بسم الله الرحمن الرحيم

٩٣٩٨٨

دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر

بهینه‌سازی ساختارهای نرم‌افزاری سیستم‌های چندپردازنده‌ای برای کاربردهای چندرسانه‌ای

نگارش:
محمدعلی توتونچیان

استاد راهنما:
دکتر محمود رضا هاشمی

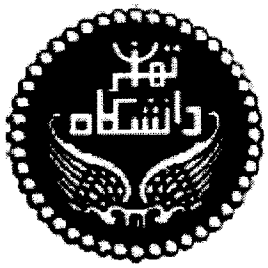
استاد مشاور:
دکتر سید امید فاطمی

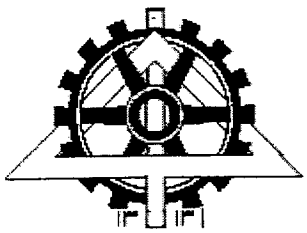
پایان‌نامه برای دریافت درجه کارشناسی ارشد در رشته مهندسی کامپیوتر گرایش معماری کامپیوتر

بهمن ۱۳۸۶

۱۳۸۷ /۲/ ۱۰۲

۹۳۹۸۸





دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

پایان‌نامه برای دریافت درجه کارشناسی ارشد در رشته

مهندسی کامپیوتر - معماری کامپیوتر

عنوان:

بهینه‌سازی ساختارهای نرم‌افزاری سیستم‌های چندپردازنده‌ای برای کاربردهای چندرسانه‌ای

نگارش:

محمدعلی توتونچیان

از این پایان نامه در تاریخ ۱۳۸۶/۱۱/۲۸ در مقابل هیات داوران دفاع گردید و مورد تصویب قرار گرفت.

معاون آموزشی و تحصیلات تکمیلی پردیس دانشکده‌های فنی:	دکتر جواد فیض
رئیس دانشکده مهندسی برق و کامپیوتر:	دکتر پرویز جبه‌دار مارالانی
معاون پژوهشی و تحصیلات تکمیلی دانشکده مهندسی برق و کامپیوتر:	دکتر سعید نادر اصفهانی
استاد راهنما:	دکتر محمودرضا هاشمی
استاد مشاور:	دکتر سید امید فاطمی
عضو هیات داوران و نماینده تحصیلات تکمیلی:	دکتر سیامک محمدی
عضو هیات داوران:	دکتر حسین پدرام
عضو هیات داوران:	دکتر سعید صفری

تعهدنامه اصالت اثر

این جانب **محمدعلی توتونچیان** تایید می‌کنم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی این جانب است و به دستاوردهای پژوهشی دیگران که در این نوشته از آن‌ها استفاده شده است، مطابق مقررات ارجاع گردیده است. این پایان نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتری ارائه نشده است.

کلیه حقوق مادی و معنوی این اثر متعلق به دانشکده فنی دانشگاه تهران می‌باشد.

نام و نام خانوادگی: **محمدعلی توتونچیان**

تقدیم به:

همسر مهربانم،

پدر و مادر عزیزم

و خواهر و برادران گرامی‌ام .

تشکر و قدردانی:

لازم می دانم از زحمات **جناب آقای دکتر محمود رضا هاشمی** و **جناب آقای دکتر سید امید فاطمی** که راهنمایی و مشاوره اینجانب را بر عهده داشتند، صمیمانه تشکر و سپاسگزاری کنم. بی شک انجام این رساله بدون راهنمایی و نظرات ایشان میسر نبود.

این رساله از حمایت مالی مرکز تحقیقات مخابرات ایران برخوردار بوده است که به این ترتیب مراتب قدردانی و سپاسگزاری خود را از آن مرکز اعلام می دارم .

همچنین از تمامی دوستانم در آزمایشگاه چند رسانه‌ای مراتب سپاس و تشکر را دارم.

