              (XOR_Z1(Z1_16)(i-14) XOR
                              (XOR_Z1(Z1_16)(i-16))
```

```
∀ sm : String_L1; sn : String_L2 •
    Φ(sm, sn) =
        (let Z1_16 == Split_String_to_StringWord(sm) •
        (let Y == Split_String_to_StringWord(sn) •
        (let Z == XOR_Z(Z1_16) •
```

(let X == L80(Y, Z, 1) •
(let Y0 == Y(1) ⊞ X(1) •
(let Y1 == Y(1) ⊞ X(1) •
(let Y2 == Y(1) ⊞ X(1) •
(let Y3 == Y(1) ⊞ X(1) •
Y_0□ Y_1□ Y_2□ Y_3 □ Y_4)))))))))

پیوست پ

(الزامی)

پیمانه ASN.1

این پیوست، پیمانه ASN.1 انتصابی به توابع درهم ساز اختصاصی مشخص شده در این استاندارد را فهرست می‌کند.

```
-- ISO/IEC FDIS 10118-3 Proposed ASN.1 Module

-- Based on ISO/IEC JTC 1/SC 27 N 3340 2002-10-21

--

DedicatedHashFunctions {

iso(1) standard(0) hash-functions(10118) part(3)

asn1-module(1) dedicated-hash-functions(0) }

DEFINITIONS EXPLICIT TAGS ::= BEGIN

-- EXPORTS All; --

-- IMPORTS None; --

OID ::= OBJECT IDENTIFIER -- alias

-- Synonyms --

id-dhf OID ::= {

iso(1) standard(0) hash-functions(10118) part3(3) algorithm(0) }

-- Assignments --

id-dhf-ripemd160 OID ::= { id-dhf ripemd160(49) }

id-dhf-ripemd128 OID ::= { id-dhf ripemd128(50) }

id-dhf-whirlpool OID ::= { id-dhf whirlpool(55) }
```

```
-- note: assign any new OIDs above 55
-- FIPS 180-1 and FIPS 180-2 Secure Hash Algorithm --
id-sha1 OID ::= {
iso(1) identified-organization(3) oiw(14) secsig(3)
algorithm(2) 26
}
sha2Algorithm OID ::= {
joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)
csor(3) nistAlgorithm(4) hashAlgs(2)
}
id-sha256 OID ::= { sha2Algorithm sha256(1) }
id-sha384 OID ::= { sha2Algorithm sha384(2) }
id-sha512 OID ::= { sha2Algorithm sha512(3) }
HashFunctions ::= SEQUENCE {
algorithm ALGORITHM.&id({HashFunctionAlgs}),
parameters ALGORITHM.&Type({HashFunctionAlgs}{@algorithm}) OPTIONAL
}
HashFunctionAlgs ALGORITHM ::= {
dhf-ripemd160 |
dhf-ripemd128 |
dhf-whirlpool |
SHA-Algorithms,
... -- Expect additional algorithms --
}
dhf-ripemd160 ALGORITHM ::= {
OID id-dhf-ripemd160 PARMS NullParms
}
dhf-ripemd128 ALGORITHM ::= {
OID id-dhf-ripemd128 PARMS NullParms
}
```

```
dhf-whirlpool ALGORITHM ::= {
OID id-dhf-whirlpool PARMS NullParms
}
SHA-Algorithms ALGORITHM ::= {
-- The parameters associated with id-sha1, id-sha256, id-sha384, --
-- and id-sha512 should be omitted, but if present, shall have --
-- a value of ASN.1 type NULL. This is to align with the original --
-- NIST definitions (which did not have parameters) and certain --
-- exisiting implementations (which have them). For these SHA --
-- algorithms, implementations shall accept AlgorithmIdentifier --
-- values with NULL parameters and with the optional parameters --
-- component not present. --
sha-1 |
sha-256 |
sha-384 |
sha-512,
... -- Expect additional algorithms --
}
sha-1 ALGORITHM ::= {
OID id-sha1 PARMS NullParms
}
sha-256 ALGORITHM ::= {
OID id-sha256 PARMS NullParms
}
sha-384 ALGORITHM ::= {
OID id-sha384 PARMS NullParms
}
sha-512 ALGORITHM ::= {
OID id-sha512 PARMS NullParms
}
```

```
NullParms ::= NULL

-- Cryptographic algorithm identification --

ALGORITHM ::= CLASS {

&id OBJECT IDENTIFIER UNIQUE,

&Type OPTIONAL

}

WITH SYNTAX { OID &id [PARMS &Type] }

END -- DedicatedHashFunctions --
```

کتاب‌نامه

[1] J.M. Spivey, *The Z Notation — A Reference Manual*, Prentice-Hall, 1992 (2nd edition)

[2] U.S. Department of Commerce/National Institute of Standards and Technology, *Secure Hash Standard*, Federal Information Processing Standards Publication (FIPS PUB) 180-2, 1st August 2002

[3] Bosselaers, H. Dobbertin and B. Preneel, *The new cryptographic hash function RIPEMD-160*, Dr. Dobbs, Vol. 22 No.1, pp.24-28, January 1997

[4] P.S.L.M. Barreto and V. Rijmen. *The WHIRLPOOL Hashing Function*, First open NESSIE Workshop, Leuven, 13-14 November 2000