چکیده

امروزه از ساختارهای چندپردازنده نهفته درونی به‌عنوان یک بستر برای پیاده‌سازی برنامه‌های کاربردی چندرسانه‌ای یاد می‌شود. زمان‌بندی فرآیندهای تشکیل‌دهنده و تقسیم منابع میان این فرآیندها، یکی از مهم‌ترین چالش‌هایی است که طراحان این سیستم‌ها با آن مواجه هستند. از سوی دیگر وجود جریان داده‌ای با ویژگی‌های متغیر و پارامترهای متغیر در اجرای فرآیندها، به‌عنوان دو مشخصه اصلی برنامه‌های کاربردی چندرسانه‌ای، سبب می‌گردد الگوریتم‌های زمان‌بندی و تقسیم منابع بهینه برای این‌گونه کاربردها بسیار متفاوت و متمایز از الگوریتم‌های متداول زمان‌بندی باشد.

در این رساله ساختار بهینه طراحی الگوریتم‌های زمان‌بندی بلادرنگ برای برنامه‌های کاربردی چندرسانه‌ای در ساختارهای چندپردازنده‌ای معرفی شده‌است. با استفاده از ساختار توصیف‌شده و بر مبنای الگوریتم‌های زمان‌بندی مرسوم ایستا (TDMA) و پویا (EDF) ایده‌های جدیدی برای طراحی الگوریتم‌های زمان‌بندی پویایی با نام‌های D-TDMA و D-EDF معرفی گردیده‌اند. نتایج شبیه‌سازی‌های متنوع انجام‌شده روی کدگشای MPEG-2 بعنوان یک نمونه نشان می‌دهد که ضریب بهره‌وری منابع پردازشی و حافظه‌ای برای اجرای این برنامه‌ی کاربردی در یک ساختار چندپردازنده‌ای با استفاده از الگوریتم‌های یادشده در مقایسه با الگوریتم‌های زمان‌بندی متداول بترتیب تا ۳۰٪و ۴۰٪ افزایش می‌یابد.

در این رساله هم‌چنین برای شبیه‌سازی سطح بالای برنامه های کاربردی در ساختارهای چندپردازنده‌ای یک بستر کاری معرفی و طراحی شده‌است. در مقایسه با سایر نرم افزارهای شبیه‌سازی سطح بالای برنامه‌های کاربردی در ساختارهای چندپردازنده‌ای بستر کاری معرفی‌شده علاوه بر دارا بودن ویژگی‌های خاص و یکتا، از سرعت بالا در شبیه‌سازی نیز برخوردار می‌باشد. هم‌چنین نتایج بدست آمده از شبیه‌سازی

برنامه‌های کاربردی با استفاده از این بستر نتایجی قابل مقایسه با نرم‌افزارهای شبیه‌سازی سطح پایین را در کسری از زمان آنها ارائه می دهد.

فهرست مطالب

فصل ۱. مقدمه ۱
فصل ۲. معرفی ساختارهای چندپردازنده‌ای و برنامه‌های کاربردی چندرسانه‌ای ۷
۲-۱. مقدمه ۸
۲-۲. ویژگی برنامه‌های کاربردی چندرسانه‌ای ۸
۲-۲-۱. جریان داده در سیستم‌های چندپردازنده‌ای ۹
۲-۲-۲. پردازش داده در سیستم‌های چندرسانه‌ای ۹
۲-۲-۳. عملیات موازی در استاندارهای چندرسانه‌ای ۱۰
۲-۲. ویژگی ساختارهای چندپردازنده‌ای ۱۱
۲-۳. طراحی برنامه‌های چندرسانه‌ای در ساختارهای چندپردازنده‌ای ۱۵
۲-۳-۱. توصیف برنامه‌ی کاربردی ۱۵
۲-۳-۲. توصیف محدودیت‌های برنامه‌ی کاربردی: کیفیت ارائه‌ی سرویس ۱۷
۲-۳-۳. چگونگی تخصیص منابع و زمان‌بندی فرآیندها ۱۷
فصل ۳. مروری بر الگوریتم‌های زمان‌بندی بلادرنگ و معرفی ساختار بهینه ۱۸
۳-۱. مقدمه ۱۹
۳-۲. معرفی الگوریتم‌های زمان‌بندی و مفاهیم سیستم‌های بلادرنگ ۲۰
۳-۲-۱. سیستم‌های بلادرنگ ۲۱
۳-۲-۲. اصول زمان‌بندی بلادرنگ ۲۳
۳-۳. مروری بر الگوریتم‌های زمان‌بندی ارائه شده ۲۷
۳-۳-۱. الگوریتم‌های زمان‌بندی جریان داده‌ای ۲۷
۳-۳-۲. الگوریتم‌های زمان‌بندی برای ساختارهای چندپردازنده‌ای ۳۰
۳-۴. ساختار بهینه برای الگوریتم‌های زمان‌بندی در سیستم‌های چندرسانه‌ای و چندپردازنده‌ای ۳۲
۳-۴-۱. طراحی سیستم بدون تشریک منابع ۳۳
۳-۴-۲. طراحی سیستم با تشریک منابع به‌صورت ایستا ۳۴
۳-۴-۳. طراحی سیستم با تشریک منابع به‌صورت پویا ۳۵
فصل ۴. الگوریتم‌های پیشنهادی براساس الگوهای تخصیص منبع پیش از اجرا (TDMA) ۳۹
۴-۱. معرفی ویژگی‌های الگوریتم TDMA به‌عنوان الگوریتم پایه ۴۰
۴-۲. معرفی ایده طراحی الگوریتم TDMA پویا ۴۰

۴-۳. الگوریتم‌های مختلف طراحی شده براساس ایده D-TDMA ۴۳
۴-۳-۱. پارامترهای تصمیم‌گیری ۴۳
۴-۳-۲. زمان تصمیم‌گیری ۴۶
۴-۳-۳. مدل چندپردازنده‌ای ۴۸
۴-۳-۴. خلاصه‌ای از ویژگی‌های الگوریتم‌های طراحی شده بر اساس ایده D-TDMA ۴۸
فصل ۵. الگوریتم‌های پیشنهادی براساس الگوهای تخصیص منبع در حین اجرا (EDF) ۵۲
۵-۱. مقدمه ۵۳
۵-۲. معرفی الگوریتم EDF به‌عنوان الگوریتم زمان‌بندی پایه ۵۳
۵-۳. معرفی ایده طراحی الگوریتم EDF پویا ۵۴
۵-۴. الگوریتم‌های مختلف طراحی شده براساس ایده D-EDF ۵۷
۵-۴-۱. استفاده از تکنیک صف‌های چندگانه ۵۷
۵-۴-۲. استفاده از ضرب‌العجل‌های چندگانه ۶۰
۵-۴-۳. استفاده از ضرب‌العجل‌های متغیر ۶۱
۵-۴-۴. مدل چندپردازنده‌ای ۶۲
۵-۴-۵. خلاصه و چکیده الگوریتم‌های طراحی شده براساس ایده D-EDF ۶۲
فصل ۶. پیاده‌سازی الگوریتم‌های طراحی شده ۶۶
۶-۱. مقدمه ۶۷
۶-۲. توصیف سیستم ارزیابی الگوریتم‌های بلادرنگ در ساختارهای چندپردازنده‌ای ۶۷
۶-۲-۱. هدف پیاده‌سازی ۶۷
۶-۲-۲. معیارهای ارزیابی الگوریتم‌های زمان‌بندی و کارآیی سیستم ۶۹
۶-۲-۳. معرفی عناصر سازنده سیستم DSSP ۷۰
۶-۲-۴. انتقال جریان داده‌ها: ارتباط میان منابع و فرآیندها ۷۱
۶-۲-۵. مولدهای جریان داده ۷۵
۶-۲-۶. فرآیندهای اجرایی ۷۹
۶-۲-۷. پردازنده‌ها و الگوریتم‌های تخصیص منابع و زمان‌بندی ۷۹
۶-۳. پیاده‌سازی ساختارهای پویا ۸۴
۶-۳-۱. ساختار داده‌ای مورد استفاده برای الگوریتم‌های زمان‌بندی پویا ۸۴
۶-۳-۲. پیاده‌سازی الگوریتم‌های زمان‌بندی پویا ۹۰
۶-۳-۳. خلاصه‌ای از ویژگی‌های سیستم DSSP ۹۵

۶-۴. برنامه‌ی کاربردی با استفاده از هسته مرکزی DSSP ۹۸
۶-۴-۱. تنظیم پارامترهای توصیف کننده سیستم ۹۸
۶-۴-۲. تنظیم مدل‌های پویای زمان‌بندی ۱۰۱
فصل ۷. ارزیابی الگوریتم‌ها و مقایسه نتایج ۱۰۳
۷-۱. مقدمه ۱۰۴
۷-۲. ارزیابی الگوریتم‌ها با استفاده از ورودی‌های اتفاقی ۱۰۴
۷-۲-۱. مقایسه الگوریتم‌های خانواده D-TDMA با الگوریتم پایه ۱۰۶
۷-۲-۲. مقایسه الگوریتم‌های خانواده D-EDF با الگوریتم پایه ۱۱۷
۷-۳. استفاده از الگوریتم‌های زمان‌بندی پویا در پیاده‌سازی برنامه‌ی کاربردی MPEG-2 ..... ۱۲۸
۷-۳-۱. معرفی ساختار چندپردازنده‌ای برای پیاده‌سازی برنامه‌ی کاربردی MPEG-2 ..... ۱۲۸
۷-۳-۲. نتایج به‌دست آمده از اجرای برنامه‌ی کاربردی MPEG-2 در سیستم DSSP ۱۳۳
۷-۳-۳. مقایسه نتایج به‌دست‌آمده با پیاده‌سازی‌های مشابه ۱۴۲
فصل ۸. نتیجه‌گیری و کارهای آینده ۱۴۴
۸-۱. نتیجه‌گیری ۱۴۵
۸-۲. کارهای آینده ۱۴۶
فصل ۹. مراجع و منابع ۱۴۹
فصل ۱۰. مقالات منتشرشده ۱۵۴
۱۰-۱. مقالات منتشرشده در کنفرانس‌ها ۱۵۵
۱۰-۲. مقالات منتشرشده در مجلات ۱۵۵
۱۰-۳. مقالات ارسال‌شده به کنفرانس‌ها ۱۵۵

فهرست جدول‌ها

جدول ۴-۱ : ویژگی‌های الگوریتم‌های زمان‌بندی خانواده D-TDMA ۵۱
جدول ۵-۱ : ویژگی‌های الگوریتم‌های زمان‌بندی خانواده D-EDF ۶۴
جدول ۶-۱ : عناصر تشکیل دهنده ساختار داده‌ای Core_Param ۸۶
جدول ۶-۲ : مدل‌های زمان‌بندی پویای پیاده‌سازی شده ۸۶
جدول ۶-۳ : توصیف چگونگی تعیین مقدار جدید انواع مختلف متغیرهای پویا ۸۷
جدول ۶-۴ : عناصر تشکیل‌دهنده ساختار داده‌ای فرآیندهای پویا ۸۸
جدول ۶-۵ : معرفی عناصر تشکیل دهنده ساختار داده‌ای ویژگی‌های لحظه‌ای فرآیندها ۹۰
جدول ۶-۶ : عناصر تشکیل‌دهنده ساختار TDMA_Param ۹۱
جدول ۶-۷ : عناصر تشکیل‌دهنده ساختار EDF_Param ۹۴
جدول ۶-۸ : پارامترهای توصیف کننده یک برنامه‌ی کاربردی و ساختار اجرایی آن در سیستم DSSP ۹۷
جدول ۷-۱ : ویژگی‌های آزمون‌های توصیف شده ۱۰۶
جدول ۷-۲ : ویژگی‌های جریان‌های ویدیو ۱۳۳
جدول ۷-۳ : ویژگی جریان صدا ۱۳۳
جدول ۷-۴ : معرفی جریان‌های چندرسانه‌ای ورودی ۱۳۴

فهرست شکل‌ها

شکل ۲-۱ : ساختار یک سیستم چندپردازنده‌ای ۱۲
شکل ۲-۲ : ساختار کدکننده MPEG-2 ..... ۱۳
شکل ۲-۳ : مراحل پیاده‌سازی یک برنامه‌ی کاربردی در یک ساختار چندپردازنده‌ای ۱۵
شکل ۳-۱: زمان‌های مهم در اجرای یک فرآیند در سیستم‌های بلادرنگ ۲۴
شکل ۴-۱ : سیستم ساده برای ارزیابی الگوریتم‌های تقسیم منابع پویا ۴۱
شکل ۵-۱ : سیستم ساده برای ارزیابی الگوریتم‌های تقسیم منابع پویا ۵۵
شکل ۶-۱ : فرآیند فعال شونده اجماعی ۷۲
شکل ۶-۲ : ورودی متناوب ۷۵
شکل ۶-۳ : ورودی متناوب با تقدم و تأخر ۷۶
شکل ۶-۴ : ورودی متناوب و پیوسته ۷۷
شکل ۶-۵ : نمایی از صفحه نمایش پارامترهای سیستم توصیف شده در DSSP-UI ۹۹
شکل ۶-۶ : فرم تنظیم پارامترهای جریان داده ۱۰۱
شکل ۶-۷ : فرم تنظیم پارامترهای فرآیند ۱۰۱
شکل ۶-۸ : فرم انتخاب الگوریتم‌های زمان‌بندی پویا برای ارزیابی سیستم ۱۰۲
شکل ۷-۱ : ساختار شماره ۱ - یک‌پردازنده، دو جریان داده ۱۰۵
شکل ۷-۲ : ساختار شماره ۲ - یک‌پردازنده، سه جریان داده ۱۰۵
شکل ۷-۳ : ساختار شماره ۳ - دوپردازنده، دو جریان داده ۱۰۵
شکل ۷-۴ : ساختار شماره ۴ - دوپردازنده، سه جریان داده ۱۰۵
شکل ۷-۵ : ساختار عناصر تشکیل‌دهنده برنامه‌ی کاربردی کدگشای ویدیو MPEG-2 ..... ۱۲۸
شکل ۷-۶ : ساختار آزمایشی برای کدگشایی صوت و تصویر به‌صورت هم‌زمان ۱۳۰
شکل ۷-۷ : ساختار آزمایشی برای کدگشایی جریان ویدیو MPEG-2 با دو کیفیت متفاوت ۱۳۲
شکل ۷-۸ : ساختار در نظر گرفته شده برای پردازش هم‌زمان دو جریان تصویر متفاوت ۱۳۳

فصل ۱. مقدمه

بیش از یک دهه است که روش طراحی سیستم بر روی یک تراشه[1] به‌عنوان یک رویکرد جدید و کارا در نزد طراحان سیستم شناخته می‌شود. طراحی یک سیستم به صورت کاملاً سخت‌افزاری و یا طراحی همان سیستم به صورت برنامه‌ای نرم‌افزاری وقابل اجرا بر روی یک پردازنده، هر کدام مزایا و معایبی دارند. در برخی موارد سیستم کاملاً سخت‌افزاری و در پاره‌ای از موارد سیستم‌های نرم‌افزاری بر رقیب خویش برتری می‌یابند. سیستم سخت‌افزاری برای یک کاربرد خاص پیاده‌سازی می‌شود و در طراحی آن ملاحظاتی در نظر گرفته می‌شود که ویژه‌ی همان سیستم می‌باشد. تغییر ویژگی‌های یک سیستم سخت‌افزاری پیاده‌سازی شده مستلزم طراحی و بازنگری مجدد می‌باشد. اما به‌دلیل آن‌که پردازنده‌ای که باید پیاده‌سازی نرم‌افزاری سیستم بر روی آن اجرا گردد عام منظوره[2] می‌باشد امکان تغییر نرم‌افزار راحت‌تر و سریع‌تر می‌باشد. اما معمولاً پیاده‌سازی نرم‌افزاری به اندازه‌ی سیستم سخت‌افزاری برای کاربرد خاص بهینه نمی‌باشد. به طور کلی انتظار طراح از یک پردازنده آن است که بتواند پیاده‌سازی‌های گوناگونی را اجرا نموده و کاربری‌های متنوعی داشته باشد حال آن که از یک سیستم سخت‌افزاری تنها این توقع وجود دارد که کاربردی را داشته باشد که به منظور آن طراحی شده و انتظار رفتارهای متنوع از آن وجود ندارد.

سیستم بر روی یک تراشه، که اکنون به یک ایده‌ی محوری در طراحی سیستم‌های دیجیتال تبدیل شده، تلاش می‌کند تا مزایای هر دو روش پیاده‌سازی را تجمیع نموده و از معایب آن‌ها نیز اجتناب نمایند. پیاده‌سازی سخت‌افزاری سبب تولید سیستمی با کارآیی بالاتر خواهد شد، در حالی که انعطاف پذیری[3] چندانی ندارد و زمان طراحی و تولید آن نیز زمان قابل توجهی خواهد بود. اما استفاده از سیستم‌های نرم‌افزاری که برای یک پردازنده‌ی نهفته[4] طراحی می‌شوند سبب کاهش زمان رسیدن محصول به بازار[5] می‌شود. هم‌چنین این سیستم‌ها انعطاف‌پذیری و قابلیت تغییر و توسعه‌ی بیش‌تری دارند و قابلیت تنظیم[6] و تغییر را به ویژگی‌های سیستم اضافه می‌نماید. اما باید توجه داشت سیستمی که به صورت نرم‌افزاری و برای یک پردازنده‌ی نهفته تولید می‌گردد عموماً در مقایسه با همان سیستم که به‌صورت سخت‌افزاری طراحی شده است از کارآیی کم‌تری برخوردار است.

توسعه‌ی فناوری و افزایش نیاز کاربردهای متداول برای سیستم‌های دیجیتال سبب گردید که سیستم‌های بر روی یک تراشه که ترکیبی از یک پردازنده و تعدادی شتاب‌دهنده‌ی سخت‌افزاری[7] است توان رسیدن به بالاتر از حدی مشخص در کارآیی را نداشته باشند. اولین محدودیت که بر سر راه یک سیستم یکپارچه و هم‌زمان بر روی یک تراشه وجود دارد محدودیت سرعت نور است. اگر فرض کنیم قطر یک تراشه که

[1] System on Chip (SoC)
[2] General Purpose Processor
[3] Flexibility
[4] Embedded Processor
[5] Time to Market
[6] Configurable
[7] Hardware Accelerator

طولانی‌ترین مسیر در آن است برابر سه سانتی‌متر باشد با در نظر گرفتن سرعت نور درمی‌یابیم که اگر داده‌ای بخواهد این مسیر را طی کند، در آن تراشه در بهترین شرایط هرگز به فرکانسی کاری بالاتر از ده گیگاهرتز نخواهیم رسید[۱]. علاوه برآن توسعه‌ی تکنولوژی و کاهش ابعاد آن مشکل اتصالات را بسیار حائز اهمیت نموده‌است. افزایش شدید مصرف توان با کاهش ابعاد سیستم سبب شده‌است که در نظر گرفتن طول و توپولوژی اتصالات به چالشی مهم در طراحی سیستم بدل شود و امکان گسترش ابعاد سیستم به‌راحتی وجود نداشته باشد [۲].

بنابراین اگر برنامه‌ی کاربردی[۸] نیاز به اجرای دستورالعمل‌های فراوانی در واحد زمان داشته باشد، با توجه به محدودیت‌هایی یاد شده در طراحی سیستم‌های دیجیتال، در پاره‌ای از موارد امکان پیاده‌سازی برنامه‌ی کاربردی در یک سیستم بر روی یک تراشه وجود ندارد. ناگزیر باید راه‌حل‌های جدیدی را برای طراحی سیستم برگزید. راه‌حلی که در سالیان اخیر در مسیر طراحی سیستم ایجاد گردید و مورد اقبال مهندسین طراح واقع شد، سیستم‌های چندپردازنده‌ای بر روی یک تراشه[۹] است. ایده‌ی مرکزی این سیستم‌ها رویکرد به سیستم‌های توزیع شده در مقابل سیستم‌های متمرکز می‌باشد [۳]. در حقیقت به‌جای طراحی یکپارچه‌ی یک سیستم بزرگ و با کارآیی پایین، از تعداد زیادی سیستم کوچک با کارآیی بالاتر استفاده کرده و عملیات محاسباتی برنامه‌ی کاربردی را برروی آن‌ها توزیع می‌نماییم. به این طریق هم عناصر سازنده‌ی کوچک‌تر ما کارآیی بالاتری خواهند داشت و هم اجرای هم‌زمان عناصر سیستم سبب هم‌افزایی عناصر کوچک و دست‌یابی به بهره‌وری بیش‌تر منابع خواهد شد. مهم‌ترین پیش‌فرض‌ها در طراحی یک برنامه‌ی کاربردی برروی سیستم‌های چندپردازنده‌ای بر روی یک تراشه آن است که اولاً حجم عملیات سیستم به حدی باشد که هزینه‌ی طراحی و اجرای آن بر روی یک سیستم تک‌پردازنده‌ای بر روی یک تراشه بیش‌تر از مقیاس قابل قبول باشد و ثانیاً امکان استخراج شاخه‌های عملیات اجرایی موازی وجود داشته باشد [۲]. با وجود این دو شرط می‌توان از روش طراحی چند پردازنده‌ای بر روی یک تراشه به عنوان یک از راه‌حل‌های موجود برای افزایش کارآیی و کاهش هزینه‌ی طراحی برنامه‌ی کاربردی استفاده نمود [۴].

طراحی سیستم‌های چندپردازنده‌ای به سبب افزودن ویژگی‌های جدید به دنیای طراحی، دارای مسائل نوینی است که در هنگام طراحی سیستم باید مورد توجه قرار گیرد. این مسائل بر آن‌چه که پیش از این در فضای طراحی سیستم‌های بر روی تراشه وجود داشت افزوده می‌گردد و در نتیجه سبب بزرگ‌تر شدن فضای طراحی خواهد شد.

مهم‌ترین مسائلی که در طراحی یک سیستم چندپردازنده برروی تراشه باید مورد توجه قرارگیرد و پیش از طراحی سیستم باید برای آن‌ها پاسخ مناسب یافته شود، عبارتند از [۲] [۵] [۶]:

[8] Application
[9] Multiprocessor System on Chip (MPSoC)

- معماری پردازنده‌ها
- ساختار واحدهای حافظه
- معماری اتصالات موجود میان عناصر سیستم.

در بررسی هر کدام از این موضوعات پارامترها و مسایل حائز اهمیتی یافته خواهد شد که مطالعه برروی هرکدام و یافتن نقطه‌ی بهینه پیاده‌سازی مستلزم جستجو در فضای مهیب طراحی چند بعدی می‌باشد. بر انتخاب این پاسخ‌ها و رسیدن به راه‌حل مساله معیارهایی دلالت می‌کنند که آن‌ها نیز باید از سوی مهندس طراح سیستم برگزیده شوند. ویژگی‌های برنامه‌ی کاربردی و نیازها و ساختارهای آن، بستری که سیستم باید بر روی آن پیاده‌سازی شود، معیارهای بهینه‌سازی، زمان طراحی و تولید سیستم و سایر محدودیت‌هایی که به مسیر طراحی جهت می‌دهند، می‌توانند مجموعه‌ی معیارهای طراحی سیستم را شکل دهند.

برنامه‌ی کاربردی خاصی که در این رساله مورد توجه قرار گرفته، استانداردهای سیستم‌های چندرسانه‌ای[10] می‌باشد. این استانداردها مخصوص کدگذاری و کدگشایی جریان تصاویر و صوت است. بدیهی است که کدگذاری جریان چندرسانه[11] در این استانداردها عملیاتی است که تنها یک‌بار و در فرستنده صورت می‌پذیرد. حجم عملیات بالای مورد نیاز برای کدگذاری استفاده از دستگاه‌های کامپیوتری را موجه‌تر از سیستم‌های بر روی یک تراشه می‌نماید. چرا که در سمت کدگذار کمبود منابعی هم‌چون انرژی، منابع محاسباتی و حافظه‌ای به‌سبب عدم نیاز به قابل حمل بودن کدگذار و عدم استفاده‌ی مشتری عام از آن وجود ندارد و می‌توان از کامپیوترهای قدرتمند برای این منظور استفاده نمود. اما در سمت کدگشا که باید بر روی هر گیرنده‌ی تصویر وجود داشته باشد، احتمال مواجهه با این محدودیت‌ها بیش‌تر است. این محدودیت‌ها دامنه‌ی بسیار متنوعی دارند و شامل محدودیت در سیلیکون، انرژی، فرکانس‌کاری، حافظه و دیگر محدودیت‌های مشابه خواهند بود. بنابراین کاربرد سیستم‌های برروی یک تراشه بیش‌تر برای سمت کدگشای استاندارد قابل توجیه خواهد بود.

با توجه به مسایل گفته‌شده، پیاده‌سازی برنامه‌های کاربردی چندرسانه‌ای در سیستم‌های چندپردازنده‌ای از پیچیدگی‌های خاصی برخوردار است. یکی از موضوعات مورد علاقه محققان در چند سال اخیر، طراحی و پیاده‌سازی ابزارهایی می‌باشد که به کمک آن‌ها بتوان مدلی از نگاشت[12] برنامه‌ی کاربردی بر منابع موجود در سیستم‌های چندپردازنده‌ای ارائه نمود. این نگاشت‌ها عموماً با توجه به توصیف سطح بالای برنامه‌ی کاربردی چندرسانه‌ای صورت می‌پذیرد. نگاشت یک برنامه‌ی کاربردی بر روی یک سیستم چندپردازنده‌ای دارای سه مرحله اصلی می‌باشد.

[10] Multimedia
[11] Media Stream
[12] Mapping

- تعیین و تخصیص پردازنده به فرآیندهای[۱۳] تشکیل‌دهنده‌ی برنامه‌ی کاربردی
- تعیین ترتیب اجرای فرآیندهای تخصیص داده شده به هر پردازنده
- تعیین زمان شروع اجرای هر فرآیند بر روی هر پردازنده
- تخصیص منابع حافظه‌ای

مجموع مراحل یاد شده زمان‌بندی[۱۴] برنامه‌ی کاربردی نامیده می‌شود. نکته کلیدی در طراحی و پیاده‌سازی زمان‌بند یک برنامه‌های کاربردی چندرسانه‌ای در سیستم‌های چندپردازنده‌ای که سبب تمایز آن از ساختارهای زمان‌بندی کلاسیک می‌گردد، نقش جابجایی داده‌ها میان فرآیندهایی است که توسط پردازنده‌های مختلف اجرا می‌شوند. در برنامه‌های کاربردی چندرسانه‌های جابجایی داده‌ها به اندازه پردازش داده‌ها در تعیین زمان اجرا و توان مصرفی حائز اهمیت می‌باشد. هزینه ارتباط میان پردازنده‌ها می‌تواند اثر قابل توجهی بر سرعت اجرای برنامه‌ی کاربردی داشته باشد. توصیف و تعریف چگونگی ارتباط میان پردازنده‌ها نقش مهمی در طراحی و پیاده‌سازی مقرون به‌صرفه و بهینه برنامه‌های کاربردی چندرسانه‌ای در سیستم‌های چندپردازنده‌ای دارد.

هدف اصلی این رساله معرفی الگوریتم‌های بهینه زمان‌بندی برای برنامه‌های کاربردی چندرسانه‌ای در ساختارهای چندپردازنده‌ای می‌باشد. در گفتار اول، ضمن بیان ویژگی‌های برنامه‌های کاربردی چندرسانه‌ای و چندپردازنده‌ای، از طراحی زمان‌بند بعنوان یکی از چالش‌های مهم طراحی این‌گونه سیستم‌ها یاد می‌شود. هم‌چنین نواقص موجود در الگوریتم‌های زمان‌بندی طراحی شده به تفصیل مورد بررسی قرارگرفته و ساختار بهینه زمان‌بندی برای سیستم‌های چندرسانه‌ای در ساختارهای چندپردازنده‌ای معرفی می‌گردد. در گفتار دوم الگوریتم‌ها و ایده‌های جدید طراحی الگوریتم‌های زمان‌بندی بلادرنگ[۱۵] معرفی خواهد شد. ایده‌های معرفی شده بر پایه ساختار بهینه توصیف شده در گفتاراول طراحی گردیده است. در گفتار پایانی بستر کاری جدیدی را معرفی می‌نماییم که برای ارزیابی کارآیی برنامه‌های کاربردی در ساختارهای چندپردازنده‌ای مورد استفاده قرار می‌گیرد. این بستر کاری دارای سرعت بالایی در شبیه سازی و اجرای برنامه‌های کاربردی می‌باشد. هم‌چنین در مقایسه با مدل‌های مشابه نتایج بدست آمده از اجرای برنامه کاربردی در این ساختار دارای دقت بالایی می‌باشد. چگونگی پیاده‌سازی الگوریتم‌های توصیف شده در گفتار دوم استفاده از بستر کاری معرفی‌شده نیز در گفتار پایانی تشریح می‌شود. با ارزیابی و مقایسه نتایج استفاده از الگوریتم‌های طراحی شده این گفتار به پایان می‌رسد.

[13] Task
[14] Scheduling
[15] Real Time

گفتار اول

توصیف مسأله:

نیاز برنامه‌های کاربردی چندرسانه‌ای
و ساختارهای چندپردازنده‌ای
به الگوریتم‌های زمان‌بندی پویا

فصل ۲. معرفی ساختارهای چندپردازنده‌ای و برنامه‌های کاربردی چندرسانه‌ای